!!!!!!! DO NOT EDIT THIS FILE !!!!!!! # This file is machine-generated by lib/unicore/mktables from the Unicode # database, Version 5.2.0. Any changes made here will be lost!

To change this file, edit lib/unicore/mktables instead.

NAME

perluniprops - Index of Unicode Version 5.2.0 properties in Perl

DESCRIPTION

There are many properties in Unicode, and Perl provides access to almost all of them, as well as some additional extensions and short-cut synonyms.

And just about all of the few that aren't accessible through the Perl core are accessible through the modules: Unicode::Normalize and Unicode::UCD, and for Unihan properties, via the CPAN module Unicode::Unihan.

This document merely lists all available properties and does not attempt to explain what each property really means. There is a brief description of each Perl extension. There is some detail about Blocks, Scripts, General_Category, and Bidi_Class in *perlunicode*, but to find out about the intricacies of the Unicode properties, refer to the Unicode standard. A good starting place is *http://www.unicode.org/reports/tr44/*. More information on the Perl extensions is in *perlrecharclass*.

Note that you can define your own properties; see *"User-Defined Character Properties" in perlunicode* .

Properties accessible through \p{} and \P{}

The Perl regular expression \p{} and \P{} constructs give access to most of the Unicode character properties. The table below shows all these constructs, both single and compound forms.

Compound forms consist of two components, separated by an equals sign or a colon. The first component is the property name, and the second component is the particular value of the property to match against, for example, '\p{Script: Greek}' or '\p{Script=Greek}' both mean to match characters whose Script property is Greek.

Single forms, like '\p{Greek}', are mostly Perl-defined shortcuts for their equivalent compound forms. The table shows these equivalences. (In our example, '\p{Greek}' is a just a shortcut for '\p{Script=Greek}'.) There are also a few Perl-defined single forms that are not shortcuts for a compound form. One such is \p{Word}. These are also listed in the table.

In parsing these constructs, Perl always ignores Upper/lower case differences everywhere within the {braces}. Thus '\p{Greek}' means the same thing as '\p{greek}'. But note that changing the case of the 'p' or 'P' before the left brace completely changes the meaning of the construct, from "match" (for '\p{}') to "doesn't match" (for '\P{}'). Casing in this document is for improved legibility.

Also, white space, hyphens, and underscores are also normally ignored everywhere between the {braces}, and hence can be freely added or removed even if the `/x` modifier hasn't been specified on the regular expression. But a '**T**' at the beginning of an entry in the table below means that tighter (stricter) rules are used for that entry:

Single form (\p{name}) tighter rules:

> White space, hyphens, and underscores ARE significant except for:
>
> * white space adjacent to a non-word character
>
> * underscores separating digits in numbers
>
> That means, for example, that you can freely add or remove white space adjacent to (but within) the braces without affecting the meaning.

Compound form (\p{name=value} or \p{name:value}) tighter rules:

The tighter rules given above for the single form apply to everything to the right of the colon or equals; the looser rules still apply to everything to the left.

That means, for example, that you can freely add or remove white space adjacent to (but within) the braces and the colon or equal sign.

Some properties are considered obsolete, but still available. There are several varieties of obsolesence:

Obsolete

Properties marked with an '**O**' in the table are considered obsolete. At the time of this writing (Unicode version 5.2) there is no information in the Unicode standard about the implications of a property being obsolete.

Stabilized

Obsolete properties may be stabilized. This means that they are not actively maintained by Unicode, and will not be extended as new characters are added to the standard. Such properties are marked with an '**S**' in the table. At the time of this writing (Unicode version 5.2) there is no further information in the Unicode standard about the implications of a property being stabilized.

Deprecated

Obsolete properties may be deprecated. This means that their use is strongly discouraged, so much so that a warning will be issued if used, unless the regular expression is in the scope of a `no warnings 'deprecated'` statement. A '**D**' flags each such entry in the table, and the entry there for the longest, most descriptive version of the property will give the reason it is deprecated, and perhaps advice. Perl may issue such a warning, even for properties that aren't officially deprecated by Unicode, when there used to be characters or code points that were matched by them, but no longer. This is to warn you that your program may not work like it did on earlier Unicode releases.

A deprecated property may be made unavailable in a future Perl version, so it is best to move away from them.

Some Perl extensions are present for backwards compatibility and are discouraged from being used, but not obsolete. An '**X**' flags each such entry in the table.

Matches in the Block property have shortcuts that begin with 'In_'. For example, \p{Block=Latin1} can be written as \p{In_Latin1}. For backward compatibility, if there is no conflict with another shortcut, these may also be written as \p{Latin1} or \p{Is_Latin1}. But, N.B., there are numerous such conflicting shortcuts. Use of these forms for Block is discouraged, and are flagged as such, not only because of the potential confusion as to what is meant, but also because a later release of Unicode may preempt the shortcut, and your program would no longer be correct. Use the 'In_' form instead to avoid this, or even more clearly, use the compound form, e.g., \p{blk:latin1}. See *"Blocks" in perlunicode* for more information about this.

The table below has two columns. The left column contains the \p{} constructs to look up, possibly preceeded by the flags mentioned above; and the right column contains information about them, like a description, or synonyms. It shows both the single and compound forms for each property that has them. If the left column is a short name for a property, the right column will give its longer, more descriptive name; and if the left column is the longest name, the right column will show any equivalent shortest name, in both single and compound forms if applicable.

The right column will also caution you if a property means something different than what might normally be expected.

All single forms are Perl extensions; a few compound forms are as well, and are noted as such.

Numbers in (parentheses) indicate the total number of code points matched by the property. For emphasis, those properties that match no code points at all are listed as well in a separate section

following the table.

There is no description given for most non-Perl defined properties (See http://www.unicode.org/reports/tr44/ for that).

For compactness, '*****' is used as a wildcard instead of showing all possible combinations. For example, entries like:

```
\p{Gc: *}                                   \p{General_Category: *}
```

mean that 'Gc' is a synonym for 'General_Category', and anything that is valid for the latter is also valid for the former. Similarly,

```
\p{Is_*}                                    \p{*}
```

means that if and only if, for example, \p{Foo} exists, then \p{Is_Foo} and \p{IsFoo} are also valid and all mean the same thing. And similarly, \p{Foo=Bar} means the same as \p{Is_Foo=Bar} and \p{IsFoo=Bar}. '*' here is restricted to something not beginning with an underscore.

Also, in binary properties, 'Yes', 'T', and 'True' are all synonyms for 'Y'. And 'No', 'F', and 'False' are all synonyms for 'N'. The table shows 'Y*' and 'N*' to indicate this, and doesn't have separate entries for the other possibilities. Note that not all properties which have values 'Yes' and 'No' are binary, and they have all their values spelled out without using this wild card, and a `NOT` clause in their description that highlights their not being binary. These also require the compound form to match them, whereas true binary properties have both single and compound forms available.

Note that all non-essential underscores are removed in the display of the short names below.

Summary legend:

***** is a wild-card

(\d+) in the info column gives the number of code points matched by this property.

D means this is deprecated.

O means this is obsolete.

S means this is stabilized.

T means tighter (stricter) name matching applies.

X means use of this form is discouraged.

```
        NAME                            INFO

 X \p{Aegean_Numbers}       \p{Block=Aegean_Numbers} (64)
 T \p{Age: 1.1}             Code point's usage introduced in version
                              1.1 (33_979)
 T \p{Age: 2.0}             Code point's usage was introduced in
                              version 2.0; See also Property
                              'Present_In' (144_521)
 T \p{Age: 2.1}             Code point's usage was introduced in
                              version 2.1; See also Property
                              'Present_In' (2)
 T \p{Age: 3.0}             Code point's usage was introduced in
                              version 3.0; See also Property
                              'Present_In' (10_307)
 T \p{Age: 3.1}             Code point's usage was introduced in
                              version 3.1; See also Property
                              'Present_In' (44_978)
 T \p{Age: 3.2}             Code point's usage was introduced in
```

```
                                version 3.2; See also Property
                                'Present_In' (1016)
T \p{Age: 4.0}                  Code point's usage was introduced in
                                version 4.0; See also Property
                                'Present_In' (1226)
T \p{Age: 4.1}                  Code point's usage was introduced in
                                version 4.1; See also Property
                                'Present_In' (1273)
T \p{Age: 5.0}                  Code point's usage was introduced in
                                version 5.0; See also Property
                                'Present_In' (1369)
T \p{Age: 5.1}                  Code point's usage was introduced in
                                version 5.1; See also Property
                                'Present_In' (1624)
T \p{Age: 5.2}                  Code point's usage was introduced in
                                version 5.2; See also Property
                                'Present_In' (6648)
  \p{Age: Unassigned}           Code point's usage has not been assigned
                                in any Unicode release thus far.
                                (867_169)
  \p{AHex}                      \p{ASCII_Hex_Digit} (= \p{ASCII_Hex_Digit=
                                Y}) (22)
  \p{AHex: *}                   \p{ASCII_Hex_Digit: *}
  \p{All}                       \p{Any} (1_114_112)
  \p{Alnum}                     Alphabetic and (Decimal) Numeric (100_931)
  \p{Alpha}                     \p{Alphabetic=Y} (100_520)
  \p{Alpha: *}                  \p{Alphabetic: *}
  \p{Alphabetic}                \p{Alpha} (= \p{Alphabetic=Y}) (100_520)
  \p{Alphabetic: N*}            (Short: \p{Alpha=N}, \P{Alpha}) (1_013_592)
  \p{Alphabetic: Y*}            (Short: \p{Alpha=Y}, \p{Alpha}) (100_520)
X \p{Alphabetic_Presentation_Forms} \p{Block=
                                Alphabetic_Presentation_Forms} (80)
X \p{Ancient_Greek_Musical_Notation} \p{Block=
                                Ancient_Greek_Musical_Notation} (80)
X \p{Ancient_Greek_Numbers} \p{Block=Ancient_Greek_Numbers} (80)
X \p{Ancient_Symbols}           \p{Block=Ancient_Symbols} (64)
  \p{Any}                       [\x{0000}-\x{10FFFF}] (1_114_112)
  \p{Arab}                      \p{Arabic} (= \p{Script=Arabic}) (NOT
                                \p{Block=Arabic}) (1030)
  \p{Arabic}                    \p{Script=Arabic} (Short: \p{Arab}; NOT
                                \p{Block=Arabic}) (1030)
X \p{Arabic_Presentation_Forms_A} \p{Block=
                                Arabic_Presentation_Forms_A} (688)
X \p{Arabic_Presentation_Forms_B} \p{Block=
                                Arabic_Presentation_Forms_B} (144)
X \p{Arabic_Supplement}         \p{Block=Arabic_Supplement} (48)
  \p{Armenian}                  \p{Script=Armenian} (Short: \p{Armn}; NOT
                                \p{Block=Armenian}) (90)
  \p{Armi}                      \p{Imperial_Aramaic} (= \p{Script=
                                Imperial_Aramaic}) (NOT \p{Block=
                                Imperial_Aramaic}) (31)
  \p{Armn}                      \p{Armenian} (= \p{Script=Armenian}) (NOT
                                \p{Block=Armenian}) (90)
X \p{Arrows}                    \p{Block=Arrows} (112)
  \p{ASCII}                     \p{Block=Basic_Latin} [[:ASCII:]] (128)
  \p{ASCII_Hex_Digit}           \p{ASCII_Hex_Digit=Y} (Short: \p{AHex})
```

```
                                (22)
    \p{ASCII_Hex_Digit: N*} (Short: \p{AHex=N}, \P{AHex}) (1_114_090)
    \p{ASCII_Hex_Digit: Y*} (Short: \p{AHex=Y}, \p{AHex}) (22)
    \p{Assigned}            All assigned code points (246_877)
    \p{Avestan}             \p{Script=Avestan} (Short: \p{Avst}; NOT
                              \p{Block=Avestan}) (61)
    \p{Avst}                \p{Avestan} (= \p{Script=Avestan}) (NOT
                              \p{Block=Avestan}) (61)
    \p{Bali}                \p{Balinese} (= \p{Script=Balinese}) (NOT
                              \p{Block=Balinese}) (121)
    \p{Balinese}            \p{Script=Balinese} (Short: \p{Bali}; NOT
                              \p{Block=Balinese}) (121)
    \p{Bamu}                \p{Bamum} (= \p{Script=Bamum}) (NOT
                              \p{Block=Bamum}) (88)
    \p{Bamum}               \p{Script=Bamum} (Short: \p{Bamu}; NOT
                              \p{Block=Bamum}) (88)
  X \p{Basic_Latin}         \p{ASCII} (= \p{Block=Basic_Latin}) (128)
    \p{Bc: *}               \p{Bidi_Class: *}
    \p{Beng}                \p{Bengali} (= \p{Script=Bengali}) (NOT
                              \p{Block=Bengali}) (92)
    \p{Bengali}             \p{Script=Bengali} (Short: \p{Beng}; NOT
                              \p{Block=Bengali}) (92)
    \p{Bidi_C}              \p{Bidi_Control} (= \p{Bidi_Control=Y}) (7)
    \p{Bidi_C: *}           \p{Bidi_Control: *}
    \p{Bidi_Class: AL}      \p{Bidi_Class=Arabic_Letter} (1116)
    \p{Bidi_Class: AN}      \p{Bidi_Class=Arabic_Number} (48)
    \p{Bidi_Class: Arabic_Letter} (Short: \p{Bc=AL}) (1116)
    \p{Bidi_Class: Arabic_Number} (Short: \p{Bc=AN}) (48)
    \p{Bidi_Class: B}       \p{Bidi_Class=Paragraph_Separator} (7)
    \p{Bidi_Class: BN}      \p{Bidi_Class=Boundary_Neutral} (4016)
    \p{Bidi_Class: Boundary_Neutral} (Short: \p{Bc=BN}) (4016)
    \p{Bidi_Class: Common_Separator} (Short: \p{Bc=CS}) (15)
    \p{Bidi_Class: CS}      \p{Bidi_Class=Common_Separator} (15)
    \p{Bidi_Class: EN}      \p{Bidi_Class=European_Number} (131)
    \p{Bidi_Class: ES}      \p{Bidi_Class=European_Separator} (12)
    \p{Bidi_Class: ET}      \p{Bidi_Class=European_Terminator} (63)
    \p{Bidi_Class: European_Number} (Short: \p{Bc=EN}) (131)
    \p{Bidi_Class: European_Separator} (Short: \p{Bc=ES}) (12)
    \p{Bidi_Class: European_Terminator} (Short: \p{Bc=ET}) (63)
    \p{Bidi_Class: L}       \p{Bidi_Class=Left_To_Right} (1_099_541)
    \p{Bidi_Class: Left_To_Right} (Short: \p{Bc=L}) (1_099_541)
    \p{Bidi_Class: Left_To_Right_Embedding} (Short: \p{Bc=LRE}) (1)
    \p{Bidi_Class: Left_To_Right_Override} (Short: \p{Bc=LRO}) (1)
    \p{Bidi_Class: LRE}     \p{Bidi_Class=Left_To_Right_Embedding} (1)
    \p{Bidi_Class: LRO}     \p{Bidi_Class=Left_To_Right_Override} (1)
    \p{Bidi_Class: Nonspacing_Mark} (Short: \p{Bc=NSM}) (1173)
    \p{Bidi_Class: NSM}     \p{Bidi_Class=Nonspacing_Mark} (1173)
    \p{Bidi_Class: ON}      \p{Bidi_Class=Other_Neutral} (3523)
    \p{Bidi_Class: Other_Neutral} (Short: \p{Bc=ON}) (3523)
    \p{Bidi_Class: Paragraph_Separator} (Short: \p{Bc=B}) (7)
    \p{Bidi_Class: PDF}     \p{Bidi_Class=Pop_Directional_Format} (1)
    \p{Bidi_Class: Pop_Directional_Format} (Short: \p{Bc=PDF}) (1)
    \p{Bidi_Class: R}       \p{Bidi_Class=Right_To_Left} (4441)
    \p{Bidi_Class: Right_To_Left} (Short: \p{Bc=R}) (4441)
    \p{Bidi_Class: Right_To_Left_Embedding} (Short: \p{Bc=RLE}) (1)
    \p{Bidi_Class: Right_To_Left_Override} (Short: \p{Bc=RLO}) (1)
```

```
\p{Bidi_Class: RLE}     \p{Bidi_Class=Right_To_Left_Embedding} (1)
\p{Bidi_Class: RLO}     \p{Bidi_Class=Right_To_Left_Override} (1)
\p{Bidi_Class: S}       \p{Bidi_Class=Segment_Separator} (3)
\p{Bidi_Class: Segment_Separator} (Short: \p{Bc=S}) (3)
\p{Bidi_Class: White_Space} (Short: \p{Bc=WS}) (18)
\p{Bidi_Class: WS}      \p{Bidi_Class=White_Space} (18)
\p{Bidi_Control}        \p{Bidi_Control=Y} (Short: \p{BidiC}) (7)
\p{Bidi_Control: N*}    (Short: \p{BidiC=N}, \P{BidiC}) (1_114_105)
\p{Bidi_Control: Y*}    (Short: \p{BidiC=Y}, \p{BidiC}) (7)
\p{Bidi_M}              \p{Bidi_Mirrored} (= \p{Bidi_Mirrored=Y})
                          (543)
\p{Bidi_M: *}           \p{Bidi_Mirrored: *}
\p{Bidi_Mirrored}       \p{Bidi_Mirrored=Y} (Short: \p{BidiM})
                          (543)
\p{Bidi_Mirrored: N*}   (Short: \p{BidiM=N}, \P{BidiM}) (1_113_569)
\p{Bidi_Mirrored: Y*}   (Short: \p{BidiM=Y}, \p{BidiM}) (543)
\p{Blank}               \h, Horizontal white space (19)
\p{Blk: *}              \p{Block: *}
\p{Block: Aegean_Numbers} (Single: \p{InAegeanNumbers}) (64)
\p{Block: Alphabetic_Presentation_Forms} (Single:
                          \p{InAlphabeticPresentationForms}) (80)
\p{Block: Ancient_Greek_Musical_Notation} (Single:
                          \p{InAncientGreekMusicalNotation}) (80)
\p{Block: Ancient_Greek_Numbers} (Single:
                          \p{InAncientGreekNumbers}) (80)
\p{Block: Ancient_Symbols} (Single: \p{InAncientSymbols}) (64)
\p{Block: Arabic}       (Single: \p{InArabic}; NOT \p{Arabic} NOR
                          \p{Is_Arabic}) (256)
\p{Block: Arabic_Presentation_Forms_A} (Single:
                          \p{InArabicPresentationFormsA}) (688)
\p{Block: Arabic_Presentation_Forms_B} (Single:
                          \p{InArabicPresentationFormsB}) (144)
\p{Block: Arabic_Supplement} (Single: \p{InArabicSupplement}) (48)
\p{Block: Armenian}     (Single: \p{InArmenian}; NOT \p{Armenian}
                          NOR \p{Is_Armenian}) (96)
\p{Block: Arrows}       (Single: \p{InArrows}) (112)
\p{Block: ASCII}        \p{Block=Basic_Latin} (128)
\p{Block: Avestan}      (Single: \p{InAvestan}; NOT \p{Avestan}
                          NOR \p{Is_Avestan}) (64)
\p{Block: Balinese}     (Single: \p{InBalinese}; NOT \p{Balinese}
                          NOR \p{Is_Balinese}) (128)
\p{Block: Bamum}        (Single: \p{InBamum}; NOT \p{Bamum} NOR
                          \p{Is_Bamum}) (96)
\p{Block: Basic_Latin}  (Short: \p{Blk=ASCII}, \p{ASCII}) (128)
\p{Block: Bengali}      (Single: \p{InBengali}; NOT \p{Bengali}
                          NOR \p{Is_Bengali}) (128)
\p{Block: Block_Elements} (Single: \p{InBlockElements}) (32)
\p{Block: Bopomofo}     (Single: \p{InBopomofo}; NOT \p{Bopomofo}
                          NOR \p{Is_Bopomofo}) (48)
\p{Block: Bopomofo_Extended} (Single: \p{InBopomofoExtended}) (32)
\p{Block: Box_Drawing}  (Single: \p{InBoxDrawing}) (128)
\p{Block: Braille_Patterns} (Single: \p{InBraillePatterns}) (256)
\p{Block: Buginese}     (Single: \p{InBuginese}; NOT \p{Buginese}
                          NOR \p{Is_Buginese}) (32)
\p{Block: Buhid}        (Single: \p{InBuhid}; NOT \p{Buhid} NOR
                          \p{Is_Buhid}) (32)
```

```
\p{Block: Byzantine_Musical_Symbols} (Single:
                          \p{InByzantineMusicalSymbols}) (256)
\p{Block: Canadian_Syllabics} \p{Block=
                          Unified_Canadian_Aboriginal_Syllabics}
                          (640)
\p{Block: Carian}       (Single: \p{InCarian}; NOT \p{Carian} NOR
                          \p{Is_Carian}) (64)
\p{Block: Cham}         (Single: \p{InCham}; NOT \p{Cham} NOR
                          \p{Is_Cham}) (96)
\p{Block: Cherokee}     (Single: \p{InCherokee}; NOT \p{Cherokee}
                          NOR \p{Is_Cherokee}) (96)
\p{Block: CJK_Compatibility} (Single: \p{InCJKCompatibility}) (256)
\p{Block: CJK_Compatibility_Forms} (Single:
                          \p{InCJKCompatibilityForms}) (32)
\p{Block: CJK_Compatibility_Ideographs} (Single:
                          \p{InCJKCompatibilityIdeographs}) (512)
\p{Block: CJK_Compatibility_Ideographs_Supplement} (Single:
                          \p{InCJKCompatibilityIdeographs-
                          Supplement}) (544)
\p{Block: CJK_Radicals_Supplement} (Single:
                          \p{InCJKRadicalsSupplement}) (128)
\p{Block: CJK_Strokes}  (Single: \p{InCJKStrokes}) (48)
\p{Block: CJK_Symbols_And_Punctuation} (Single:
                          \p{InCJKSymbolsAndPunctuation}) (64)
\p{Block: CJK_Unified_Ideographs} (Single:
                          \p{InCJKUnifiedIdeographs}) (20_992)
\p{Block: CJK_Unified_Ideographs_Extension_A} (Single:
                          \p{InCJKUnifiedIdeographsExtensionA})
                          (6592)
\p{Block: CJK_Unified_Ideographs_Extension_B} (Single:
                          \p{InCJKUnifiedIdeographsExtensionB})
                          (42_720)
\p{Block: CJK_Unified_Ideographs_Extension_C} (Single:
                          \p{InCJKUnifiedIdeographsExtensionC})
                          (4160)
\p{Block: Combining_Diacritical_Marks} (Single:
                          \p{InCombiningDiacriticalMarks}) (112)
\p{Block: Combining_Diacritical_Marks_For_Symbols} (Short: \p{Blk=
                          CombiningMarksForSymbols},
                          \p{InCombiningMarksForSymbols}) (48)
\p{Block: Combining_Diacritical_Marks_Supplement} (Single:
                          \p{InCombiningDiacriticalMarks-
                          Supplement}) (64)
\p{Block: Combining_Half_Marks} (Single: \p{InCombiningHalfMarks})
                          (16)
\p{Block: Combining_Marks_For_Symbols} \p{Block=
                          Combining_Diacritical_Marks_For_Symbols}
                          (48)
\p{Block: Common_Indic_Number_Forms} (Single:
                          \p{InCommonIndicNumberForms}) (16)
\p{Block: Control_Pictures} (Single: \p{InControlPictures}) (64)
\p{Block: Coptic}       (Single: \p{InCoptic}; NOT \p{Coptic} NOR
                          \p{Is_Coptic}) (128)
\p{Block: Counting_Rod_Numerals} (Single:
                          \p{InCountingRodNumerals}) (32)
\p{Block: Cuneiform}    (Single: \p{InCuneiform}; NOT
```

```
                              \p{Cuneiform} NOR \p{Is_Cuneiform})
                              (1024)
\p{Block: Cuneiform_Numbers_And_Punctuation} (Single:
                              \p{InCuneiformNumbersAndPunctuation})
                              (128)
\p{Block: Currency_Symbols} (Single: \p{InCurrencySymbols}) (48)
\p{Block: Cypriot_Syllabary} (Single: \p{InCypriotSyllabary}) (64)
\p{Block: Cyrillic}     (Single: \p{InCyrillic}; NOT \p{Cyrillic}
                              NOR \p{Is_Cyrillic}) (256)
\p{Block: Cyrillic_Extended_A} (Single: \p{InCyrillicExtendedA})
                              (32)
\p{Block: Cyrillic_Extended_B} (Single: \p{InCyrillicExtendedB})
                              (96)
\p{Block: Cyrillic_Supplement} (Single: \p{InCyrillicSupplement})
                              (48)
\p{Block: Cyrillic_Supplementary} \p{Block=Cyrillic_Supplement}
                              (48)
\p{Block: Deseret}      (Single: \p{InDeseret}) (80)
\p{Block: Devanagari}   (Single: \p{InDevanagari}; NOT
                              \p{Devanagari} NOR \p{Is_Devanagari})
                              (128)
\p{Block: Devanagari_Extended} (Single: \p{InDevanagariExtended})
                              (32)
\p{Block: Dingbats}     (Single: \p{InDingbats}) (192)
\p{Block: Domino_Tiles} (Single: \p{InDominoTiles}) (112)
\p{Block: Egyptian_Hieroglyphs} (Single:
                              \p{InEgyptianHieroglyphs}; NOT
                              \p{Egyptian_Hieroglyphs} NOR
                              \p{Is_Egyptian_Hieroglyphs}) (1072)
\p{Block: Enclosed_Alphanumeric_Supplement} (Single:
                              \p{InEnclosedAlphanumericSupplement})
                              (256)
\p{Block: Enclosed_Alphanumerics} (Single:
                              \p{InEnclosedAlphanumerics}) (160)
\p{Block: Enclosed_CJK_Letters_And_Months} (Single:
                              \p{InEnclosedCJKLettersAndMonths}) (256)
\p{Block: Enclosed_Ideographic_Supplement} (Single:
                              \p{InEnclosedIdeographicSupplement})
                              (256)
\p{Block: Ethiopic}     (Single: \p{InEthiopic}; NOT \p{Ethiopic}
                              NOR \p{Is_Ethiopic}) (384)
\p{Block: Ethiopic_Extended} (Single: \p{InEthiopicExtended}) (96)
\p{Block: Ethiopic_Supplement} (Single: \p{InEthiopicSupplement})
                              (32)
\p{Block: General_Punctuation} (Single: \p{InGeneralPunctuation})
                              (112)
\p{Block: Geometric_Shapes} (Single: \p{InGeometricShapes}) (96)
\p{Block: Georgian}     (Single: \p{InGeorgian}; NOT \p{Georgian}
                              NOR \p{Is_Georgian}) (96)
\p{Block: Georgian_Supplement} (Single: \p{InGeorgianSupplement})
                              (48)
\p{Block: Glagolitic}   (Single: \p{InGlagolitic}; NOT
                              \p{Glagolitic} NOR \p{Is_Glagolitic})
                              (96)
\p{Block: Gothic}       (Single: \p{InGothic}; NOT \p{Gothic} NOR
                              \p{Is_Gothic}) (32)
```

```
\p{Block: Greek}        \p{Block=Greek_And_Coptic} (NOT \p{Greek}
                          NOR \p{Is_Greek}) (144)
\p{Block: Greek_And_Coptic} (Short: \p{Blk=Greek}, \p{InGreek};
                          NOT \p{Greek} NOR \p{Is_Greek}) (144)
\p{Block: Greek_Extended} (Single: \p{InGreekExtended}) (256)
\p{Block: Gujarati}     (Single: \p{InGujarati}; NOT \p{Gujarati}
                          NOR \p{Is_Gujarati}) (128)
\p{Block: Gurmukhi}     (Single: \p{InGurmukhi}; NOT \p{Gurmukhi}
                          NOR \p{Is_Gurmukhi}) (128)
\p{Block: Halfwidth_And_Fullwidth_Forms} (Single:
                          \p{InHalfwidthAndFullwidthForms}) (240)
\p{Block: Hangul_Compatibility_Jamo} (Single:
                          \p{InHangulCompatibilityJamo}) (96)
\p{Block: Hangul_Jamo}  (Single: \p{InHangulJamo}) (256)
\p{Block: Hangul_Jamo_Extended_A} (Single:
                          \p{InHangulJamoExtendedA}) (32)
\p{Block: Hangul_Jamo_Extended_B} (Single:
                          \p{InHangulJamoExtendedB}) (80)
\p{Block: Hangul_Syllables} (Single: \p{InHangulSyllables})
                          (11_184)
\p{Block: Hanunoo}      (Single: \p{InHanunoo}; NOT \p{Hanunoo}
                          NOR \p{Is_Hanunoo}) (32)
\p{Block: Hebrew}       (Single: \p{InHebrew}; NOT \p{Hebrew} NOR
                          \p{Is_Hebrew}) (112)
\p{Block: High_Private_Use_Surrogates} (Single:
                          \p{InHighPrivateUseSurrogates}) (128)
\p{Block: High_Surrogates} (Single: \p{InHighSurrogates}) (896)
\p{Block: Hiragana}     (Single: \p{InHiragana}; NOT \p{Hiragana}
                          NOR \p{Is_Hiragana}) (96)
\p{Block: Ideographic_Description_Characters} (Single:
                          \p{InIdeographicDescriptionCharacters})
                          (16)
\p{Block: Imperial_Aramaic} (Single: \p{InImperialAramaic}; NOT
                          \p{Imperial_Aramaic} NOR
                          \p{Is_Imperial_Aramaic}) (32)
\p{Block: Inscriptional_Pahlavi} (Single:
                          \p{InInscriptionalPahlavi}; NOT
                          \p{Inscriptional_Pahlavi} NOR
                          \p{Is_Inscriptional_Pahlavi}) (32)
\p{Block: Inscriptional_Parthian} (Single:
                          \p{InInscriptionalParthian}; NOT
                          \p{Inscriptional_Parthian} NOR
                          \p{Is_Inscriptional_Parthian}) (32)
\p{Block: IPA_Extensions} (Single: \p{InIPAExtensions}) (96)
\p{Block: Javanese}     (Single: \p{InJavanese}; NOT \p{Javanese}
                          NOR \p{Is_Javanese}) (96)
\p{Block: Kaithi}       (Single: \p{InKaithi}; NOT \p{Kaithi} NOR
                          \p{Is_Kaithi}) (80)
\p{Block: Kanbun}       (Single: \p{InKanbun}) (16)
\p{Block: Kangxi_Radicals} (Single: \p{InKangxiRadicals}) (224)
\p{Block: Kannada}      (Single: \p{InKannada}; NOT \p{Kannada}
                          NOR \p{Is_Kannada}) (128)
\p{Block: Katakana}     (Single: \p{InKatakana}; NOT \p{Katakana}
                          NOR \p{Is_Katakana}) (96)
\p{Block: Katakana_Phonetic_Extensions} (Single:
                          \p{InKatakanaPhoneticExtensions}) (16)
```

```
\p{Block: Kayah_Li}     (Single: \p{InKayahLi}) (48)
\p{Block: Kharoshthi}   (Single: \p{InKharoshthi}; NOT
                          \p{Kharoshthi} NOR \p{Is_Kharoshthi})
                          (96)
\p{Block: Khmer}        (Single: \p{InKhmer}; NOT \p{Khmer} NOR
                          \p{Is_Khmer}) (128)
\p{Block: Khmer_Symbols} (Single: \p{InKhmerSymbols}) (32)
\p{Block: Lao}          (Single: \p{InLao}; NOT \p{Lao} NOR
                          \p{Is_Lao}) (128)
\p{Block: Latin_1}      \p{Block=Latin_1_Supplement} (128)
\p{Block: Latin_1_Supplement} (Short: \p{Blk=Latin1},
                          \p{InLatin1}) (128)
\p{Block: Latin_Extended_A} (Single: \p{InLatinExtendedA}) (128)
\p{Block: Latin_Extended_Additional} (Single:
                          \p{InLatinExtendedAdditional}) (256)
\p{Block: Latin_Extended_B} (Single: \p{InLatinExtendedB}) (208)
\p{Block: Latin_Extended_C} (Single: \p{InLatinExtendedC}) (32)
\p{Block: Latin_Extended_D} (Single: \p{InLatinExtendedD}) (224)
\p{Block: Lepcha}       (Single: \p{InLepcha}; NOT \p{Lepcha} NOR
                          \p{Is_Lepcha}) (80)
\p{Block: Letterlike_Symbols} (Single: \p{InLetterlikeSymbols})
                          (80)
\p{Block: Limbu}        (Single: \p{InLimbu}; NOT \p{Limbu} NOR
                          \p{Is_Limbu}) (80)
\p{Block: Linear_B_Ideograms} (Single: \p{InLinearBIdeograms})
                          (128)
\p{Block: Linear_B_Syllabary} (Single: \p{InLinearBSyllabary})
                          (128)
\p{Block: Lisu}         (Single: \p{InLisu}) (48)
\p{Block: Low_Surrogates} (Single: \p{InLowSurrogates}) (1024)
\p{Block: Lycian}       (Single: \p{InLycian}; NOT \p{Lycian} NOR
                          \p{Is_Lycian}) (32)
\p{Block: Lydian}       (Single: \p{InLydian}; NOT \p{Lydian} NOR
                          \p{Is_Lydian}) (32)
\p{Block: Mahjong_Tiles} (Single: \p{InMahjongTiles}) (48)
\p{Block: Malayalam}    (Single: \p{InMalayalam}; NOT
                          \p{Malayalam} NOR \p{Is_Malayalam}) (128)
\p{Block: Mathematical_Alphanumeric_Symbols} (Single:
                          \p{InMathematicalAlphanumericSymbols})
                          (1024)
\p{Block: Mathematical_Operators} (Single:
                          \p{InMathematicalOperators}) (256)
\p{Block: Meetei_Mayek} (Single: \p{InMeeteiMayek}; NOT
                          \p{Meetei_Mayek} NOR
                          \p{Is_Meetei_Mayek}) (64)
\p{Block: Miscellaneous_Mathematical_Symbols_A} (Single:
                          \p{InMiscellaneousMathematicalSymbolsA})
                          (48)
\p{Block: Miscellaneous_Mathematical_Symbols_B} (Single:
                          \p{InMiscellaneousMathematicalSymbolsB})
                          (128)
\p{Block: Miscellaneous_Symbols} (Single:
                          \p{InMiscellaneousSymbols}) (256)
\p{Block: Miscellaneous_Symbols_And_Arrows} (Single:
                          \p{InMiscellaneousSymbolsAndArrows})
                          (256)
```

```
\p{Block: Miscellaneous_Technical} (Single:
                          \p{InMiscellaneousTechnical}) (256)
\p{Block: Modifier_Tone_Letters} (Single:
                          \p{InModifierToneLetters}) (32)
\p{Block: Mongolian}    (Single: \p{InMongolian}; NOT
                          \p{Mongolian} NOR \p{Is_Mongolian}) (176)
\p{Block: Musical_Symbols} (Single: \p{InMusicalSymbols}) (256)
\p{Block: Myanmar}      (Single: \p{InMyanmar}; NOT \p{Myanmar}
                          NOR \p{Is_Myanmar}) (160)
\p{Block: Myanmar_Extended_A} (Single: \p{InMyanmarExtendedA}) (32)
\p{Block: New_Tai_Lue}  (Single: \p{InNewTaiLue}; NOT
                          \p{New_Tai_Lue} NOR \p{Is_New_Tai_Lue})
                          (96)
\p{Block: NKo}          (Single: \p{InNKo}; NOT \p{Nko} NOR
                          \p{Is_NKo}) (64)
\p{Block: No_Block}     (Single: \p{InNoBlock}) (864_192)
\p{Block: Number_Forms} (Single: \p{InNumberForms}) (64)
\p{Block: Ogham}        (Single: \p{InOgham}; NOT \p{Ogham} NOR
                          \p{Is_Ogham}) (32)
\p{Block: Ol_Chiki}     (Single: \p{InOlChiki}) (48)
\p{Block: Old_Italic}   (Single: \p{InOldItalic}; NOT
                          \p{Old_Italic} NOR \p{Is_Old_Italic})
                          (48)
\p{Block: Old_Persian}  (Single: \p{InOldPersian}; NOT
                          \p{Old_Persian} NOR \p{Is_Old_Persian})
                          (64)
\p{Block: Old_South_Arabian} (Single: \p{InOldSouthArabian}) (32)
\p{Block: Old_Turkic}   (Single: \p{InOldTurkic}; NOT
                          \p{Old_Turkic} NOR \p{Is_Old_Turkic})
                          (80)
\p{Block: Optical_Character_Recognition} (Single:
                          \p{InOpticalCharacterRecognition}) (32)
\p{Block: Oriya}        (Single: \p{InOriya}; NOT \p{Oriya} NOR
                          \p{Is_Oriya}) (128)
\p{Block: Osmanya}      (Single: \p{InOsmanya}; NOT \p{Osmanya}
                          NOR \p{Is_Osmanya}) (48)
\p{Block: Phags_Pa}     (Single: \p{InPhagsPa}; NOT \p{Phags_Pa}
                          NOR \p{Is_Phags_Pa}) (64)
\p{Block: Phaistos_Disc} (Single: \p{InPhaistosDisc}) (48)
\p{Block: Phoenician}   (Single: \p{InPhoenician}; NOT
                          \p{Phoenician} NOR \p{Is_Phoenician})
                          (32)
\p{Block: Phonetic_Extensions} (Single: \p{InPhoneticExtensions})
                          (128)
\p{Block: Phonetic_Extensions_Supplement} (Single:
                          \p{InPhoneticExtensionsSupplement}) (64)
\p{Block: Private_Use}  \p{Block=Private_Use_Area} (NOT
                          \p{Private_Use} NOR \p{Is_Private_Use})
                          (6400)
\p{Block: Private_Use_Area} (Short: \p{Blk=PrivateUse},
                          \p{InPrivateUse}; NOT \p{Private_Use}
                          NOR \p{Is_Private_Use}) (6400)
\p{Block: Rejang}       (Single: \p{InRejang}; NOT \p{Rejang} NOR
                          \p{Is_Rejang}) (48)
\p{Block: Rumi_Numeral_Symbols} (Single: \p{InRumiNumeralSymbols})
                          (32)
```

```
\p{Block: Runic}        (Single: \p{InRunic}; NOT \p{Runic} NOR
                          \p{Is_Runic}) (96)
\p{Block: Samaritan}    (Single: \p{InSamaritan}; NOT
                          \p{Samaritan} NOR \p{Is_Samaritan}) (64)
\p{Block: Saurashtra}   (Single: \p{InSaurashtra}; NOT
                          \p{Saurashtra} NOR \p{Is_Saurashtra})
                          (96)
\p{Block: Shavian}      (Single: \p{InShavian}) (48)
\p{Block: Sinhala}      (Single: \p{InSinhala}; NOT \p{Sinhala}
                          NOR \p{Is_Sinhala}) (128)
\p{Block: Small_Form_Variants} (Single: \p{InSmallFormVariants})
                          (32)
\p{Block: Spacing_Modifier_Letters} (Single:
                          \p{InSpacingModifierLetters}) (80)
\p{Block: Specials}     (Single: \p{InSpecials}) (16)
\p{Block: Sundanese}    (Single: \p{InSundanese}; NOT
                          \p{Sundanese} NOR \p{Is_Sundanese}) (64)
\p{Block: Superscripts_And_Subscripts} (Single:
                          \p{InSuperscriptsAndSubscripts}) (48)
\p{Block: Supplemental_Arrows_A} (Single:
                          \p{InSupplementalArrowsA}) (16)
\p{Block: Supplemental_Arrows_B} (Single:
                          \p{InSupplementalArrowsB}) (128)
\p{Block: Supplemental_Mathematical_Operators} (Single:
                          \p{InSupplementalMathematicalOperators})
                          (256)
\p{Block: Supplemental_Punctuation} (Single:
                          \p{InSupplementalPunctuation}) (128)
\p{Block: Supplementary_Private_Use_Area_A} (Single:
                          \p{InSupplementaryPrivateUseAreaA})
                          (65_536)
\p{Block: Supplementary_Private_Use_Area_B} (Single:
                          \p{InSupplementaryPrivateUseAreaB})
                          (65_536)
\p{Block: Syloti_Nagri} (Single: \p{InSylotiNagri}; NOT
                          \p{Syloti_Nagri} NOR
                          \p{Is_Syloti_Nagri}) (48)
\p{Block: Syriac}       (Single: \p{InSyriac}; NOT \p{Syriac} NOR
                          \p{Is_Syriac}) (80)
\p{Block: Tagalog}      (Single: \p{InTagalog}; NOT \p{Tagalog}
                          NOR \p{Is_Tagalog}) (32)
\p{Block: Tagbanwa}     (Single: \p{InTagbanwa}; NOT \p{Tagbanwa}
                          NOR \p{Is_Tagbanwa}) (32)
\p{Block: Tags}         (Single: \p{InTags}) (128)
\p{Block: Tai_Le}       (Single: \p{InTaiLe}; NOT \p{Tai_Le} NOR
                          \p{Is_Tai_Le}) (48)
\p{Block: Tai_Tham}     (Single: \p{InTaiTham}; NOT \p{Tai_Tham}
                          NOR \p{Is_Tai_Tham}) (144)
\p{Block: Tai_Viet}     (Single: \p{InTaiViet}; NOT \p{Tai_Viet}
                          NOR \p{Is_Tai_Viet}) (96)
\p{Block: Tai_Xuan_Jing_Symbols} (Single:
                          \p{InTaiXuanJingSymbols}) (96)
\p{Block: Tamil}        (Single: \p{InTamil}; NOT \p{Tamil} NOR
                          \p{Is_Tamil}) (128)
\p{Block: Telugu}       (Single: \p{InTelugu}; NOT \p{Telugu} NOR
                          \p{Is_Telugu}) (128)
```

```
    \p{Block: Thaana}       (Single: \p{InThaana}; NOT \p{Thaana} NOR
                              \p{Is_Thaana}) (64)
    \p{Block: Thai}         (Single: \p{InThai}; NOT \p{Thai} NOR
                              \p{Is_Thai}) (128)
    \p{Block: Tibetan}      (Single: \p{InTibetan}; NOT \p{Tibetan}
                              NOR \p{Is_Tibetan}) (256)
    \p{Block: Tifinagh}     (Single: \p{InTifinagh}; NOT \p{Tifinagh}
                              NOR \p{Is_Tifinagh}) (80)
    \p{Block: Ugaritic}     (Single: \p{InUgaritic}; NOT \p{Ugaritic}
                              NOR \p{Is_Ugaritic}) (32)
    \p{Block: Unified_Canadian_Aboriginal_Syllabics} (Short: \p{Blk=
                              CanadianSyllabics},
                              \p{InCanadianSyllabics}) (640)
    \p{Block: Unified_Canadian_Aboriginal_Syllabics_Extended} (Single:
                              \p{InUnifiedCanadianAboriginalSyllabics-
                              Extended}) (80)
    \p{Block: Vai}          (Single: \p{InVai}; NOT \p{Vai} NOR
                              \p{Is_Vai}) (320)
    \p{Block: Variation_Selectors} (Single: \p{InVariationSelectors})
                              (16)
    \p{Block: Variation_Selectors_Supplement} (Single:
                              \p{InVariationSelectorsSupplement}) (240)
    \p{Block: Vedic_Extensions} (Single: \p{InVedicExtensions}) (48)
    \p{Block: Vertical_Forms} (Single: \p{InVerticalForms}) (16)
    \p{Block: Yi_Radicals}  (Single: \p{InYiRadicals}) (64)
    \p{Block: Yi_Syllables} (Single: \p{InYiSyllables}) (1168)
    \p{Block: Yijing_Hexagram_Symbols} (Single:
                              \p{InYijingHexagramSymbols}) (64)
X   \p{Block_Elements}      \p{Block=Block_Elements} (32)
    \p{Bopo}                \p{Bopomofo} (= \p{Script=Bopomofo}) (NOT
                              \p{Block=Bopomofo}) (65)
    \p{Bopomofo}            \p{Script=Bopomofo} (Short: \p{Bopo}; NOT
                              \p{Block=Bopomofo}) (65)
X   \p{Bopomofo_Extended}   \p{Block=Bopomofo_Extended} (32)
X   \p{Box_Drawing}         \p{Block=Box_Drawing} (128)
    \p{Brai}                \p{Braille} (= \p{Script=Braille}) (256)
    \p{Braille}             \p{Script=Braille} (Short: \p{Brai}) (256)
X   \p{Braille_Patterns}    \p{Block=Braille_Patterns} (256)
    \p{Bugi}                \p{Buginese} (= \p{Script=Buginese}) (NOT
                              \p{Block=Buginese}) (30)
    \p{Buginese}            \p{Script=Buginese} (Short: \p{Bugi}; NOT
                              \p{Block=Buginese}) (30)
    \p{Buhd}                \p{Buhid} (= \p{Script=Buhid}) (NOT
                              \p{Block=Buhid}) (20)
    \p{Buhid}               \p{Script=Buhid} (Short: \p{Buhd}; NOT
                              \p{Block=Buhid}) (20)
X   \p{Byzantine_Musical_Symbols} \p{Block=Byzantine_Musical_Symbols}
                              (256)
    \p{C}                   \p{Other} (= \p{General_Category=Other})
                              (1_006_956)
    \p{Canadian_Aboriginal} \p{Script=Canadian_Aboriginal} (Short:
                              \p{Cans}) (710)
X   \p{Canadian_Syllabics}  \p{Unified_Canadian_Aboriginal_Syllabics}
                              (= \p{Block=
                              Unified_Canadian_Aboriginal_Syllabics})
                              (640)
```

```
T \p{Canonical_Combining_Class: 0} \p{Canonical_Combining_Class=
                            Not_Reordered} (1_113_518)
T \p{Canonical_Combining_Class: 1} \p{Canonical_Combining_Class=
                            Overlay} (26)
T \p{Canonical_Combining_Class: 7} \p{Canonical_Combining_Class=
                            Nukta} (11)
T \p{Canonical_Combining_Class: 8} \p{Canonical_Combining_Class=
                            Kana_Voicing} (2)
T \p{Canonical_Combining_Class: 9} \p{Canonical_Combining_Class=
                            Virama} (27)
T \p{Canonical_Combining_Class: 10} (Short: \p{Ccc=10}) (1)
T \p{Canonical_Combining_Class: 11} (Short: \p{Ccc=11}) (1)
T \p{Canonical_Combining_Class: 12} (Short: \p{Ccc=12}) (1)
T \p{Canonical_Combining_Class: 13} (Short: \p{Ccc=13}) (1)
T \p{Canonical_Combining_Class: 14} (Short: \p{Ccc=14}) (1)
T \p{Canonical_Combining_Class: 15} (Short: \p{Ccc=15}) (1)
T \p{Canonical_Combining_Class: 16} (Short: \p{Ccc=16}) (1)
T \p{Canonical_Combining_Class: 17} (Short: \p{Ccc=17}) (1)
T \p{Canonical_Combining_Class: 18} (Short: \p{Ccc=18}) (2)
T \p{Canonical_Combining_Class: 19} (Short: \p{Ccc=19}) (2)
T \p{Canonical_Combining_Class: 20} (Short: \p{Ccc=20}) (1)
T \p{Canonical_Combining_Class: 21} (Short: \p{Ccc=21}) (1)
T \p{Canonical_Combining_Class: 22} (Short: \p{Ccc=22}) (1)
T \p{Canonical_Combining_Class: 23} (Short: \p{Ccc=23}) (1)
T \p{Canonical_Combining_Class: 24} (Short: \p{Ccc=24}) (1)
T \p{Canonical_Combining_Class: 25} (Short: \p{Ccc=25}) (1)
T \p{Canonical_Combining_Class: 26} (Short: \p{Ccc=26}) (1)
T \p{Canonical_Combining_Class: 27} (Short: \p{Ccc=27}) (1)
T \p{Canonical_Combining_Class: 28} (Short: \p{Ccc=28}) (1)
T \p{Canonical_Combining_Class: 29} (Short: \p{Ccc=29}) (1)
T \p{Canonical_Combining_Class: 30} (Short: \p{Ccc=30}) (2)
T \p{Canonical_Combining_Class: 31} (Short: \p{Ccc=31}) (2)
T \p{Canonical_Combining_Class: 32} (Short: \p{Ccc=32}) (2)
T \p{Canonical_Combining_Class: 33} (Short: \p{Ccc=33}) (1)
T \p{Canonical_Combining_Class: 34} (Short: \p{Ccc=34}) (1)
T \p{Canonical_Combining_Class: 35} (Short: \p{Ccc=35}) (1)
T \p{Canonical_Combining_Class: 36} (Short: \p{Ccc=36}) (1)
T \p{Canonical_Combining_Class: 84} (Short: \p{Ccc=84}) (1)
T \p{Canonical_Combining_Class: 91} (Short: \p{Ccc=91}) (1)
T \p{Canonical_Combining_Class: 103} (Short: \p{Ccc=103}) (2)
T \p{Canonical_Combining_Class: 107} (Short: \p{Ccc=107}) (4)
T \p{Canonical_Combining_Class: 118} (Short: \p{Ccc=118}) (2)
T \p{Canonical_Combining_Class: 122} (Short: \p{Ccc=122}) (4)
T \p{Canonical_Combining_Class: 129} (Short: \p{Ccc=129}) (1)
T \p{Canonical_Combining_Class: 130} (Short: \p{Ccc=130}) (6)
T \p{Canonical_Combining_Class: 132} (Short: \p{Ccc=132}) (1)
T \p{Canonical_Combining_Class: 200} \p{Canonical_Combining_Class=
                            Attached_Below_Left} (0)
T \p{Canonical_Combining_Class: 202} \p{Canonical_Combining_Class=
                            Attached_Below} (5)
T \p{Canonical_Combining_Class: 214} \p{Canonical_Combining_Class=
                            Attached_Above} (1)
T \p{Canonical_Combining_Class: 216} \p{Canonical_Combining_Class=
                            Attached_Above_Right} (9)
T \p{Canonical_Combining_Class: 218} \p{Canonical_Combining_Class=
                            Below_Left} (1)
```

```
T \p{Canonical_Combining_Class: 220} \p{Canonical_Combining_Class=
                            Below} (117)
T \p{Canonical_Combining_Class: 222} \p{Canonical_Combining_Class=
                            Below_Right} (4)
T \p{Canonical_Combining_Class: 224} \p{Canonical_Combining_Class=
                            Left} (2)
T \p{Canonical_Combining_Class: 226} \p{Canonical_Combining_Class=
                            Right} (1)
T \p{Canonical_Combining_Class: 228} \p{Canonical_Combining_Class=
                            Above_Left} (3)
T \p{Canonical_Combining_Class: 230} \p{Canonical_Combining_Class=
                            Above} (318)
T \p{Canonical_Combining_Class: 232} \p{Canonical_Combining_Class=
                            Above_Right} (4)
T \p{Canonical_Combining_Class: 233} \p{Canonical_Combining_Class=
                            Double_Below} (3)
T \p{Canonical_Combining_Class: 234} \p{Canonical_Combining_Class=
                            Double_Above} (5)
T \p{Canonical_Combining_Class: 240} \p{Canonical_Combining_Class=
                            Iota_Subscript} (1)
  \p{Canonical_Combining_Class: A} \p{Canonical_Combining_Class=
                            Above} (318)
  \p{Canonical_Combining_Class: Above} (Short: \p{Ccc=A}) (318)
  \p{Canonical_Combining_Class: Above_Left} (Short: \p{Ccc=AL}) (3)
  \p{Canonical_Combining_Class: Above_Right} (Short: \p{Ccc=AR}) (4)
  \p{Canonical_Combining_Class: AL} \p{Canonical_Combining_Class=
                            Above_Left} (3)
  \p{Canonical_Combining_Class: AR} \p{Canonical_Combining_Class=
                            Above_Right} (4)
  \p{Canonical_Combining_Class: ATA} \p{Canonical_Combining_Class=
                            Attached_Above} (1)
  \p{Canonical_Combining_Class: ATAR} \p{Canonical_Combining_Class=
                            Attached_Above_Right} (9)
  \p{Canonical_Combining_Class: ATB} \p{Canonical_Combining_Class=
                            Attached_Below} (5)
  \p{Canonical_Combining_Class: ATBL} \p{Canonical_Combining_Class=
                            Attached_Below_Left} (0)
  \p{Canonical_Combining_Class: Attached_Above} (Short: \p{Ccc=ATA})
                            (1)
  \p{Canonical_Combining_Class: Attached_Above_Right} (Short:
                            \p{Ccc=ATAR}) (9)
  \p{Canonical_Combining_Class: Attached_Below} (Short: \p{Ccc=ATB})
                            (5)
  \p{Canonical_Combining_Class: Attached_Below_Left} (Short: \p{Ccc=
                            ATBL}) (0)
  \p{Canonical_Combining_Class: B} \p{Canonical_Combining_Class=
                            Below} (117)
  \p{Canonical_Combining_Class: Below} (Short: \p{Ccc=B}) (117)
  \p{Canonical_Combining_Class: Below_Left} (Short: \p{Ccc=BL}) (1)
  \p{Canonical_Combining_Class: Below_Right} (Short: \p{Ccc=BR}) (4)
  \p{Canonical_Combining_Class: BL} \p{Canonical_Combining_Class=
                            Below_Left} (1)
  \p{Canonical_Combining_Class: BR} \p{Canonical_Combining_Class=
                            Below_Right} (4)
  \p{Canonical_Combining_Class: DA} \p{Canonical_Combining_Class=
                            Double_Above} (5)
```

```
\p{Canonical_Combining_Class: DB} \p{Canonical_Combining_Class=
                          Double_Below} (3)
\p{Canonical_Combining_Class: Double_Above} (Short: \p{Ccc=DA}) (5)
\p{Canonical_Combining_Class: Double_Below} (Short: \p{Ccc=DB}) (3)
\p{Canonical_Combining_Class: Iota_Subscript} (Short: \p{Ccc=IS})
                          (1)
\p{Canonical_Combining_Class: IS} \p{Canonical_Combining_Class=
                          Iota_Subscript} (1)
\p{Canonical_Combining_Class: Kana_Voicing} (Short: \p{Ccc=KV}) (2)
\p{Canonical_Combining_Class: KV} \p{Canonical_Combining_Class=
                          Kana_Voicing} (2)
\p{Canonical_Combining_Class: L} \p{Canonical_Combining_Class=
                          Left} (2)
\p{Canonical_Combining_Class: Left} (Short: \p{Ccc=L}) (2)
\p{Canonical_Combining_Class: NK} \p{Canonical_Combining_Class=
                          Nukta} (11)
\p{Canonical_Combining_Class: Not_Reordered} (Short: \p{Ccc=NR})
                          (1_113_518)
\p{Canonical_Combining_Class: NR} \p{Canonical_Combining_Class=
                          Not_Reordered} (1_113_518)
\p{Canonical_Combining_Class: Nukta} (Short: \p{Ccc=NK}) (11)
\p{Canonical_Combining_Class: OV} \p{Canonical_Combining_Class=
                          Overlay} (26)
\p{Canonical_Combining_Class: Overlay} (Short: \p{Ccc=OV}) (26)
\p{Canonical_Combining_Class: R} \p{Canonical_Combining_Class=
                          Right} (1)
\p{Canonical_Combining_Class: Right} (Short: \p{Ccc=R}) (1)
\p{Canonical_Combining_Class: Virama} (Short: \p{Ccc=VR}) (27)
\p{Canonical_Combining_Class: VR} \p{Canonical_Combining_Class=
                          Virama} (27)
\p{Cans}                \p{Canadian_Aboriginal} (= \p{Script=
                          Canadian_Aboriginal}) (710)
\p{Cari}                \p{Carian} (= \p{Script=Carian}) (NOT
                          \p{Block=Carian}) (49)
\p{Carian}              \p{Script=Carian} (Short: \p{Cari}; NOT
                          \p{Block=Carian}) (49)
\p{Case_Ignorable}      \p{Case_Ignorable=Y} (Short: \p{CI}) (1632)
\p{Case_Ignorable: N*}  (Short: \p{CI=N}, \P{CI}) (1_112_480)
\p{Case_Ignorable: Y*}  (Short: \p{CI=Y}, \p{CI}) (1632)
\p{Cased}               \p{Cased=Y} (3408)
\p{Cased: N*}           (Single: \P{Cased}) (1_110_704)
\p{Cased: Y*}           (Single: \p{Cased}) (3408)
\p{Cased_Letter}        \p{General_Category=Cased_Letter} (Short:
                          \p{LC}) (3207)
\p{Category: *}         \p{General_Category: *}
\p{Cc}                  \p{Cntrl} (= \p{General_Category=Control})
                          (65)
\p{Ccc: *}              \p{Canonical_Combining_Class: *}
\p{CE}                  \p{Composition_Exclusion} (=
                          \p{Composition_Exclusion=Y}) (81)
\p{CE: *}               \p{Composition_Exclusion: *}
\p{Cf}                  \p{Format} (= \p{General_Category=Format})
                          (140)
\p{Cham}                \p{Script=Cham} (NOT \p{Block=Cham}) (83)
\p{Changes_When_Casefolded} \p{Changes_When_Casefolded=Y} (Short:
                          \p{CWCF}) (1093)
```

```
    \p{Changes_When_Casefolded: N*} (Short: \p{CWCF=N}, \P{CWCF})
                                (1_113_019)
    \p{Changes_When_Casefolded: Y*} (Short: \p{CWCF=Y}, \p{CWCF})
                                (1093)
    \p{Changes_When_Casemapped} \p{Changes_When_Casemapped=Y} (Short:
                                \p{CWCM}) (2110)
    \p{Changes_When_Casemapped: N*} (Short: \p{CWCM=N}, \P{CWCM})
                                (1_112_002)
    \p{Changes_When_Casemapped: Y*} (Short: \p{CWCM=Y}, \p{CWCM})
                                (2110)
    \p{Changes_When_Lowercased} \p{Changes_When_Lowercased=Y} (Short:
                                \p{CWL}) (1029)
    \p{Changes_When_Lowercased: N*} (Short: \p{CWL=N}, \P{CWL})
                                (1_113_083)
    \p{Changes_When_Lowercased: Y*} (Short: \p{CWL=Y}, \p{CWL}) (1029)
    \p{Changes_When_NFKC_Casefolded} \p{Changes_When_NFKC_Casefolded=
                                Y} (Short: \p{CWKCF}) (9740)
    \p{Changes_When_NFKC_Casefolded: N*} (Short: \p{CWKCF=N},
                                \P{CWKCF}) (1_104_372)
    \p{Changes_When_NFKC_Casefolded: Y*} (Short: \p{CWKCF=Y},
                                \p{CWKCF}) (9740)
    \p{Changes_When_Titlecased} \p{Changes_When_Titlecased=Y} (Short:
                                \p{CWT}) (1085)
    \p{Changes_When_Titlecased: N*} (Short: \p{CWT=N}, \P{CWT})
                                (1_113_027)
    \p{Changes_When_Titlecased: Y*} (Short: \p{CWT=Y}, \p{CWT}) (1085)
    \p{Changes_When_Uppercased} \p{Changes_When_Uppercased=Y} (Short:
                                \p{CWU}) (1112)
    \p{Changes_When_Uppercased: N*} (Short: \p{CWU=N}, \P{CWU})
                                (1_113_000)
    \p{Changes_When_Uppercased: Y*} (Short: \p{CWU=Y}, \p{CWU}) (1112)
    \p{Cher}                \p{Cherokee} (= \p{Script=Cherokee}) (NOT
                              \p{Block=Cherokee}) (85)
    \p{Cherokee}            \p{Script=Cherokee} (Short: \p{Cher}; NOT
                              \p{Block=Cherokee}) (85)
    \p{CI}                  \p{Case_Ignorable} (= \p{Case_Ignorable=
                              Y}) (1632)
    \p{CI: *}               \p{Case_Ignorable: *}
  X \p{CJK_Compatibility}   \p{Block=CJK_Compatibility} (256)
  X \p{CJK_Compatibility_Forms} \p{Block=CJK_Compatibility_Forms} (32)
  X \p{CJK_Compatibility_Ideographs} \p{Block=
                                CJK_Compatibility_Ideographs} (512)
  X \p{CJK_Compatibility_Ideographs_Supplement} \p{Block=
                                CJK_Compatibility_Ideographs_Supplement}
                                (544)
  X \p{CJK_Radicals_Supplement} \p{Block=CJK_Radicals_Supplement} (128)
  X \p{CJK_Strokes}         \p{Block=CJK_Strokes} (48)
  X \p{CJK_Symbols_And_Punctuation} \p{Block=
                                CJK_Symbols_And_Punctuation} (64)
  X \p{CJK_Unified_Ideographs} \p{Block=CJK_Unified_Ideographs}
                                (20_992)
  X \p{CJK_Unified_Ideographs_Extension_A} \p{Block=
                                CJK_Unified_Ideographs_Extension_A}
                                (6592)
  X \p{CJK_Unified_Ideographs_Extension_B} \p{Block=
                                CJK_Unified_Ideographs_Extension_B}
```

```
                              (42_720)
X \p{CJK_Unified_Ideographs_Extension_C} \p{Block=
                              CJK_Unified_Ideographs_Extension_C}
                              (4160)
  \p{Close_Punctuation}   \p{General_Category=Close_Punctuation}
                              (Short: \p{Pe}) (71)
  \p{Cn}                  \p{Unassigned} (= \p{General_Category=
                              Unassigned}) (867_235)
  \p{Cntrl}               \p{General_Category=Control} Control
                              characters (Short: \p{Cc}) (65)
  \p{Co}                  \p{Private_Use} (= \p{General_Category=
                              Private_Use}) (NOT \p{Private_Use_Area})
                              (137_468)
X \p{Combining_Diacritical_Marks} \p{Block=
                              Combining_Diacritical_Marks} (112)
X \p{Combining_Diacritical_Marks_For_Symbols} \p{Block=
                              Combining_Diacritical_Marks_For_Symbols}
                              (Short: \p{InCombiningMarksForSymbols})
                              (48)
X \p{Combining_Diacritical_Marks_Supplement} \p{Block=
                              Combining_Diacritical_Marks_Supplement}
                              (64)
X \p{Combining_Half_Marks} \p{Block=Combining_Half_Marks} (16)
X \p{Combining_Marks_For_Symbols}
                              \p{Combining_Diacritical_Marks_For_-
                              Symbols} (= \p{Block=
                              Combining_Diacritical_Marks_For_-
                              Symbols}) (48)
  \p{Common}              \p{Script=Common} (Short: \p{Zyyy}) (5395)
X \p{Common_Indic_Number_Forms} \p{Block=Common_Indic_Number_Forms}
                              (16)
  \p{Comp_Ex}             \p{Full_Composition_Exclusion} (=
                              \p{Full_Composition_Exclusion=Y}) (1118)
  \p{Comp_Ex: *}          \p{Full_Composition_Exclusion: *}
  \p{Composition_Exclusion} \p{Composition_Exclusion=Y} (Short:
                              \p{CE}) (81)
  \p{Composition_Exclusion: N*} (Short: \p{CE=N}, \P{CE}) (1_114_031)
  \p{Composition_Exclusion: Y*} (Short: \p{CE=Y}, \p{CE}) (81)
  \p{Connector_Punctuation} \p{General_Category=
                              Connector_Punctuation} (Short: \p{Pc})
                              (10)
  \p{Control}             \p{Cntrl} (= \p{General_Category=Control})
                              (65)
X \p{Control_Pictures}    \p{Block=Control_Pictures} (64)
  \p{Copt}                \p{Coptic} (= \p{Script=Coptic}) (NOT
                              \p{Block=Coptic}) (135)
  \p{Coptic}              \p{Script=Coptic} (Short: \p{Copt}; NOT
                              \p{Block=Coptic}) (135)
X \p{Counting_Rod_Numerals} \p{Block=Counting_Rod_Numerals} (32)
  \p{Cprt}                \p{Cypriot} (= \p{Script=Cypriot}) (55)
  \p{Cs}                  \p{Surrogate} (= \p{General_Category=
                              Surrogate}) (2048)
  \p{Cuneiform}           \p{Script=Cuneiform} (Short: \p{Xsux}; NOT
                              \p{Block=Cuneiform}) (982)
X \p{Cuneiform_Numbers_And_Punctuation} \p{Block=
                              Cuneiform_Numbers_And_Punctuation} (128)
```

```
  \p{Currency_Symbol}     \p{General_Category=Currency_Symbol}
                            (Short: \p{Sc}) (46)
X \p{Currency_Symbols}    \p{Block=Currency_Symbols} (48)
  \p{CWCF}                \p{Changes_When_Casefolded} (=
                            \p{Changes_When_Casefolded=Y}) (1093)
  \p{CWCF: *}             \p{Changes_When_Casefolded: *}
  \p{CWCM}                \p{Changes_When_Casemapped} (=
                            \p{Changes_When_Casemapped=Y}) (2110)
  \p{CWCM: *}             \p{Changes_When_Casemapped: *}
  \p{CWKCF}               \p{Changes_When_NFKC_Casefolded} (=
                            \p{Changes_When_NFKC_Casefolded=Y})
                            (9740)
  \p{CWKCF: *}            \p{Changes_When_NFKC_Casefolded: *}
  \p{CWL}                 \p{Changes_When_Lowercased} (=
                            \p{Changes_When_Lowercased=Y}) (1029)
  \p{CWL: *}              \p{Changes_When_Lowercased: *}
  \p{CWT}                 \p{Changes_When_Titlecased} (=
                            \p{Changes_When_Titlecased=Y}) (1085)
  \p{CWT: *}              \p{Changes_When_Titlecased: *}
  \p{CWU}                 \p{Changes_When_Uppercased} (=
                            \p{Changes_When_Uppercased=Y}) (1112)
  \p{CWU: *}              \p{Changes_When_Uppercased: *}
  \p{Cypriot}             \p{Script=Cypriot} (Short: \p{Cprt}) (55)
X \p{Cypriot_Syllabary}   \p{Block=Cypriot_Syllabary} (64)
  \p{Cyrillic}            \p{Script=Cyrillic} (Short: \p{Cyrl}; NOT
                            \p{Block=Cyrillic}) (404)
X \p{Cyrillic_Extended_A} \p{Block=Cyrillic_Extended_A} (32)
X \p{Cyrillic_Extended_B} \p{Block=Cyrillic_Extended_B} (96)
X \p{Cyrillic_Supplement} \p{Block=Cyrillic_Supplement} (48)
X \p{Cyrillic_Supplementary} \p{Cyrillic_Supplement} (= \p{Block=
                           Cyrillic_Supplement}) (48)
  \p{Cyrl}                \p{Cyrillic} (= \p{Script=Cyrillic}) (NOT
                            \p{Block=Cyrillic}) (404)
  \p{Dash}                \p{Dash=Y} (25)
  \p{Dash: N*}            (Single: \P{Dash}) (1_114_087)
  \p{Dash: Y*}            (Single: \p{Dash}) (25)
  \p{Dash_Punctuation}    \p{General_Category=Dash_Punctuation}
                            (Short: \p{Pd}) (21)
  \p{Decimal_Number}      \p{Digit} (= \p{General_Category=
                            Decimal_Number}) (411)
  \p{Decomposition_Type: Can} \p{Decomposition_Type=Canonical}
                            (13_221)
  \p{Decomposition_Type: Canonical} (Short: \p{Dt=Can}) (13_221)
  \p{Decomposition_Type: Circle} (Short: \p{Dt=Enc}) (238)
  \p{Decomposition_Type: Com} \p{Decomposition_Type=Compat} (720)
  \p{Decomposition_Type: Compat} (Short: \p{Dt=Com}) (720)
  \p{Decomposition_Type: Enc} \p{Decomposition_Type=Circle} (238)
  \p{Decomposition_Type: Fin} \p{Decomposition_Type=Final} (240)
  \p{Decomposition_Type: Final} (Short: \p{Dt=Fin}) (240)
  \p{Decomposition_Type: Font} (Short: \p{Dt=Font}) (1043)
  \p{Decomposition_Type: Fra} \p{Decomposition_Type=Fraction} (20)
  \p{Decomposition_Type: Fraction} (Short: \p{Dt=Fra}) (20)
  \p{Decomposition_Type: Init} \p{Decomposition_Type=Initial} (171)
  \p{Decomposition_Type: Initial} (Short: \p{Dt=Init}) (171)
  \p{Decomposition_Type: Iso} \p{Decomposition_Type=Isolated} (238)
  \p{Decomposition_Type: Isolated} (Short: \p{Dt=Iso}) (238)
```

```
    \p{Decomposition_Type: Med} \p{Decomposition_Type=Medial} (82)
    \p{Decomposition_Type: Medial} (Short: \p{Dt=Med}) (82)
    \p{Decomposition_Type: Nar} \p{Decomposition_Type=Narrow} (122)
    \p{Decomposition_Type: Narrow} (Short: \p{Dt=Nar}) (122)
    \p{Decomposition_Type: Nb} \p{Decomposition_Type=Nobreak} (5)
    \p{Decomposition_Type: Nobreak} (Short: \p{Dt=Nb}) (5)
    \p{Decomposition_Type: Non_Canon} \p{Decomposition_Type=
                          Non_Canonical} (Perl extension) (3467)
    \p{Decomposition_Type: Non_Canonical} Union of all non-canonical
                          decompositions (Short: \p{Dt=NonCanon})
                          (Perl extension) (3467)
    \p{Decomposition_Type: None} (Short: \p{Dt=None}) (1_097_424)
    \p{Decomposition_Type: Small} (Short: \p{Dt=Sml}) (26)
    \p{Decomposition_Type: Sml} \p{Decomposition_Type=Small} (26)
    \p{Decomposition_Type: Sqr} \p{Decomposition_Type=Square} (251)
    \p{Decomposition_Type: Square} (Short: \p{Dt=Sqr}) (251)
    \p{Decomposition_Type: Sub} (Short: \p{Dt=Sub}) (30)
    \p{Decomposition_Type: Sup} \p{Decomposition_Type=Super} (142)
    \p{Decomposition_Type: Super} (Short: \p{Dt=Sup}) (142)
    \p{Decomposition_Type: Vert} \p{Decomposition_Type=Vertical} (35)
    \p{Decomposition_Type: Vertical} (Short: \p{Dt=Vert}) (35)
    \p{Decomposition_Type: Wide} (Short: \p{Dt=Wide}) (104)
    \p{Default_Ignorable_Code_Point} \p{Default_Ignorable_Code_Point=
                          Y} (Short: \p{DI}) (4167)
    \p{Default_Ignorable_Code_Point: N*} (Short: \p{DI=N}, \P{DI})
                          (1_109_945)
    \p{Default_Ignorable_Code_Point: Y*} (Short: \p{DI=Y}, \p{DI})
                          (4167)
    \p{Dep}               \p{Deprecated} (= \p{Deprecated=Y}) (110)
    \p{Dep: *}            \p{Deprecated: *}
    \p{Deprecated}        \p{Deprecated=Y} (Short: \p{Dep}) (110)
    \p{Deprecated: N*}    (Short: \p{Dep=N}, \P{Dep}) (1_114_002)
    \p{Deprecated: Y*}    (Short: \p{Dep=Y}, \p{Dep}) (110)
    \p{Deseret}           \p{Script=Deseret} (Short: \p{Dsrt}) (80)
    \p{Deva}              \p{Devanagari} (= \p{Script=Devanagari})
                          (NOT \p{Block=Devanagari}) (140)
    \p{Devanagari}        \p{Script=Devanagari} (Short: \p{Deva};
                          NOT \p{Block=Devanagari}) (140)
X   \p{Devanagari_Extended} \p{Block=Devanagari_Extended} (32)
    \p{DI}                \p{Default_Ignorable_Code_Point} (=
                          \p{Default_Ignorable_Code_Point=Y})
                          (4167)
    \p{DI: *}             \p{Default_Ignorable_Code_Point: *}
    \p{Dia}               \p{Diacritic} (= \p{Diacritic=Y}) (639)
    \p{Dia: *}            \p{Diacritic: *}
    \p{Diacritic}         \p{Diacritic=Y} (Short: \p{Dia}) (639)
    \p{Diacritic: N*}     (Short: \p{Dia=N}, \P{Dia}) (1_113_473)
    \p{Diacritic: Y*}     (Short: \p{Dia=Y}, \p{Dia}) (639)
    \p{Digit}             \p{General_Category=Decimal_Number} \d,
                          extended beyond just [0-9] (Short:
                          \p{Nd}) (411)
X   \p{Dingbats}          \p{Block=Dingbats} (192)
X   \p{Domino_Tiles}      \p{Block=Domino_Tiles} (112)
    \p{Dsrt}              \p{Deseret} (= \p{Script=Deseret}) (80)
    \p{Dt: *}             \p{Decomposition_Type: *}
    \p{Ea: *}             \p{East_Asian_Width: *}
```

```
    \p{East_Asian_Width: A} \p{East_Asian_Width=Ambiguous} (138_666)
    \p{East_Asian_Width: Ambiguous} (Short: \p{Ea=A}) (138_666)
    \p{East_Asian_Width: F} \p{East_Asian_Width=Fullwidth} (104)
    \p{East_Asian_Width: Fullwidth} (Short: \p{Ea=F}) (104)
    \p{East_Asian_Width: H} \p{East_Asian_Width=Halfwidth} (123)
    \p{East_Asian_Width: Halfwidth} (Short: \p{Ea=H}) (123)
    \p{East_Asian_Width: N} \p{East_Asian_Width=Neutral} (801_909)
    \p{East_Asian_Width: Na} \p{East_Asian_Width=Narrow} (111)
    \p{East_Asian_Width: Narrow} (Short: \p{Ea=Na}) (111)
    \p{East_Asian_Width: Neutral} (Short: \p{Ea=N}) (801_909)
    \p{East_Asian_Width: W} \p{East_Asian_Width=Wide} (173_199)
    \p{East_Asian_Width: Wide} (Short: \p{Ea=W}) (173_199)
    \p{Egyp}                \p{Egyptian_Hieroglyphs} (= \p{Script=
                              Egyptian_Hieroglyphs}) (NOT \p{Block=
                              Egyptian_Hieroglyphs}) (1071)
    \p{Egyptian_Hieroglyphs} \p{Script=Egyptian_Hieroglyphs} (Short:
                              \p{Egyp}; NOT \p{Block=
                              Egyptian_Hieroglyphs}) (1071)
  X \p{Enclosed_Alphanumeric_Supplement} \p{Block=
                              Enclosed_Alphanumeric_Supplement} (256)
  X \p{Enclosed_Alphanumerics} \p{Block=Enclosed_Alphanumerics} (160)
  X \p{Enclosed_CJK_Letters_And_Months} \p{Block=
                              Enclosed_CJK_Letters_And_Months} (256)
  X \p{Enclosed_Ideographic_Supplement} \p{Block=
                              Enclosed_Ideographic_Supplement} (256)
    \p{Enclosing_Mark}      \p{General_Category=Enclosing_Mark}
                              (Short: \p{Me}) (13)
    \p{Ethi}                \p{Ethiopic} (= \p{Script=Ethiopic}) (NOT
                              \p{Block=Ethiopic}) (461)
    \p{Ethiopic}            \p{Script=Ethiopic} (Short: \p{Ethi}; NOT
                              \p{Block=Ethiopic}) (461)
  X \p{Ethiopic_Extended}   \p{Block=Ethiopic_Extended} (96)
  X \p{Ethiopic_Supplement} \p{Block=Ethiopic_Supplement} (32)
    \p{Ext}                 \p{Extender} (= \p{Extender=Y}) (28)
    \p{Ext: *}              \p{Extender: *}
    \p{Extender}            \p{Extender=Y} (Short: \p{Ext}) (28)
    \p{Extender: N*}        (Short: \p{Ext=N}, \P{Ext}) (1_114_084)
    \p{Extender: Y*}        (Short: \p{Ext=Y}, \p{Ext}) (28)
    \p{Final_Punctuation}   \p{General_Category=Final_Punctuation}
                              (Short: \p{Pf}) (10)
    \p{Format}              \p{General_Category=Format} (Short:
                              \p{Cf}) (140)
    \p{Full_Composition_Exclusion} \p{Full_Composition_Exclusion=Y}
                              (Short: \p{CompEx}) (1118)
    \p{Full_Composition_Exclusion: N*} (Short: \p{CompEx=N},
                              \P{CompEx}) (1_112_994)
    \p{Full_Composition_Exclusion: Y*} (Short: \p{CompEx=Y},
                              \p{CompEx}) (1118)
    \p{Gc: *}               \p{General_Category: *}
    \p{GCB: *}              \p{Grapheme_Cluster_Break: *}
    \p{General_Category: C} \p{General_Category=Other} (1_006_956)
    \p{General_Category: Cased_Letter} [\p{Ll}\p{Lu}\p{Lt}] (Short:
                              \p{Gc=LC}, \p{LC}) (3207)
    \p{General_Category: Cc} \p{General_Category=Control} (65)
    \p{General_Category: Cf} \p{General_Category=Format} (140)
    \p{General_Category: Close_Punctuation} (Short: \p{Gc=Pe}, \p{Pe})
```

```
                                (71)
    \p{General_Category: Cn} \p{General_Category=Unassigned} (867_235)
    \p{General_Category: Cntrl} \p{General_Category=Control} (65)
    \p{General_Category: Co} \p{General_Category=Private_Use} (137_468)
    \p{General_Category: Connector_Punctuation} (Short: \p{Gc=Pc},
                                \p{Pc}) (10)
    \p{General_Category: Control} (Short: \p{Gc=Cc}, \p{Cc}) (65)
    \p{General_Category: Cs} \p{General_Category=Surrogate} (2048)
    \p{General_Category: Currency_Symbol} (Short: \p{Gc=Sc}, \p{Sc})
                                (46)
    \p{General_Category: Dash_Punctuation} (Short: \p{Gc=Pd}, \p{Pd})
                                (21)
    \p{General_Category: Decimal_Number} (Short: \p{Gc=Nd}, \p{Nd})
                                (411)
    \p{General_Category: Digit} \p{General_Category=Decimal_Number}
                                (411)
    \p{General_Category: Enclosing_Mark} (Short: \p{Gc=Me}, \p{Me})
                                (13)
    \p{General_Category: Final_Punctuation} (Short: \p{Gc=Pf}, \p{Pf})
                                (10)
    \p{General_Category: Format} (Short: \p{Gc=Cf}, \p{Cf}) (140)
    \p{General_Category: Initial_Punctuation} (Short: \p{Gc=Pi},
                                \p{Pi}) (12)
    \p{General_Category: L} \p{General_Category=Letter} (99_537)
  X \p{General_Category: L&} \p{General_Category=Cased_Letter} (3207)
  X \p{General_Category: L_} \p{General_Category=Cased_Letter} (3207)
    \p{General_Category: LC} \p{General_Category=Cased_Letter} (3207)
    \p{General_Category: Letter} (Short: \p{Gc=L}, \p{L}) (99_537)
    \p{General_Category: Letter_Number} (Short: \p{Gc=Nl}, \p{Nl})
                                (224)
    \p{General_Category: Line_Separator} (Short: \p{Gc=Zl}, \p{Zl}) (1)
    \p{General_Category: Ll} \p{General_Category=Lowercase_Letter}
                                (1749)
    \p{General_Category: Lm} \p{General_Category=Modifier_Letter} (202)
    \p{General_Category: Lo} \p{General_Category=Other_Letter} (96_128)
    \p{General_Category: Lowercase_Letter} (Short: \p{Gc=Ll}, \p{Ll})
                                (1749)
    \p{General_Category: Lt} \p{General_Category=Titlecase_Letter} (31)
    \p{General_Category: Lu} \p{General_Category=Uppercase_Letter}
                                (1427)
    \p{General_Category: M} \p{General_Category=Mark} (1451)
    \p{General_Category: Mark} (Short: \p{Gc=M}, \p{M}) (1451)
    \p{General_Category: Math_Symbol} (Short: \p{Gc=Sm}, \p{Sm}) (945)
    \p{General_Category: Mc} \p{General_Category=Spacing_Mark} (276)
    \p{General_Category: Me} \p{General_Category=Enclosing_Mark} (13)
    \p{General_Category: Mn} \p{General_Category=Nonspacing_Mark}
                                (1162)
    \p{General_Category: Modifier_Letter} (Short: \p{Gc=Lm}, \p{Lm})
                                (202)
    \p{General_Category: Modifier_Symbol} (Short: \p{Gc=Sk}, \p{Sk})
                                (99)
    \p{General_Category: N} \p{General_Category=Number} (1064)
    \p{General_Category: Nd} \p{General_Category=Decimal_Number} (411)
    \p{General_Category: Nl} \p{General_Category=Letter_Number} (224)
    \p{General_Category: No} \p{General_Category=Other_Number} (429)
    \p{General_Category: Nonspacing_Mark} (Short: \p{Gc=Mn}, \p{Mn})
```

```
                              (1162)
    \p{General_Category: Number} (Short: \p{Gc=N}, \p{N}) (1064)
    \p{General_Category: Open_Punctuation} (Short: \p{Gc=Ps}, \p{Ps})
                              (72)
    \p{General_Category: Other} (Short: \p{Gc=C}, \p{C}) (1_006_956)
    \p{General_Category: Other_Letter} (Short: \p{Gc=Lo}, \p{Lo})
                              (96_128)
    \p{General_Category: Other_Number} (Short: \p{Gc=No}, \p{No}) (429)
    \p{General_Category: Other_Punctuation} (Short: \p{Gc=Po}, \p{Po})
                              (389)
    \p{General_Category: Other_Symbol} (Short: \p{Gc=So}, \p{So})
                              (3409)
    \p{General_Category: P} \p{General_Category=Punctuation} (585)
    \p{General_Category: Paragraph_Separator} (Short: \p{Gc=Zp},
                              \p{Zp}) (1)
    \p{General_Category: Pc} \p{General_Category=
                              Connector_Punctuation} (10)
    \p{General_Category: Pd} \p{General_Category=Dash_Punctuation} (21)
    \p{General_Category: Pe} \p{General_Category=Close_Punctuation}
                              (71)
    \p{General_Category: Pf} \p{General_Category=Final_Punctuation}
                              (10)
    \p{General_Category: Pi} \p{General_Category=Initial_Punctuation}
                              (12)
    \p{General_Category: Po} \p{General_Category=Other_Punctuation}
                              (389)
    \p{General_Category: Private_Use} (Short: \p{Gc=Co}, \p{Co})
                              (137_468)
    \p{General_Category: Ps} \p{General_Category=Open_Punctuation} (72)
    \p{General_Category: Punct} \p{General_Category=Punctuation} (585)
    \p{General_Category: Punctuation} (Short: \p{Gc=P}, \p{P}) (585)
    \p{General_Category: S} \p{General_Category=Symbol} (4499)
    \p{General_Category: Sc} \p{General_Category=Currency_Symbol} (46)
    \p{General_Category: Separator} (Short: \p{Gc=Z}, \p{Z}) (20)
    \p{General_Category: Sk} \p{General_Category=Modifier_Symbol} (99)
    \p{General_Category: Sm} \p{General_Category=Math_Symbol} (945)
    \p{General_Category: So} \p{General_Category=Other_Symbol} (3409)
    \p{General_Category: Space_Separator} (Short: \p{Gc=Zs}, \p{Zs})
                              (18)
    \p{General_Category: Spacing_Mark} (Short: \p{Gc=Mc}, \p{Mc}) (276)
    \p{General_Category: Surrogate} Mostly not usable in Perl. (Short:
                              \p{Gc=Cs}, \p{Cs}) (2048)
    \p{General_Category: Symbol} (Short: \p{Gc=S}, \p{S}) (4499)
    \p{General_Category: Titlecase_Letter} (Short: \p{Gc=Lt}, \p{Lt})
                              (31)
    \p{General_Category: Unassigned} (Short: \p{Gc=Cn}, \p{Cn})
                              (867_235)
    \p{General_Category: Uppercase_Letter} (Short: \p{Gc=Lu}, \p{Lu})
                              (1427)
    \p{General_Category: Z} \p{General_Category=Separator} (20)
    \p{General_Category: Zl} \p{General_Category=Line_Separator} (1)
    \p{General_Category: Zp} \p{General_Category=Paragraph_Separator}
                              (1)
    \p{General_Category: Zs} \p{General_Category=Space_Separator} (18)
  X \p{General_Punctuation} \p{Block=General_Punctuation} (112)
  X \p{Geometric_Shapes}    \p{Block=Geometric_Shapes} (96)
```

```
  \p{Geor}                \p{Georgian} (= \p{Script=Georgian}) (NOT
                            \p{Block=Georgian}) (120)
  \p{Georgian}            \p{Script=Georgian} (Short: \p{Geor}; NOT
                            \p{Block=Georgian}) (120)
X \p{Georgian_Supplement} \p{Block=Georgian_Supplement} (48)
  \p{Glag}                \p{Glagolitic} (= \p{Script=Glagolitic})
                            (NOT \p{Block=Glagolitic}) (94)
  \p{Glagolitic}          \p{Script=Glagolitic} (Short: \p{Glag};
                            NOT \p{Block=Glagolitic}) (94)
  \p{Goth}                \p{Gothic} (= \p{Script=Gothic}) (NOT
                            \p{Block=Gothic}) (27)
  \p{Gothic}              \p{Script=Gothic} (Short: \p{Goth}; NOT
                            \p{Block=Gothic}) (27)
  \p{Gr_Base}             \p{Grapheme_Base} (= \p{Grapheme_Base=Y})
                            (105_958)
  \p{Gr_Base: *}          \p{Grapheme_Base: *}
  \p{Gr_Ext}              \p{Grapheme_Extend} (= \p{Grapheme_Extend=
                            Y}) (1198)
  \p{Gr_Ext: *}           \p{Grapheme_Extend: *}
  \p{Graph}               Characters that are graphical (244_744)
  \p{Grapheme_Base}       \p{Grapheme_Base=Y} (Short: \p{GrBase})
                            (105_958)
  \p{Grapheme_Base: N*}   (Short: \p{GrBase=N}, \P{GrBase})
                            (1_008_154)
  \p{Grapheme_Base: Y*}   (Short: \p{GrBase=Y}, \p{GrBase}) (105_958)
  \p{Grapheme_Cluster_Break: CN} \p{Grapheme_Cluster_Break=Control}
                            (203)
  \p{Grapheme_Cluster_Break: Control} (Short: \p{GCB=CN}) (203)
  \p{Grapheme_Cluster_Break: CR} (Short: \p{GCB=CR}) (1)
  \p{Grapheme_Cluster_Break: EX} \p{Grapheme_Cluster_Break=Extend}
                            (1205)
  \p{Grapheme_Cluster_Break: Extend} (Short: \p{GCB=EX}) (1205)
  \p{Grapheme_Cluster_Break: L} (Short: \p{GCB=L}) (125)
  \p{Grapheme_Cluster_Break: LF} (Short: \p{GCB=LF}) (1)
  \p{Grapheme_Cluster_Break: LV} (Short: \p{GCB=LV}) (399)
  \p{Grapheme_Cluster_Break: LVT} (Short: \p{GCB=LVT}) (10_773)
  \p{Grapheme_Cluster_Break: Other} (Short: \p{GCB=XX}) (1_100_901)
  \p{Grapheme_Cluster_Break: PP} \p{Grapheme_Cluster_Break=Prepend}
                            (15)
  \p{Grapheme_Cluster_Break: Prepend} (Short: \p{GCB=PP}) (15)
  \p{Grapheme_Cluster_Break: SM} \p{Grapheme_Cluster_Break=
                           SpacingMark} (257)
  \p{Grapheme_Cluster_Break: SpacingMark} (Short: \p{GCB=SM}) (257)
  \p{Grapheme_Cluster_Break: T} (Short: \p{GCB=T}) (137)
  \p{Grapheme_Cluster_Break: V} (Short: \p{GCB=V}) (95)
  \p{Grapheme_Cluster_Break: XX} \p{Grapheme_Cluster_Break=Other}
                            (1_100_901)
  \p{Grapheme_Extend}     \p{Grapheme_Extend=Y} (Short: \p{GrExt})
                            (1198)
  \p{Grapheme_Extend: N*} (Short: \p{GrExt=N}, \P{GrExt}) (1_112_914)
  \p{Grapheme_Extend: Y*} (Short: \p{GrExt=Y}, \p{GrExt}) (1198)
  \p{Greek}               \p{Script=Greek} (Short: \p{Grek}; NOT
                            \p{Greek_And_Coptic}) (511)
X \p{Greek_And_Coptic}    \p{Block=Greek_And_Coptic} (Short:
                            \p{InGreek}) (144)
X \p{Greek_Extended}      \p{Block=Greek_Extended} (256)
```

```
    \p{Grek}                \p{Greek} (= \p{Script=Greek}) (NOT
                              \p{Greek_And_Coptic}) (511)
    \p{Gujarati}            \p{Script=Gujarati} (Short: \p{Gujr}; NOT
                              \p{Block=Gujarati}) (83)
    \p{Gujr}                \p{Gujarati} (= \p{Script=Gujarati}) (NOT
                              \p{Block=Gujarati}) (83)
    \p{Gurmukhi}            \p{Script=Gurmukhi} (Short: \p{Guru}; NOT
                              \p{Block=Gurmukhi}) (79)
    \p{Guru}                \p{Gurmukhi} (= \p{Script=Gurmukhi}) (NOT
                              \p{Block=Gurmukhi}) (79)
  X \p{Halfwidth_And_Fullwidth_Forms} \p{Block=
                             Halfwidth_And_Fullwidth_Forms} (240)
    \p{Han}                 \p{Script=Han} (75_738)
    \p{Hang}                \p{Hangul} (= \p{Script=Hangul}) (11_737)
    \p{Hangul}              \p{Script=Hangul} (Short: \p{Hang})
                              (11_737)
  X \p{Hangul_Compatibility_Jamo} \p{Block=Hangul_Compatibility_Jamo}
                              (96)
  X \p{Hangul_Jamo}         \p{Block=Hangul_Jamo} (256)
  X \p{Hangul_Jamo_Extended_A} \p{Block=Hangul_Jamo_Extended_A} (32)
  X \p{Hangul_Jamo_Extended_B} \p{Block=Hangul_Jamo_Extended_B} (80)
    \p{Hangul_Syllable_Type: L} \p{Hangul_Syllable_Type=Leading_Jamo}
                              (125)
    \p{Hangul_Syllable_Type: Leading_Jamo} (Short: \p{Hst=L}) (125)
    \p{Hangul_Syllable_Type: LV} \p{Hangul_Syllable_Type=LV_Syllable}
                              (399)
    \p{Hangul_Syllable_Type: LV_Syllable} (Short: \p{Hst=LV}) (399)
    \p{Hangul_Syllable_Type: LVT} \p{Hangul_Syllable_Type=
                              LVT_Syllable} (10_773)
    \p{Hangul_Syllable_Type: LVT_Syllable} (Short: \p{Hst=LVT})
                              (10_773)
    \p{Hangul_Syllable_Type: NA} \p{Hangul_Syllable_Type=
                              Not_Applicable} (1_102_583)
    \p{Hangul_Syllable_Type: Not_Applicable} (Short: \p{Hst=NA})
                              (1_102_583)
    \p{Hangul_Syllable_Type: T} \p{Hangul_Syllable_Type=Trailing_Jamo}
                              (137)
    \p{Hangul_Syllable_Type: Trailing_Jamo} (Short: \p{Hst=T}) (137)
    \p{Hangul_Syllable_Type: V} \p{Hangul_Syllable_Type=Vowel_Jamo}
                              (95)
    \p{Hangul_Syllable_Type: Vowel_Jamo} (Short: \p{Hst=V}) (95)
  X \p{Hangul_Syllables}    \p{Block=Hangul_Syllables} (11_184)
    \p{Hani}                \p{Han} (= \p{Script=Han}) (75_738)
    \p{Hano}                \p{Hanunoo} (= \p{Script=Hanunoo}) (NOT
                              \p{Block=Hanunoo}) (21)
    \p{Hanunoo}             \p{Script=Hanunoo} (Short: \p{Hano}; NOT
                              \p{Block=Hanunoo}) (21)
    \p{Hebr}                \p{Hebrew} (= \p{Script=Hebrew}) (NOT
                              \p{Block=Hebrew}) (133)
    \p{Hebrew}              \p{Script=Hebrew} (Short: \p{Hebr}; NOT
                              \p{Block=Hebrew}) (133)
    \p{Hex}                 \p{XDigit} (= \p{Hex_Digit=Y}) (44)
    \p{Hex: *}              \p{Hex_Digit: *}
    \p{Hex_Digit}           \p{XDigit} (= \p{Hex_Digit=Y}) (44)
    \p{Hex_Digit: N*}       (Short: \p{Hex=N}, \P{Hex}) (1_114_068)
    \p{Hex_Digit: Y*}       (Short: \p{Hex=Y}, \p{Hex}) (44)
```

```
X \p{High_Private_Use_Surrogates} \p{Block=
                          High_Private_Use_Surrogates} (128)
X \p{High_Surrogates}     \p{Block=High_Surrogates} (896)
  \p{Hira}                \p{Hiragana} (= \p{Script=Hiragana}) (NOT
                          \p{Block=Hiragana}) (90)
  \p{Hiragana}            \p{Script=Hiragana} (Short: \p{Hira}; NOT
                          \p{Block=Hiragana}) (90)
  \p{HorizSpace}          \p{Blank} (19)
  \p{Hst: *}              \p{Hangul_Syllable_Type: *}
S \p{Hyphen}              \p{Hyphen=Y} (11)
S \p{Hyphen: N*}          Use the Line_Break property instead; see
                          www.unicode.org/reports/tr14 (Single:
                          \P{Hyphen}) (1_114_101)
S \p{Hyphen: Y*}          Use the Line_Break property instead; see
                          www.unicode.org/reports/tr14 (Single:
                          \p{Hyphen}) (11)
  \p{ID_Continue}         \p{ID_Continue=Y} (Short: \p{IDC})
                          (101_634)
  \p{ID_Continue: N*}     (Short: \p{IDC=N}, \P{IDC}) (1_012_478)
  \p{ID_Continue: Y*}     (Short: \p{IDC=Y}, \p{IDC}) (101_634)
  \p{ID_Start}            \p{ID_Start=Y} (Short: \p{IDS}) (99_764)
  \p{ID_Start: N*}        (Short: \p{IDS=N}, \P{IDS}) (1_014_348)
  \p{ID_Start: Y*}        (Short: \p{IDS=Y}, \p{IDS}) (99_764)
  \p{IDC}                 \p{ID_Continue} (= \p{ID_Continue=Y})
                          (101_634)
  \p{IDC: *}              \p{ID_Continue: *}
  \p{Ideo}                \p{Ideographic} (= \p{Ideographic=Y})
                          (75_408)
  \p{Ideo: *}             \p{Ideographic: *}
  \p{Ideographic}         \p{Ideographic=Y} (Short: \p{Ideo})
                          (75_408)
  \p{Ideographic: N*}     (Short: \p{Ideo=N}, \P{Ideo}) (1_038_704)
  \p{Ideographic: Y*}     (Short: \p{Ideo=Y}, \p{Ideo}) (75_408)
X \p{Ideographic_Description_Characters} \p{Block=
                          Ideographic_Description_Characters} (16)
  \p{IDS}                 \p{ID_Start} (= \p{ID_Start=Y}) (99_764)
  \p{IDS: *}              \p{ID_Start: *}
  \p{IDS_Binary_Operator} \p{IDS_Binary_Operator=Y} (Short:
                          \p{IDSB}) (10)
  \p{IDS_Binary_Operator: N*} (Short: \p{IDSB=N}, \P{IDSB})
                          (1_114_102)
  \p{IDS_Binary_Operator: Y*} (Short: \p{IDSB=Y}, \p{IDSB}) (10)
  \p{IDS_Trinary_Operator} \p{IDS_Trinary_Operator=Y} (Short:
                          \p{IDST}) (2)
  \p{IDS_Trinary_Operator: N*} (Short: \p{IDST=N}, \P{IDST})
                          (1_114_110)
  \p{IDS_Trinary_Operator: Y*} (Short: \p{IDST=Y}, \p{IDST}) (2)
  \p{IDSB}                \p{IDS_Binary_Operator} (=
                          \p{IDS_Binary_Operator=Y}) (10)
  \p{IDSB: *}             \p{IDS_Binary_Operator: *}
  \p{IDST}                \p{IDS_Trinary_Operator} (=
                          \p{IDS_Trinary_Operator=Y}) (2)
  \p{IDST: *}             \p{IDS_Trinary_Operator: *}
  \p{Imperial_Aramaic}    \p{Script=Imperial_Aramaic} (Short:
                          \p{Armi}; NOT \p{Block=
                          Imperial_Aramaic}) (31)
```

```
    \p{In: *}               \p{Present_In: *} (Perl extension)
    \p{In_*}                \p{Block: *}
    \p{Inherited}           \p{Script=Inherited} (Short: \p{Zinh})
                              (523)
    \p{Initial_Punctuation} \p{General_Category=Initial_Punctuation}
                              (Short: \p{Pi}) (12)
    \p{Inscriptional_Pahlavi} \p{Script=Inscriptional_Pahlavi} (Short:
                              \p{Phli}; NOT \p{Block=
                              Inscriptional_Pahlavi}) (27)
    \p{Inscriptional_Parthian} \p{Script=Inscriptional_Parthian}
                              (Short: \p{Prti}; NOT \p{Block=
                              Inscriptional_Parthian}) (30)
  X \p{IPA_Extensions}      \p{Block=IPA_Extensions} (96)
    \p{Is_*}                \p{*} (Any exceptions are individually
                              noted beginning with the word NOT.) If
                              an entry has flag(s) at its beginning,
                              like 'D', the 'Is_' form has the same
                              flag(s)
    \p{Ital}                \p{Old_Italic} (= \p{Script=Old_Italic})
                              (NOT \p{Block=Old_Italic}) (35)
    \p{Java}                \p{Javanese} (= \p{Script=Javanese}) (NOT
                              \p{Block=Javanese}) (91)
    \p{Javanese}            \p{Script=Javanese} (Short: \p{Java}; NOT
                              \p{Block=Javanese}) (91)
    \p{Jg: *}               \p{Joining_Group: *}
    \p{Join_C}              \p{Join_Control} (= \p{Join_Control=Y}) (2)
    \p{Join_C: *}           \p{Join_Control: *}
    \p{Join_Control}        \p{Join_Control=Y} (Short: \p{JoinC}) (2)
    \p{Join_Control: N*}    (Short: \p{JoinC=N}, \P{JoinC}) (1_114_110)
    \p{Join_Control: Y*}    (Short: \p{JoinC=Y}, \p{JoinC}) (2)
    \p{Joining_Group: Ain}  (Short: \p{Jg=Ain}) (7)
    \p{Joining_Group: Alaph} (Short: \p{Jg=Alaph}) (1)
    \p{Joining_Group: Alef} (Short: \p{Jg=Alef}) (10)
    \p{Joining_Group: Beh}  (Short: \p{Jg=Beh}) (19)
    \p{Joining_Group: Beth} (Short: \p{Jg=Beth}) (2)
    \p{Joining_Group: Burushaski_Yeh_Barree} (Short: \p{Jg=
                              BurushaskiYehBarree}) (2)
    \p{Joining_Group: Dal}  (Short: \p{Jg=Dal}) (14)
    \p{Joining_Group: Dalath_Rish} (Short: \p{Jg=DalathRish}) (4)
    \p{Joining_Group: E}    (Short: \p{Jg=E}) (1)
    \p{Joining_Group: Farsi_Yeh} (Short: \p{Jg=FarsiYeh}) (7)
    \p{Joining_Group: Fe}   (Short: \p{Jg=Fe}) (1)
    \p{Joining_Group: Feh}  (Short: \p{Jg=Feh}) (9)
    \p{Joining_Group: Final_Semkath} (Short: \p{Jg=FinalSemkath}) (1)
    \p{Joining_Group: Gaf}  (Short: \p{Jg=Gaf}) (13)
    \p{Joining_Group: Gamal} (Short: \p{Jg=Gamal}) (3)
    \p{Joining_Group: Hah}  (Short: \p{Jg=Hah}) (17)
    \p{Joining_Group: Hamza_On_Heh_Goal} (Short: \p{Jg=
                              HamzaOnHehGoal}) (1)
    \p{Joining_Group: He}   (Short: \p{Jg=He}) (1)
    \p{Joining_Group: Heh}  (Short: \p{Jg=Heh}) (1)
    \p{Joining_Group: Heh_Goal} (Short: \p{Jg=HehGoal}) (2)
    \p{Joining_Group: Heth} (Short: \p{Jg=Heth}) (1)
    \p{Joining_Group: Kaf}  (Short: \p{Jg=Kaf}) (5)
    \p{Joining_Group: Kaph} (Short: \p{Jg=Kaph}) (1)
    \p{Joining_Group: Khaph} (Short: \p{Jg=Khaph}) (1)
```

```
  \p{Joining_Group: Knotted_Heh} (Short: \p{Jg=KnottedHeh}) (2)
  \p{Joining_Group: Lam}  (Short: \p{Jg=Lam}) (6)
  \p{Joining_Group: Lamadh} (Short: \p{Jg=Lamadh}) (1)
  \p{Joining_Group: Meem} (Short: \p{Jg=Meem}) (3)
  \p{Joining_Group: Mim}  (Short: \p{Jg=Mim}) (1)
  \p{Joining_Group: No_Joining_Group} (Short: \p{Jg=NoJoiningGroup})
                          (1_113_883)
  \p{Joining_Group: Noon} (Short: \p{Jg=Noon}) (8)
  \p{Joining_Group: Nun}  (Short: \p{Jg=Nun}) (1)
  \p{Joining_Group: Nya}  (Short: \p{Jg=Nya}) (1)
  \p{Joining_Group: Pe}   (Short: \p{Jg=Pe}) (1)
  \p{Joining_Group: Qaf}  (Short: \p{Jg=Qaf}) (4)
  \p{Joining_Group: Qaph} (Short: \p{Jg=Qaph}) (1)
  \p{Joining_Group: Reh}  (Short: \p{Jg=Reh}) (16)
  \p{Joining_Group: Reversed_Pe} (Short: \p{Jg=ReversedPe}) (1)
  \p{Joining_Group: Sad}  (Short: \p{Jg=Sad}) (5)
  \p{Joining_Group: Sadhe} (Short: \p{Jg=Sadhe}) (1)
  \p{Joining_Group: Seen} (Short: \p{Jg=Seen}) (11)
  \p{Joining_Group: Semkath} (Short: \p{Jg=Semkath}) (1)
  \p{Joining_Group: Shin} (Short: \p{Jg=Shin}) (1)
  \p{Joining_Group: Swash_Kaf} (Short: \p{Jg=SwashKaf}) (1)
  \p{Joining_Group: Syriac_Waw} (Short: \p{Jg=SyriacWaw}) (1)
  \p{Joining_Group: Tah}  (Short: \p{Jg=Tah}) (3)
  \p{Joining_Group: Taw}  (Short: \p{Jg=Taw}) (1)
  \p{Joining_Group: Teh_Marbuta} (Short: \p{Jg=TehMarbuta}) (3)
  \p{Joining_Group: Teth} (Short: \p{Jg=Teth}) (2)
  \p{Joining_Group: Waw}  (Short: \p{Jg=Waw}) (15)
  \p{Joining_Group: Yeh}  (Short: \p{Jg=Yeh}) (7)
  \p{Joining_Group: Yeh_Barree} (Short: \p{Jg=YehBarree}) (2)
  \p{Joining_Group: Yeh_With_Tail} (Short: \p{Jg=YehWithTail}) (1)
  \p{Joining_Group: Yudh} (Short: \p{Jg=Yudh}) (1)
  \p{Joining_Group: Yudh_He} (Short: \p{Jg=YudhHe}) (1)
  \p{Joining_Group: Zain} (Short: \p{Jg=Zain}) (1)
  \p{Joining_Group: Zhain} (Short: \p{Jg=Zhain}) (1)
  \p{Joining_Type: C}     \p{Joining_Type=Join_Causing} (3)
  \p{Joining_Type: D}     \p{Joining_Type=Dual_Joining} (188)
  \p{Joining_Type: Dual_Joining} (Short: \p{Jt=D}) (188)
  \p{Joining_Type: Join_Causing} (Short: \p{Jt=C}) (3)
  \p{Joining_Type: L}     \p{Joining_Type=Left_Joining} (0)
  \p{Joining_Type: Left_Joining} (Short: \p{Jt=L}) (0)
  \p{Joining_Type: Non_Joining} (Short: \p{Jt=U}) (1_112_539)
  \p{Joining_Type: R}     \p{Joining_Type=Right_Joining} (74)
  \p{Joining_Type: Right_Joining} (Short: \p{Jt=R}) (74)
  \p{Joining_Type: T}     \p{Joining_Type=Transparent} (1308)
  \p{Joining_Type: Transparent} (Short: \p{Jt=T}) (1308)
  \p{Joining_Type: U}     \p{Joining_Type=Non_Joining} (1_112_539)
  \p{Jt: *}               \p{Joining_Type: *}
  \p{Kaithi}              \p{Script=Kaithi} (Short: \p{Kthi}; NOT
                            \p{Block=Kaithi}) (66)
  \p{Kali}                \p{Kayah_Li} (= \p{Script=Kayah_Li}) (48)
  \p{Kana}                \p{Katakana} (= \p{Script=Katakana}) (NOT
                            \p{Block=Katakana}) (299)
X \p{Kanbun}              \p{Block=Kanbun} (16)
X \p{Kangxi_Radicals}     \p{Block=Kangxi_Radicals} (224)
  \p{Kannada}             \p{Script=Kannada} (Short: \p{Knda}; NOT
                            \p{Block=Kannada}) (84)
```

```
  \p{Katakana}                \p{Script=Katakana} (Short: \p{Kana}; NOT
                                \p{Block=Katakana}) (299)
X \p{Katakana_Phonetic_Extensions} \p{Block=
                                Katakana_Phonetic_Extensions} (16)
  \p{Kayah_Li}                \p{Script=Kayah_Li} (Short: \p{Kali}) (48)
  \p{Khar}                    \p{Kharoshthi} (= \p{Script=Kharoshthi})
                                (NOT \p{Block=Kharoshthi}) (65)
  \p{Kharoshthi}              \p{Script=Kharoshthi} (Short: \p{Khar};
                                NOT \p{Block=Kharoshthi}) (65)
  \p{Khmer}                   \p{Script=Khmer} (Short: \p{Khmr}; NOT
                                \p{Block=Khmer}) (146)
X \p{Khmer_Symbols}           \p{Block=Khmer_Symbols} (32)
  \p{Khmr}                    \p{Khmer} (= \p{Script=Khmer}) (NOT
                                \p{Block=Khmer}) (146)
  \p{Knda}                    \p{Kannada} (= \p{Script=Kannada}) (NOT
                                \p{Block=Kannada}) (84)
  \p{Kthi}                    \p{Kaithi} (= \p{Script=Kaithi}) (NOT
                                \p{Block=Kaithi}) (66)
  \p{L}                       \p{Letter} (= \p{General_Category=Letter})
                                (99_537)
  \p{L&}                      \p{Cased_Letter} (= \p{General_Category=
                                Cased_Letter}) (3207)
  \p{L_}                      \p{Cased_Letter} (= \p{General_Category=
                                Cased_Letter}) (3207)
  \p{Lana}                    \p{Tai_Tham} (= \p{Script=Tai_Tham}) (NOT
                                \p{Block=Tai_Tham}) (127)
  \p{Lao}                     \p{Script=Lao} (NOT \p{Block=Lao}) (65)
  \p{Laoo}                    \p{Lao} (= \p{Script=Lao}) (NOT \p{Block=
                                Lao}) (65)
  \p{Latin}                   \p{Script=Latin} (Short: \p{Latn}) (1244)
X \p{Latin_1}                 \p{Latin_1_Supplement} (= \p{Block=
                                Latin_1_Supplement}) (128)
X \p{Latin_1_Supplement}      \p{Block=Latin_1_Supplement} (Short:
                                \p{InLatin1}) (128)
X \p{Latin_Extended_A}        \p{Block=Latin_Extended_A} (128)
X \p{Latin_Extended_Additional} \p{Block=Latin_Extended_Additional}
                                (256)
X \p{Latin_Extended_B}        \p{Block=Latin_Extended_B} (208)
X \p{Latin_Extended_C}        \p{Block=Latin_Extended_C} (32)
X \p{Latin_Extended_D}        \p{Block=Latin_Extended_D} (224)
  \p{Latn}                    \p{Latin} (= \p{Script=Latin}) (1244)
  \p{Lb: *}                   \p{Line_Break: *}
  \p{LC}                      \p{Cased_Letter} (= \p{General_Category=
                                Cased_Letter}) (3207)
  \p{Lepc}                    \p{Lepcha} (= \p{Script=Lepcha}) (NOT
                                \p{Block=Lepcha}) (74)
  \p{Lepcha}                  \p{Script=Lepcha} (Short: \p{Lepc}; NOT
                                \p{Block=Lepcha}) (74)
  \p{Letter}                  \p{General_Category=Letter} (Short: \p{L})
                                (99_537)
  \p{Letter_Number}           \p{General_Category=Letter_Number} (Short:
                                \p{Nl}) (224)
X \p{Letterlike_Symbols}      \p{Block=Letterlike_Symbols} (80)
  \p{Limb}                    \p{Limbu} (= \p{Script=Limbu}) (NOT
                                \p{Block=Limbu}) (66)
  \p{Limbu}                   \p{Script=Limbu} (Short: \p{Limb}; NOT
```

```
                            \p{Block=Limbu}) (66)
\p{Linb}                  \p{Linear_B} (= \p{Script=Linear_B}) (211)
\p{Line_Break: AI}        \p{Line_Break=Ambiguous} (644)
\p{Line_Break: AL}        \p{Line_Break=Alphabetic} (14_092)
\p{Line_Break: Alphabetic} (Short: \p{Lb=AL}) (14_092)
\p{Line_Break: Ambiguous} (Short: \p{Lb=AI}) (644)
\p{Line_Break: B2}        \p{Line_Break=Break_Both} (1)
\p{Line_Break: BA}        \p{Line_Break=Break_After} (137)
\p{Line_Break: BB}        \p{Line_Break=Break_Before} (19)
\p{Line_Break: BK}        \p{Line_Break=Mandatory_Break} (4)
\p{Line_Break: Break_After} (Short: \p{Lb=BA}) (137)
\p{Line_Break: Break_Before} (Short: \p{Lb=BB}) (19)
\p{Line_Break: Break_Both} (Short: \p{Lb=B2}) (1)
\p{Line_Break: Break_Symbols} (Short: \p{Lb=SY}) (1)
\p{Line_Break: Carriage_Return} (Short: \p{Lb=CR}) (1)
\p{Line_Break: CB}        \p{Line_Break=Contingent_Break} (1)
\p{Line_Break: CL}        \p{Line_Break=Close_Punctuation} (87)
\p{Line_Break: Close_Parenthesis} (Short: \p{Lb=CP}) (2)
\p{Line_Break: Close_Punctuation} (Short: \p{Lb=CL}) (87)
\p{Line_Break: CM}        \p{Line_Break=Combining_Mark} (1436)
\p{Line_Break: Combining_Mark} (Short: \p{Lb=CM}) (1436)
\p{Line_Break: Complex_Context} (Short: \p{Lb=SA}) (662)
\p{Line_Break: Contingent_Break} (Short: \p{Lb=CB}) (1)
\p{Line_Break: CP}        \p{Line_Break=Close_Parenthesis} (2)
\p{Line_Break: CR}        \p{Line_Break=Carriage_Return} (1)
\p{Line_Break: EX}        \p{Line_Break=Exclamation} (34)
\p{Line_Break: Exclamation} (Short: \p{Lb=EX}) (34)
\p{Line_Break: GL}        \p{Line_Break=Glue} (16)
\p{Line_Break: Glue}      (Short: \p{Lb=GL}) (16)
\p{Line_Break: H2}        (Short: \p{Lb=H2}) (399)
\p{Line_Break: H3}        (Short: \p{Lb=H3}) (10_773)
\p{Line_Break: HY}        \p{Line_Break=Hyphen} (1)
\p{Line_Break: Hyphen}    (Short: \p{Lb=HY}) (1)
\p{Line_Break: ID}        \p{Line_Break=Ideographic} (161_775)
\p{Line_Break: Ideographic} (Short: \p{Lb=ID}) (161_775)
\p{Line_Break: IN}        \p{Line_Break=Inseparable} (4)
\p{Line_Break: Infix_Numeric} (Short: \p{Lb=IS}) (13)
\p{Line_Break: Inseparable} (Short: \p{Lb=IN}) (4)
\p{Line_Break: Inseperable} \p{Line_Break=Inseparable} (4)
\p{Line_Break: IS}        \p{Line_Break=Infix_Numeric} (13)
\p{Line_Break: JL}        (Short: \p{Lb=JL}) (125)
\p{Line_Break: JT}        (Short: \p{Lb=JT}) (137)
\p{Line_Break: JV}        (Short: \p{Lb=JV}) (95)
\p{Line_Break: LF}        \p{Line_Break=Line_Feed} (1)
\p{Line_Break: Line_Feed} (Short: \p{Lb=LF}) (1)
\p{Line_Break: Mandatory_Break} (Short: \p{Lb=BK}) (4)
\p{Line_Break: Next_Line} (Short: \p{Lb=NL}) (1)
\p{Line_Break: NL}        \p{Line_Break=Next_Line} (1)
\p{Line_Break: Nonstarter} (Short: \p{Lb=NS}) (77)
\p{Line_Break: NS}        \p{Line_Break=Nonstarter} (77)
\p{Line_Break: NU}        \p{Line_Break=Numeric} (403)
\p{Line_Break: Numeric} (Short: \p{Lb=NU}) (403)
\p{Line_Break: OP}        \p{Line_Break=Open_Punctuation} (81)
\p{Line_Break: Open_Punctuation} (Short: \p{Lb=OP}) (81)
\p{Line_Break: PO}        \p{Line_Break=Postfix_Numeric} (28)
\p{Line_Break: Postfix_Numeric} (Short: \p{Lb=PO}) (28)
```

```
    \p{Line_Break: PR}      \p{Line_Break=Prefix_Numeric} (43)
    \p{Line_Break: Prefix_Numeric} (Short: \p{Lb=PR}) (43)
    \p{Line_Break: QU}      \p{Line_Break=Quotation} (34)
    \p{Line_Break: Quotation} (Short: \p{Lb=QU}) (34)
    \p{Line_Break: SA}      \p{Line_Break=Complex_Context} (662)
  D \p{Line_Break: SG}      \p{Line_Break=Surrogate} (2048)
    \p{Line_Break: SP}      \p{Line_Break=Space} (1)
    \p{Line_Break: Space}   (Short: \p{Lb=SP}) (1)
  D \p{Line_Break: Surrogate} Deprecated by Unicode because surrogates
                              should never appear in well-formed text,
                              and therefore shouldn't be the basis for
                              line breaking (Short: \p{Lb=SG}) (2048)
    \p{Line_Break: SY}      \p{Line_Break=Break_Symbols} (1)
    \p{Line_Break: Unknown} (Short: \p{Lb=XX}) (920_933)
    \p{Line_Break: WJ}      \p{Line_Break=Word_Joiner} (2)
    \p{Line_Break: Word_Joiner} (Short: \p{Lb=WJ}) (2)
    \p{Line_Break: XX}      \p{Line_Break=Unknown} (920_933)
    \p{Line_Break: ZW}      \p{Line_Break=ZWSpace} (1)
    \p{Line_Break: ZWSpace} (Short: \p{Lb=ZW}) (1)
    \p{Line_Separator}      \p{General_Category=Line_Separator}
                              (Short: \p{Zl}) (1)
    \p{Linear_B}            \p{Script=Linear_B} (Short: \p{Linb}) (211)
  X \p{Linear_B_Ideograms}  \p{Block=Linear_B_Ideograms} (128)
  X \p{Linear_B_Syllabary}  \p{Block=Linear_B_Syllabary} (128)
    \p{Lisu}                \p{Script=Lisu} (48)
    \p{Ll}                  \p{Lowercase_Letter} (=
                              \p{General_Category=Lowercase_Letter})
                              (1749)
    \p{Lm}                  \p{Modifier_Letter} (=
                              \p{General_Category=Modifier_Letter})
                              (202)
    \p{Lo}                  \p{Other_Letter} (= \p{General_Category=
                              Other_Letter}) (96_128)
    \p{LOE}                 \p{Logical_Order_Exception} (=
                              \p{Logical_Order_Exception=Y}) (15)
    \p{LOE: *}              \p{Logical_Order_Exception: *}
    \p{Logical_Order_Exception} \p{Logical_Order_Exception=Y} (Short:
                              \p{LOE}) (15)
    \p{Logical_Order_Exception: N*} (Short: \p{LOE=N}, \P{LOE})
                              (1_114_097)
    \p{Logical_Order_Exception: Y*} (Short: \p{LOE=Y}, \p{LOE}) (15)
  X \p{Low_Surrogates}      \p{Block=Low_Surrogates} (1024)
    \p{Lower}               \p{Lowercase=Y} (1908)
    \p{Lower: *}            \p{Lowercase: *}
    \p{Lowercase}           \p{Lower} (= \p{Lowercase=Y}) (1908)
    \p{Lowercase: N*}       (Short: \p{Lower=N}, \P{Lower}) (1_112_204)
    \p{Lowercase: Y*}       (Short: \p{Lower=Y}, \p{Lower}) (1908)
    \p{Lowercase_Letter}    \p{General_Category=Lowercase_Letter}
                              (Short: \p{Ll}) (1749)
    \p{Lt}                  \p{Title} (= \p{General_Category=
                              Titlecase_Letter}) (31)
    \p{Lu}                  \p{Uppercase_Letter} (=
                              \p{General_Category=Uppercase_Letter})
                              (1427)
    \p{Lyci}                \p{Lycian} (= \p{Script=Lycian}) (NOT
                              \p{Block=Lycian}) (29)
```

```
  \p{Lycian}              \p{Script=Lycian} (Short: \p{Lyci}; NOT
                            \p{Block=Lycian}) (29)
  \p{Lydi}                \p{Lydian} (= \p{Script=Lydian}) (NOT
                            \p{Block=Lydian}) (27)
  \p{Lydian}              \p{Script=Lydian} (Short: \p{Lydi}; NOT
                            \p{Block=Lydian}) (27)
  \p{M}                   \p{Mark} (= \p{General_Category=Mark})
                            (1451)
X \p{Mahjong_Tiles}       \p{Block=Mahjong_Tiles} (48)
  \p{Malayalam}           \p{Script=Malayalam} (Short: \p{Mlym}; NOT
                            \p{Block=Malayalam}) (95)
  \p{Mark}                \p{General_Category=Mark} (Short: \p{M})
                            (1451)
  \p{Math}                \p{Math=Y} (2161)
  \p{Math: N*}            (Single: \P{Math}) (1_111_951)
  \p{Math: Y*}            (Single: \p{Math}) (2161)
  \p{Math_Symbol}         \p{General_Category=Math_Symbol} (Short:
                            \p{Sm}) (945)
X \p{Mathematical_Alphanumeric_Symbols} \p{Block=
                            Mathematical_Alphanumeric_Symbols} (1024)
X \p{Mathematical_Operators} \p{Block=Mathematical_Operators} (256)
  \p{Mc}                  \p{Spacing_Mark} (= \p{General_Category=
                            Spacing_Mark}) (276)
  \p{Me}                  \p{Enclosing_Mark} (= \p{General_Category=
                            Enclosing_Mark}) (13)
  \p{Meetei_Mayek}        \p{Script=Meetei_Mayek} (Short: \p{Mtei};
                            NOT \p{Block=Meetei_Mayek}) (56)
X \p{Miscellaneous_Mathematical_Symbols_A} \p{Block=
                            Miscellaneous_Mathematical_Symbols_A}
                            (48)
X \p{Miscellaneous_Mathematical_Symbols_B} \p{Block=
                            Miscellaneous_Mathematical_Symbols_B}
                            (128)
X \p{Miscellaneous_Symbols} \p{Block=Miscellaneous_Symbols} (256)
X \p{Miscellaneous_Symbols_And_Arrows} \p{Block=
                            Miscellaneous_Symbols_And_Arrows} (256)
X \p{Miscellaneous_Technical} \p{Block=Miscellaneous_Technical} (256)
  \p{Mlym}                \p{Malayalam} (= \p{Script=Malayalam})
                            (NOT \p{Block=Malayalam}) (95)
  \p{Mn}                  \p{Nonspacing_Mark} (=
                            \p{General_Category=Nonspacing_Mark})
                            (1162)
  \p{Modifier_Letter}     \p{General_Category=Modifier_Letter}
                            (Short: \p{Lm}) (202)
  \p{Modifier_Symbol}     \p{General_Category=Modifier_Symbol}
                            (Short: \p{Sk}) (99)
X \p{Modifier_Tone_Letters} \p{Block=Modifier_Tone_Letters} (32)
  \p{Mong}                \p{Mongolian} (= \p{Script=Mongolian})
                            (NOT \p{Block=Mongolian}) (153)
  \p{Mongolian}           \p{Script=Mongolian} (Short: \p{Mong}; NOT
                            \p{Block=Mongolian}) (153)
  \p{Mtei}                \p{Meetei_Mayek} (= \p{Script=
                            Meetei_Mayek}) (NOT \p{Block=
                            Meetei_Mayek}) (56)
X \p{Musical_Symbols}     \p{Block=Musical_Symbols} (256)
  \p{Myanmar}             \p{Script=Myanmar} (Short: \p{Mymr}; NOT
```

```
                                \p{Block=Myanmar}) (188)
X \p{Myanmar_Extended_A}  \p{Block=Myanmar_Extended_A} (32)
  \p{Mymr}                \p{Myanmar} (= \p{Script=Myanmar}) (NOT
                            \p{Block=Myanmar}) (188)
  \p{N}                   \p{Number} (= \p{General_Category=Number})
                            (1064)
  \p{NChar}               \p{Noncharacter_Code_Point} (=
                            \p{Noncharacter_Code_Point=Y}) (66)
  \p{NChar: *}            \p{Noncharacter_Code_Point: *}
  \p{Nd}                  \p{Digit} (= \p{General_Category=
                            Decimal_Number}) (411)
  \p{New_Tai_Lue}         \p{Script=New_Tai_Lue} (Short: \p{Talu};
                            NOT \p{Block=New_Tai_Lue}) (83)
  \p{NFC_QC: *}           \p{NFC_Quick_Check: *}
  \p{NFC_Quick_Check: M}  \p{NFC_Quick_Check=Maybe} (103)
  \p{NFC_Quick_Check: Maybe} (Short: \p{NFCQC=M}) (103)
  \p{NFC_Quick_Check: N}  \p{NFC_Quick_Check=No} (NOT
                            \P{NFC_Quick_Check} NOR \P{NFC_QC} NOR
                            \P{Is_NFC_Quick_Check} NOR
                            \P{Is_NFC_QC}) (1118)
  \p{NFC_Quick_Check: No} (Short: \p{NFCQC=N}; NOT
                            \P{NFC_Quick_Check} NOR \P{NFC_QC} NOR
                            \P{Is_NFC_Quick_Check} NOR
                            \P{Is_NFC_QC}) (1118)
  \p{NFC_Quick_Check: Y}  \p{NFC_Quick_Check=Yes} (NOT
                            \p{NFC_Quick_Check} NOR \p{NFC_QC} NOR
                            \p{Is_NFC_Quick_Check} NOR
                            \p{Is_NFC_QC}) (1_112_891)
  \p{NFC_Quick_Check: Yes} (Short: \p{NFCQC=Y}; NOT
                            \p{NFC_Quick_Check} NOR \p{NFC_QC} NOR
                            \p{Is_NFC_Quick_Check} NOR
                            \p{Is_NFC_QC}) (1_112_891)
  \p{NFD_QC: *}           \p{NFD_Quick_Check: *}
  \p{NFD_Quick_Check: N}  \p{NFD_Quick_Check=No} (NOT
                            \P{NFD_Quick_Check} NOR \P{NFD_QC} NOR
                            \P{Is_NFD_Quick_Check} NOR
                            \P{Is_NFD_QC}) (13_221)
  \p{NFD_Quick_Check: No} (Short: \p{NFDQC=N}; NOT
                            \P{NFD_Quick_Check} NOR \P{NFD_QC} NOR
                            \P{Is_NFD_Quick_Check} NOR
                            \P{Is_NFD_QC}) (13_221)
  \p{NFD_Quick_Check: Y}  \p{NFD_Quick_Check=Yes} (NOT
                            \p{NFD_Quick_Check} NOR \p{NFD_QC} NOR
                            \p{Is_NFD_Quick_Check} NOR
                            \p{Is_NFD_QC}) (1_100_891)
  \p{NFD_Quick_Check: Yes} (Short: \p{NFDQC=Y}; NOT
                            \p{NFD_Quick_Check} NOR \p{NFD_QC} NOR
                            \p{Is_NFD_Quick_Check} NOR
                            \p{Is_NFD_QC}) (1_100_891)
  \p{NFKC_QC: *}          \p{NFKC_Quick_Check: *}
  \p{NFKC_Quick_Check: M} \p{NFKC_Quick_Check=Maybe} (103)
  \p{NFKC_Quick_Check: Maybe} (Short: \p{NFKCQC=M}) (103)
  \p{NFKC_Quick_Check: N} \p{NFKC_Quick_Check=No} (NOT
                            \P{NFKC_Quick_Check} NOR \P{NFKC_QC} NOR
                            \P{Is_NFKC_Quick_Check} NOR
                            \P{Is_NFKC_QC}) (4597)
```

```
    \p{NFKC_Quick_Check: No} (Short: \p{NFKCQC=N}; NOT
                              \P{NFKC_Quick_Check} NOR \P{NFKC_QC} NOR
                              \P{Is_NFKC_Quick_Check} NOR
                              \P{Is_NFKC_QC}) (4597)
    \p{NFKC_Quick_Check: Y} \p{NFKC_Quick_Check=Yes} (NOT
                              \p{NFKC_Quick_Check} NOR \p{NFKC_QC} NOR
                              \p{Is_NFKC_Quick_Check} NOR
                              \p{Is_NFKC_QC}) (1_109_412)
    \p{NFKC_Quick_Check: Yes} (Short: \p{NFKCQC=Y}; NOT
                              \p{NFKC_Quick_Check} NOR \p{NFKC_QC} NOR
                              \p{Is_NFKC_Quick_Check} NOR
                              \p{Is_NFKC_QC}) (1_109_412)
    \p{NFKD_QC: *}          \p{NFKD_Quick_Check: *}
    \p{NFKD_Quick_Check: N} \p{NFKD_Quick_Check=No} (NOT
                              \P{NFKD_Quick_Check} NOR \P{NFKD_QC} NOR
                              \P{Is_NFKD_Quick_Check} NOR
                              \P{Is_NFKD_QC}) (16_688)
    \p{NFKD_Quick_Check: No} (Short: \p{NFKDQC=N}; NOT
                              \P{NFKD_Quick_Check} NOR \P{NFKD_QC} NOR
                              \P{Is_NFKD_Quick_Check} NOR
                              \P{Is_NFKD_QC}) (16_688)
    \p{NFKD_Quick_Check: Y} \p{NFKD_Quick_Check=Yes} (NOT
                              \p{NFKD_Quick_Check} NOR \p{NFKD_QC} NOR
                              \p{Is_NFKD_Quick_Check} NOR
                              \p{Is_NFKD_QC}) (1_097_424)
    \p{NFKD_Quick_Check: Yes} (Short: \p{NFKDQC=Y}; NOT
                              \p{NFKD_Quick_Check} NOR \p{NFKD_QC} NOR
                              \p{Is_NFKD_Quick_Check} NOR
                              \p{Is_NFKD_QC}) (1_097_424)
    \p{Nko}                 \p{Script=Nko} (NOT \p{NKo}) (59)
    \p{Nkoo}                \p{Nko} (= \p{Script=Nko}) (NOT \p{NKo})
                              (59)
    \p{Nl}                  \p{Letter_Number} (= \p{General_Category=
                              Letter_Number}) (224)
    \p{No}                  \p{Other_Number} (= \p{General_Category=
                              Other_Number}) (429)
  X \p{No_Block}            \p{Block=No_Block} (864_192)
    \p{Noncharacter_Code_Point} \p{Noncharacter_Code_Point=Y} (Short:
                              \p{NChar}) (66)
    \p{Noncharacter_Code_Point: N*} (Short: \p{NChar=N}, \P{NChar})
                              (1_114_046)
    \p{Noncharacter_Code_Point: Y*} (Short: \p{NChar=Y}, \p{NChar})
                              (66)
    \p{Nonspacing_Mark}     \p{General_Category=Nonspacing_Mark}
                              (Short: \p{Mn}) (1162)
    \p{Nt: *}               \p{Numeric_Type: *}
    \p{Number}              \p{General_Category=Number} (Short: \p{N})
                              (1064)
  X \p{Number_Forms}        \p{Block=Number_Forms} (64)
    \p{Numeric_Type: De}    \p{Numeric_Type=Decimal} (411)
    \p{Numeric_Type: Decimal} (Short: \p{Nt=De}) (411)
    \p{Numeric_Type: Di}    \p{Numeric_Type=Digit} (118)
    \p{Numeric_Type: Digit} (Short: \p{Nt=Di}) (118)
    \p{Numeric_Type: None}  (Short: \p{Nt=None}) (1_112_971)
    \p{Numeric_Type: Nu}    \p{Numeric_Type=Numeric} (612)
    \p{Numeric_Type: Numeric} (Short: \p{Nt=Nu}) (612)
```

```
T \p{Numeric_Value: -1/2} (Short: \p{Nv=-1/2}) (1)
T \p{Numeric_Value: 0}    (Short: \p{Nv=0}) (55)
T \p{Numeric_Value: 1/16} (Short: \p{Nv=1/16}) (2)
T \p{Numeric_Value: 1/10} (Short: \p{Nv=1/10}) (1)
T \p{Numeric_Value: 1/9}  (Short: \p{Nv=1/9}) (1)
T \p{Numeric_Value: 1/8}  (Short: \p{Nv=1/8}) (4)
T \p{Numeric_Value: 1/7}  (Short: \p{Nv=1/7}) (1)
T \p{Numeric_Value: 1/6}  (Short: \p{Nv=1/6}) (2)
T \p{Numeric_Value: 3/16} (Short: \p{Nv=3/16}) (2)
T \p{Numeric_Value: 1/5}  (Short: \p{Nv=1/5}) (1)
T \p{Numeric_Value: 1/4}  (Short: \p{Nv=1/4}) (8)
T \p{Numeric_Value: 1/3}  (Short: \p{Nv=1/3}) (4)
T \p{Numeric_Value: 3/8}  (Short: \p{Nv=3/8}) (1)
T \p{Numeric_Value: 2/5}  (Short: \p{Nv=2/5}) (1)
T \p{Numeric_Value: 1/2}  (Short: \p{Nv=1/2}) (9)
T \p{Numeric_Value: 3/5}  (Short: \p{Nv=3/5}) (1)
T \p{Numeric_Value: 5/8}  (Short: \p{Nv=5/8}) (1)
T \p{Numeric_Value: 2/3}  (Short: \p{Nv=2/3}) (5)
T \p{Numeric_Value: 3/4}  (Short: \p{Nv=3/4}) (5)
T \p{Numeric_Value: 4/5}  (Short: \p{Nv=4/5}) (1)
T \p{Numeric_Value: 5/6}  (Short: \p{Nv=5/6}) (2)
T \p{Numeric_Value: 7/8}  (Short: \p{Nv=7/8}) (1)
T \p{Numeric_Value: 1}    (Short: \p{Nv=1}) (91)
T \p{Numeric_Value: 3/2}  (Short: \p{Nv=3/2}) (1)
T \p{Numeric_Value: 2}    (Short: \p{Nv=2}) (94)
T \p{Numeric_Value: 5/2}  (Short: \p{Nv=5/2}) (1)
T \p{Numeric_Value: 3}    (Short: \p{Nv=3}) (96)
T \p{Numeric_Value: 7/2}  (Short: \p{Nv=7/2}) (1)
T \p{Numeric_Value: 4}    (Short: \p{Nv=4}) (87)
T \p{Numeric_Value: 9/2}  (Short: \p{Nv=9/2}) (1)
T \p{Numeric_Value: 5}    (Short: \p{Nv=5}) (84)
T \p{Numeric_Value: 11/2} (Short: \p{Nv=11/2}) (1)
T \p{Numeric_Value: 6}    (Short: \p{Nv=6}) (76)
T \p{Numeric_Value: 13/2} (Short: \p{Nv=13/2}) (1)
T \p{Numeric_Value: 7}    (Short: \p{Nv=7}) (75)
T \p{Numeric_Value: 15/2} (Short: \p{Nv=15/2}) (1)
T \p{Numeric_Value: 8}    (Short: \p{Nv=8}) (71)
T \p{Numeric_Value: 17/2} (Short: \p{Nv=17/2}) (1)
T \p{Numeric_Value: 9}    (Short: \p{Nv=9}) (75)
T \p{Numeric_Value: 10}   (Short: \p{Nv=10}) (38)
T \p{Numeric_Value: 11}   (Short: \p{Nv=11}) (6)
T \p{Numeric_Value: 12}   (Short: \p{Nv=12}) (6)
T \p{Numeric_Value: 13}   (Short: \p{Nv=13}) (4)
T \p{Numeric_Value: 14}   (Short: \p{Nv=14}) (4)
T \p{Numeric_Value: 15}   (Short: \p{Nv=15}) (4)
T \p{Numeric_Value: 16}   (Short: \p{Nv=16}) (5)
T \p{Numeric_Value: 17}   (Short: \p{Nv=17}) (5)
T \p{Numeric_Value: 18}   (Short: \p{Nv=18}) (5)
T \p{Numeric_Value: 19}   (Short: \p{Nv=19}) (5)
T \p{Numeric_Value: 20}   (Short: \p{Nv=20}) (17)
T \p{Numeric_Value: 21}   (Short: \p{Nv=21}) (1)
T \p{Numeric_Value: 22}   (Short: \p{Nv=22}) (1)
T \p{Numeric_Value: 23}   (Short: \p{Nv=23}) (1)
T \p{Numeric_Value: 24}   (Short: \p{Nv=24}) (1)
T \p{Numeric_Value: 25}   (Short: \p{Nv=25}) (1)
T \p{Numeric_Value: 26}   (Short: \p{Nv=26}) (1)
```

```
T \p{Numeric_Value: 27}   (Short: \p{Nv=27}) (1)
T \p{Numeric_Value: 28}   (Short: \p{Nv=28}) (1)
T \p{Numeric_Value: 29}   (Short: \p{Nv=29}) (1)
T \p{Numeric_Value: 30}   (Short: \p{Nv=30}) (9)
T \p{Numeric_Value: 31}   (Short: \p{Nv=31}) (1)
T \p{Numeric_Value: 32}   (Short: \p{Nv=32}) (1)
T \p{Numeric_Value: 33}   (Short: \p{Nv=33}) (1)
T \p{Numeric_Value: 34}   (Short: \p{Nv=34}) (1)
T \p{Numeric_Value: 35}   (Short: \p{Nv=35}) (1)
T \p{Numeric_Value: 36}   (Short: \p{Nv=36}) (1)
T \p{Numeric_Value: 37}   (Short: \p{Nv=37}) (1)
T \p{Numeric_Value: 38}   (Short: \p{Nv=38}) (1)
T \p{Numeric_Value: 39}   (Short: \p{Nv=39}) (1)
T \p{Numeric_Value: 40}   (Short: \p{Nv=40}) (8)
T \p{Numeric_Value: 41}   (Short: \p{Nv=41}) (1)
T \p{Numeric_Value: 42}   (Short: \p{Nv=42}) (1)
T \p{Numeric_Value: 43}   (Short: \p{Nv=43}) (1)
T \p{Numeric_Value: 44}   (Short: \p{Nv=44}) (1)
T \p{Numeric_Value: 45}   (Short: \p{Nv=45}) (1)
T \p{Numeric_Value: 46}   (Short: \p{Nv=46}) (1)
T \p{Numeric_Value: 47}   (Short: \p{Nv=47}) (1)
T \p{Numeric_Value: 48}   (Short: \p{Nv=48}) (1)
T \p{Numeric_Value: 49}   (Short: \p{Nv=49}) (1)
T \p{Numeric_Value: 50}   (Short: \p{Nv=50}) (18)
T \p{Numeric_Value: 60}   (Short: \p{Nv=60}) (4)
T \p{Numeric_Value: 70}   (Short: \p{Nv=70}) (4)
T \p{Numeric_Value: 80}   (Short: \p{Nv=80}) (4)
T \p{Numeric_Value: 90}   (Short: \p{Nv=90}) (5)
T \p{Numeric_Value: 100}  (Short: \p{Nv=100}) (19)
T \p{Numeric_Value: 200}  (Short: \p{Nv=200}) (2)
T \p{Numeric_Value: 300}  (Short: \p{Nv=300}) (3)
T \p{Numeric_Value: 400}  (Short: \p{Nv=400}) (2)
T \p{Numeric_Value: 500}  (Short: \p{Nv=500}) (12)
T \p{Numeric_Value: 600}  (Short: \p{Nv=600}) (2)
T \p{Numeric_Value: 700}  (Short: \p{Nv=700}) (2)
T \p{Numeric_Value: 800}  (Short: \p{Nv=800}) (2)
T \p{Numeric_Value: 900}  (Short: \p{Nv=900}) (3)
T \p{Numeric_Value: 1000} (Short: \p{Nv=1000}) (16)
T \p{Numeric_Value: 2000} (Short: \p{Nv=2000}) (1)
T \p{Numeric_Value: 3000} (Short: \p{Nv=3000}) (1)
T \p{Numeric_Value: 4000} (Short: \p{Nv=4000}) (1)
T \p{Numeric_Value: 5000} (Short: \p{Nv=5000}) (5)
T \p{Numeric_Value: 6000} (Short: \p{Nv=6000}) (1)
T \p{Numeric_Value: 7000} (Short: \p{Nv=7000}) (1)
T \p{Numeric_Value: 8000} (Short: \p{Nv=8000}) (1)
T \p{Numeric_Value: 9000} (Short: \p{Nv=9000}) (1)
T \p{Numeric_Value: 10000} (= 1.0e+04) (Short: \p{Nv=10000}) (7)
T \p{Numeric_Value: 20000} (= 2.0e+04) (Short: \p{Nv=20000}) (1)
T \p{Numeric_Value: 30000} (= 3.0e+04) (Short: \p{Nv=30000}) (1)
T \p{Numeric_Value: 40000} (= 4.0e+04) (Short: \p{Nv=40000}) (1)
T \p{Numeric_Value: 50000} (= 5.0e+04) (Short: \p{Nv=50000}) (4)
T \p{Numeric_Value: 60000} (= 6.0e+04) (Short: \p{Nv=60000}) (1)
T \p{Numeric_Value: 70000} (= 7.0e+04) (Short: \p{Nv=70000}) (1)
T \p{Numeric_Value: 80000} (= 8.0e+04) (Short: \p{Nv=80000}) (1)
T \p{Numeric_Value: 90000} (= 9.0e+04) (Short: \p{Nv=90000}) (1)
T \p{Numeric_Value: 100000} (= 1.0e+05) (Short: \p{Nv=100000}) (1)
```

```
T \p{Numeric_Value: 100000000} (= 1.0e+08) (Short: \p{Nv=100000000})
                          (2)
T \p{Numeric_Value: 1000000000000} (= 1.0e+12) (Short: \p{Nv=
                          1000000000000}) (1)
  \p{Numeric_Value: NaN}  (Short: \p{Nv=NaN}) (1_112_971)
  \p{Nv: *}               \p{Numeric_Value: *}
D \p{OAlpha}              \p{Other_Alphabetic} (=
                          \p{Other_Alphabetic=Y}) (759)
D \p{OAlpha: *}           \p{Other_Alphabetic: *}
D \p{ODI}                 \p{Other_Default_Ignorable_Code_Point} (=
                          \p{Other_Default_Ignorable_Code_Point=
                          Y}) (3778)
D \p{ODI: *}              \p{Other_Default_Ignorable_Code_Point: *}
  \p{Ogam}                \p{Ogham} (= \p{Script=Ogham}) (NOT
                          \p{Block=Ogham}) (29)
  \p{Ogham}               \p{Script=Ogham} (Short: \p{Ogam}; NOT
                          \p{Block=Ogham}) (29)
D \p{OGr_Ext}             \p{Other_Grapheme_Extend} (=
                          \p{Other_Grapheme_Extend=Y}) (23)
D \p{OGr_Ext: *}          \p{Other_Grapheme_Extend: *}
D \p{OIDC}                \p{Other_ID_Continue} (=
                          \p{Other_ID_Continue=Y}) (11)
D \p{OIDC: *}             \p{Other_ID_Continue: *}
D \p{OIDS}                \p{Other_ID_Start} (= \p{Other_ID_Start=
                          Y}) (4)
D \p{OIDS: *}             \p{Other_ID_Start: *}
  \p{Ol_Chiki}            \p{Script=Ol_Chiki} (Short: \p{Olck}) (48)
  \p{Olck}                \p{Ol_Chiki} (= \p{Script=Ol_Chiki}) (48)
  \p{Old_Italic}          \p{Script=Old_Italic} (Short: \p{Ital};
                          NOT \p{Block=Old_Italic}) (35)
  \p{Old_Persian}         \p{Script=Old_Persian} (Short: \p{Xpeo};
                          NOT \p{Block=Old_Persian}) (50)
  \p{Old_South_Arabian}   \p{Script=Old_South_Arabian} (Short:
                          \p{Sarb}) (32)
  \p{Old_Turkic}          \p{Script=Old_Turkic} (Short: \p{Orkh};
                          NOT \p{Block=Old_Turkic}) (73)
D \p{OLower}              \p{Other_Lowercase} (= \p{Other_Lowercase=
                          Y}) (159)
D \p{OLower: *}           \p{Other_Lowercase: *}
D \p{OMath}               \p{Other_Math} (= \p{Other_Math=Y}) (1216)
D \p{OMath: *}            \p{Other_Math: *}
  \p{Open_Punctuation}    \p{General_Category=Open_Punctuation}
                          (Short: \p{Ps}) (72)
X \p{Optical_Character_Recognition} \p{Block=
                          Optical_Character_Recognition} (32)
  \p{Oriya}               \p{Script=Oriya} (Short: \p{Orya}; NOT
                          \p{Block=Oriya}) (84)
  \p{Orkh}                \p{Old_Turkic} (= \p{Script=Old_Turkic})
                          (NOT \p{Block=Old_Turkic}) (73)
  \p{Orya}                \p{Oriya} (= \p{Script=Oriya}) (NOT
                          \p{Block=Oriya}) (84)
  \p{Osma}                \p{Osmanya} (= \p{Script=Osmanya}) (NOT
                          \p{Block=Osmanya}) (40)
  \p{Osmanya}             \p{Script=Osmanya} (Short: \p{Osma}; NOT
                          \p{Block=Osmanya}) (40)
  \p{Other}               \p{General_Category=Other} (Short: \p{C})
```

```
                              (1_006_956)
D \p{Other_Alphabetic}    \p{Other_Alphabetic=Y} (Short: \p{OAlpha})
                              (759)
D \p{Other_Alphabetic: N*} Used by Unicode internally for generating
                              the Alphabetic property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OAlpha=N},
                              \P{OAlpha}) (1_113_353)
D \p{Other_Alphabetic: Y*} Used by Unicode internally for generating
                              the Alphabetic property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OAlpha=Y},
                              \p{OAlpha}) (759)
D \p{Other_Default_Ignorable_Code_Point}
                              \p{Other_Default_Ignorable_Code_Point=Y}
                              (Short: \p{ODI}) (3778)
D \p{Other_Default_Ignorable_Code_Point: N*} Used by Unicode
                              internally for generating the
                              Default_Ignorable_Code_Point property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{ODI=N}, \P{ODI}) (1_110_334)
D \p{Other_Default_Ignorable_Code_Point: Y*} Used by Unicode
                              internally for generating the
                              Default_Ignorable_Code_Point property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{ODI=Y}, \p{ODI}) (3778)
D \p{Other_Grapheme_Extend} \p{Other_Grapheme_Extend=Y} (Short:
                              \p{OGrExt}) (23)
D \p{Other_Grapheme_Extend: N*} Used by Unicode internally for
                              generating the Grapheme_Extend property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{OGrExt=N}, \P{OGrExt}) (1_114_089)
D \p{Other_Grapheme_Extend: Y*} Used by Unicode internally for
                              generating the Grapheme_Extend property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{OGrExt=Y}, \p{OGrExt}) (23)
D \p{Other_ID_Continue}   \p{Other_ID_Continue=Y} (Short: \p{OIDC})
                              (11)
D \p{Other_ID_Continue: N*} Used by Unicode internally for
                              generating the ID_Continue property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{OIDC=N}, \P{OIDC}) (1_114_101)
D \p{Other_ID_Continue: Y*} Used by Unicode internally for
                              generating the ID_Continue property
                              (which should be used instead) and not
                              intended to be used stand-alone (Short:
                              \p{OIDC=Y}, \p{OIDC}) (11)
D \p{Other_ID_Start}      \p{Other_ID_Start=Y} (Short: \p{OIDS}) (4)
D \p{Other_ID_Start: N*}  Used by Unicode internally for generating
                              the ID_Start property (which should be
                              used instead) and not intended to be
```

```
                              used stand-alone (Short: \p{OIDS=N},
                              \P{OIDS}) (1_114_108)
D \p{Other_ID_Start: Y*}  Used by Unicode internally for generating
                              the ID_Start property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OIDS=Y},
                              \p{OIDS}) (4)
  \p{Other_Letter}        \p{General_Category=Other_Letter} (Short:
                              \p{Lo}) (96_128)
D \p{Other_Lowercase}     \p{Other_Lowercase=Y} (Short: \p{OLower})
                              (159)
D \p{Other_Lowercase: N*} Used by Unicode internally for generating
                              the Lowercase property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OLower=N},
                              \P{OLower}) (1_113_953)
D \p{Other_Lowercase: Y*} Used by Unicode internally for generating
                              the Lowercase property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OLower=Y},
                              \p{OLower}) (159)
D \p{Other_Math}          \p{Other_Math=Y} (Short: \p{OMath}) (1216)
D \p{Other_Math: N*}      Used by Unicode internally for generating
                              the Math property (which should be used
                              instead) and not intended to be used
                              stand-alone (Short: \p{OMath=N},
                              \P{OMath}) (1_112_896)
D \p{Other_Math: Y*}      Used by Unicode internally for generating
                              the Math property (which should be used
                              instead) and not intended to be used
                              stand-alone (Short: \p{OMath=Y},
                              \p{OMath}) (1216)
  \p{Other_Number}        \p{General_Category=Other_Number} (Short:
                              \p{No}) (429)
  \p{Other_Punctuation}   \p{General_Category=Other_Punctuation}
                              (Short: \p{Po}) (389)
  \p{Other_Symbol}        \p{General_Category=Other_Symbol} (Short:
                              \p{So}) (3409)
D \p{Other_Uppercase}     \p{Other_Uppercase=Y} (Short: \p{OUpper})
                              (42)
D \p{Other_Uppercase: N*} Used by Unicode internally for generating
                              the Uppercase property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OUpper=N},
                              \P{OUpper}) (1_114_070)
D \p{Other_Uppercase: Y*} Used by Unicode internally for generating
                              the Uppercase property (which should be
                              used instead) and not intended to be
                              used stand-alone (Short: \p{OUpper=Y},
                              \p{OUpper}) (42)
D \p{OUpper}              \p{Other_Uppercase} (= \p{Other_Uppercase=
                              Y}) (42)
D \p{OUpper: *}           \p{Other_Uppercase: *}
  \p{P}                   \p{Punct} (= \p{General_Category=
                              Punctuation}) (585)
  \p{Paragraph_Separator} \p{General_Category=Paragraph_Separator}
```

```
                                (Short: \p{Zp}) (1)
    \p{Pat_Syn}                 \p{Pattern_Syntax} (= \p{Pattern_Syntax=
                                Y}) (2760)
    \p{Pat_Syn: *}              \p{Pattern_Syntax: *}
    \p{Pat_WS}                  \p{Pattern_White_Space} (=
                                \p{Pattern_White_Space=Y}) (11)
    \p{Pat_WS: *}               \p{Pattern_White_Space: *}
    \p{Pattern_Syntax}          \p{Pattern_Syntax=Y} (Short: \p{PatSyn})
                                (2760)
    \p{Pattern_Syntax: N*}  (Short: \p{PatSyn=N}, \P{PatSyn})
                                (1_111_352)
    \p{Pattern_Syntax: Y*}  (Short: \p{PatSyn=Y}, \p{PatSyn}) (2760)
    \p{Pattern_White_Space} \p{Pattern_White_Space=Y} (Short:
                                \p{PatWS}) (11)
    \p{Pattern_White_Space: N*} (Short: \p{PatWS=N}, \P{PatWS})
                                (1_114_101)
    \p{Pattern_White_Space: Y*} (Short: \p{PatWS=Y}, \p{PatWS}) (11)
    \p{Pc}                      \p{Connector_Punctuation} (=
                                \p{General_Category=
                                Connector_Punctuation}) (10)
    \p{Pd}                      \p{Dash_Punctuation} (=
                                \p{General_Category=Dash_Punctuation})
                                (21)
    \p{Pe}                      \p{Close_Punctuation} (=
                                \p{General_Category=Close_Punctuation})
                                (71)
    \p{PerlSpace}               \s, restricted to ASCII (5)
    \p{PerlWord}                \w, restricted to ASCII = [A-Za-z0-9_] (63)
    \p{Pf}                      \p{Final_Punctuation} (=
                                \p{General_Category=Final_Punctuation})
                                (10)
    \p{Phag}                    \p{Phags_Pa} (= \p{Script=Phags_Pa}) (NOT
                                \p{Block=Phags_Pa}) (56)
    \p{Phags_Pa}                \p{Script=Phags_Pa} (Short: \p{Phag}; NOT
                                \p{Block=Phags_Pa}) (56)
  X \p{Phaistos_Disc}           \p{Block=Phaistos_Disc} (48)
    \p{Phli}                    \p{Inscriptional_Pahlavi} (= \p{Script=
                                Inscriptional_Pahlavi}) (NOT \p{Block=
                                Inscriptional_Pahlavi}) (27)
    \p{Phnx}                    \p{Phoenician} (= \p{Script=Phoenician})
                                (NOT \p{Block=Phoenician}) (29)
    \p{Phoenician}              \p{Script=Phoenician} (Short: \p{Phnx};
                                NOT \p{Block=Phoenician}) (29)
  X \p{Phonetic_Extensions} \p{Block=Phonetic_Extensions} (128)
  X \p{Phonetic_Extensions_Supplement} \p{Block=
                                Phonetic_Extensions_Supplement} (64)
    \p{Pi}                      \p{Initial_Punctuation} (=
                                \p{General_Category=
                                Initial_Punctuation}) (12)
    \p{Po}                      \p{Other_Punctuation} (=
                                \p{General_Category=Other_Punctuation})
                                (389)
    \p{PosixAlnum}              [A-Za-z0-9] (62)
    \p{PosixAlpha}              [A-Za-z] (52)
    \p{PosixBlank}              \t and ' ' (2)
    \p{PosixCntrl}              [\x00-\x1F] (33)
```

```
  \p{PosixDigit}          [0-9] (10)
  \p{PosixGraph}          [\x21-\x7E] (94)
  \p{PosixLower}          [a-z] (26)
  \p{PosixPrint}          [\x20-\x7E] (95)
  \p{PosixPunct}          Graphical characters that aren't Word
                            characters = [\x21-\x2F\x3A-\x40\x5B-
                            \x60\x7B-\x7E] (32)
  \p{PosixSpace}          \t \n, \x0B, \f, \r, and ' ' (6)
  \p{PosixUpper}          [A-Z] (26)
T \p{Present_In: 1.1}     \p{Age=1.1} (Short: \p{In=1.1}) (Perl
                            extension) (33_979)
T \p{Present_In: 2.0}     Code point's usage introduced in version
                            2.0 or earlier (Short: \p{In=2.0}) (Perl
                            extension) (178_500)
T \p{Present_In: 2.1}     Code point's usage introduced in version
                            2.1 or earlier (Short: \p{In=2.1}) (Perl
                            extension) (178_502)
T \p{Present_In: 3.0}     Code point's usage introduced in version
                            3.0 or earlier (Short: \p{In=3.0}) (Perl
                            extension) (188_809)
T \p{Present_In: 3.1}     Code point's usage introduced in version
                            3.1 or earlier (Short: \p{In=3.1}) (Perl
                            extension) (233_787)
T \p{Present_In: 3.2}     Code point's usage introduced in version
                            3.2 or earlier (Short: \p{In=3.2}) (Perl
                            extension) (234_803)
T \p{Present_In: 4.0}     Code point's usage introduced in version
                            4.0 or earlier (Short: \p{In=4.0}) (Perl
                            extension) (236_029)
T \p{Present_In: 4.1}     Code point's usage introduced in version
                            4.1 or earlier (Short: \p{In=4.1}) (Perl
                            extension) (237_302)
T \p{Present_In: 5.0}     Code point's usage introduced in version
                            5.0 or earlier (Short: \p{In=5.0}) (Perl
                            extension) (238_671)
T \p{Present_In: 5.1}     Code point's usage introduced in version
                            5.1 or earlier (Short: \p{In=5.1}) (Perl
                            extension) (240_295)
T \p{Present_In: 5.2}     Code point's usage introduced in version
                            5.2 or earlier (Short: \p{In=5.2}) (Perl
                            extension) (246_943)
  \p{Present_In: Unassigned} \p{Age=Unassigned} (Short: \p{In=
                            Unassigned}) (Perl extension) (867_169)
  \p{Print}               Characters that are graphical plus space
                            characters (but no controls) (244_762)
  \p{Private_Use}         \p{General_Category=Private_Use} (Short:
                            \p{Co}; NOT \p{Private_Use_Area})
                            (137_468)
X \p{Private_Use_Area}    \p{Block=Private_Use_Area} (Short:
                            \p{InPrivateUse}) (6400)
  \p{Prti}                \p{Inscriptional_Parthian} (= \p{Script=
                            Inscriptional_Parthian}) (NOT \p{Block=
                            Inscriptional_Parthian}) (30)
  \p{Ps}                  \p{Open_Punctuation} (=
                            \p{General_Category=Open_Punctuation})
                            (72)
```

```
    \p{Punct}                \p{General_Category=Punctuation} (Short:
                               \p{P}) (585)
    \p{Punctuation}          \p{Punct} (= \p{General_Category=
                               Punctuation}) (585)
    \p{Qaac}                 \p{Coptic} (= \p{Script=Coptic}) (NOT
                               \p{Block=Coptic}) (135)
    \p{Qaai}                 \p{Inherited} (= \p{Script=Inherited})
                               (523)
    \p{QMark}                \p{Quotation_Mark} (= \p{Quotation_Mark=
                               Y}) (29)
    \p{QMark: *}             \p{Quotation_Mark: *}
    \p{Quotation_Mark}       \p{Quotation_Mark=Y} (Short: \p{QMark})
                               (29)
    \p{Quotation_Mark: N*}  (Short: \p{QMark=N}, \P{QMark}) (1_114_083)
    \p{Quotation_Mark: Y*}  (Short: \p{QMark=Y}, \p{QMark}) (29)
    \p{Radical}              \p{Radical=Y} (329)
    \p{Radical: N*}          (Single: \P{Radical}) (1_113_783)
    \p{Radical: Y*}          (Single: \p{Radical}) (329)
    \p{Rejang}               \p{Script=Rejang} (Short: \p{Rjng}; NOT
                               \p{Block=Rejang}) (37)
    \p{Rjng}                 \p{Rejang} (= \p{Script=Rejang}) (NOT
                               \p{Block=Rejang}) (37)
X \p{Rumi_Numeral_Symbols} \p{Block=Rumi_Numeral_Symbols} (32)
    \p{Runic}                \p{Script=Runic} (Short: \p{Runr}; NOT
                               \p{Block=Runic}) (78)
    \p{Runr}                 \p{Runic} (= \p{Script=Runic}) (NOT
                               \p{Block=Runic}) (78)
    \p{S}                    \p{Symbol} (= \p{General_Category=Symbol})
                               (4499)
    \p{Samaritan}            \p{Script=Samaritan} (Short: \p{Samr}; NOT
                               \p{Block=Samaritan}) (61)
    \p{Samr}                 \p{Samaritan} (= \p{Script=Samaritan})
                               (NOT \p{Block=Samaritan}) (61)
    \p{Sarb}                 \p{Old_South_Arabian} (= \p{Script=
                               Old_South_Arabian}) (32)
    \p{Saur}                 \p{Saurashtra} (= \p{Script=Saurashtra})
                               (NOT \p{Block=Saurashtra}) (81)
    \p{Saurashtra}           \p{Script=Saurashtra} (Short: \p{Saur};
                               NOT \p{Block=Saurashtra}) (81)
    \p{SB: *}                \p{Sentence_Break: *}
    \p{Sc}                   \p{Currency_Symbol} (=
                               \p{General_Category=Currency_Symbol})
                               (46)
    \p{Sc: *}                \p{Script: *}
    \p{Script: Arab}         \p{Script=Arabic} (1030)
    \p{Script: Arabic}       (Short: \p{Sc=Arab}, \p{Arab}) (1030)
    \p{Script: Armenian}     (Short: \p{Sc=Armn}, \p{Armn}) (90)
    \p{Script: Armi}         \p{Script=Imperial_Aramaic} (31)
    \p{Script: Armn}         \p{Script=Armenian} (90)
    \p{Script: Avestan}      (Short: \p{Sc=Avst}, \p{Avst}) (61)
    \p{Script: Avst}         \p{Script=Avestan} (61)
    \p{Script: Bali}         \p{Script=Balinese} (121)
    \p{Script: Balinese}     (Short: \p{Sc=Bali}, \p{Bali}) (121)
    \p{Script: Bamu}         \p{Script=Bamum} (88)
    \p{Script: Bamum}        (Short: \p{Sc=Bamu}, \p{Bamu}) (88)
    \p{Script: Beng}         \p{Script=Bengali} (92)
```

```
\p{Script: Bengali}     (Short: \p{Sc=Beng}, \p{Beng}) (92)
\p{Script: Bopo}        \p{Script=Bopomofo} (65)
\p{Script: Bopomofo}    (Short: \p{Sc=Bopo}, \p{Bopo}) (65)
\p{Script: Brai}        \p{Script=Braille} (256)
\p{Script: Braille}     (Short: \p{Sc=Brai}, \p{Brai}) (256)
\p{Script: Bugi}        \p{Script=Buginese} (30)
\p{Script: Buginese}    (Short: \p{Sc=Bugi}, \p{Bugi}) (30)
\p{Script: Buhd}        \p{Script=Buhid} (20)
\p{Script: Buhid}       (Short: \p{Sc=Buhd}, \p{Buhd}) (20)
\p{Script: Canadian_Aboriginal} (Short: \p{Sc=Cans}, \p{Cans})
                          (710)
\p{Script: Cans}        \p{Script=Canadian_Aboriginal} (710)
\p{Script: Cari}        \p{Script=Carian} (49)
\p{Script: Carian}      (Short: \p{Sc=Cari}, \p{Cari}) (49)
\p{Script: Cham}        (Short: \p{Sc=Cham}, \p{Cham}) (83)
\p{Script: Cher}        \p{Script=Cherokee} (85)
\p{Script: Cherokee}    (Short: \p{Sc=Cher}, \p{Cher}) (85)
\p{Script: Common}      (Short: \p{Sc=Zyyy}, \p{Zyyy}) (5395)
\p{Script: Copt}        \p{Script=Coptic} (135)
\p{Script: Coptic}      (Short: \p{Sc=Copt}, \p{Copt}) (135)
\p{Script: Cprt}        \p{Script=Cypriot} (55)
\p{Script: Cuneiform}   (Short: \p{Sc=Xsux}, \p{Xsux}) (982)
\p{Script: Cypriot}     (Short: \p{Sc=Cprt}, \p{Cprt}) (55)
\p{Script: Cyrillic}    (Short: \p{Sc=Cyrl}, \p{Cyrl}) (404)
\p{Script: Cyrl}        \p{Script=Cyrillic} (404)
\p{Script: Deseret}     (Short: \p{Sc=Dsrt}, \p{Dsrt}) (80)
\p{Script: Deva}        \p{Script=Devanagari} (140)
\p{Script: Devanagari}  (Short: \p{Sc=Deva}, \p{Deva}) (140)
\p{Script: Dsrt}        \p{Script=Deseret} (80)
\p{Script: Egyp}        \p{Script=Egyptian_Hieroglyphs} (1071)
\p{Script: Egyptian_Hieroglyphs} (Short: \p{Sc=Egyp}, \p{Egyp})
                          (1071)
\p{Script: Ethi}        \p{Script=Ethiopic} (461)
\p{Script: Ethiopic}    (Short: \p{Sc=Ethi}, \p{Ethi}) (461)
\p{Script: Geor}        \p{Script=Georgian} (120)
\p{Script: Georgian}    (Short: \p{Sc=Geor}, \p{Geor}) (120)
\p{Script: Glag}        \p{Script=Glagolitic} (94)
\p{Script: Glagolitic}  (Short: \p{Sc=Glag}, \p{Glag}) (94)
\p{Script: Goth}        \p{Script=Gothic} (27)
\p{Script: Gothic}      (Short: \p{Sc=Goth}, \p{Goth}) (27)
\p{Script: Greek}       (Short: \p{Sc=Grek}, \p{Grek}) (511)
\p{Script: Grek}        \p{Script=Greek} (511)
\p{Script: Gujarati}    (Short: \p{Sc=Gujr}, \p{Gujr}) (83)
\p{Script: Gujr}        \p{Script=Gujarati} (83)
\p{Script: Gurmukhi}    (Short: \p{Sc=Guru}, \p{Guru}) (79)
\p{Script: Guru}        \p{Script=Gurmukhi} (79)
\p{Script: Han}         (Short: \p{Sc=Han}, \p{Han}) (75_738)
\p{Script: Hang}        \p{Script=Hangul} (11_737)
\p{Script: Hangul}      (Short: \p{Sc=Hang}, \p{Hang}) (11_737)
\p{Script: Hani}        \p{Script=Han} (75_738)
\p{Script: Hano}        \p{Script=Hanunoo} (21)
\p{Script: Hanunoo}     (Short: \p{Sc=Hano}, \p{Hano}) (21)
\p{Script: Hebr}        \p{Script=Hebrew} (133)
\p{Script: Hebrew}      (Short: \p{Sc=Hebr}, \p{Hebr}) (133)
\p{Script: Hira}        \p{Script=Hiragana} (90)
\p{Script: Hiragana}    (Short: \p{Sc=Hira}, \p{Hira}) (90)
```

```
\p{Script: Imperial_Aramaic} (Short: \p{Sc=Armi}, \p{Armi}) (31)
\p{Script: Inherited}   (Short: \p{Sc=Zinh}, \p{Zinh}) (523)
\p{Script: Inscriptional_Pahlavi} (Short: \p{Sc=Phli}, \p{Phli})
                        (27)
\p{Script: Inscriptional_Parthian} (Short: \p{Sc=Prti}, \p{Prti})
                        (30)
\p{Script: Ital}        \p{Script=Old_Italic} (35)
\p{Script: Java}        \p{Script=Javanese} (91)
\p{Script: Javanese}    (Short: \p{Sc=Java}, \p{Java}) (91)
\p{Script: Kaithi}      (Short: \p{Sc=Kthi}, \p{Kthi}) (66)
\p{Script: Kali}        \p{Script=Kayah_Li} (48)
\p{Script: Kana}        \p{Script=Katakana} (299)
\p{Script: Kannada}     (Short: \p{Sc=Knda}, \p{Knda}) (84)
\p{Script: Katakana}    (Short: \p{Sc=Kana}, \p{Kana}) (299)
\p{Script: Kayah_Li}    (Short: \p{Sc=Kali}, \p{Kali}) (48)
\p{Script: Khar}        \p{Script=Kharoshthi} (65)
\p{Script: Kharoshthi}  (Short: \p{Sc=Khar}, \p{Khar}) (65)
\p{Script: Khmer}       (Short: \p{Sc=Khmr}, \p{Khmr}) (146)
\p{Script: Khmr}        \p{Script=Khmer} (146)
\p{Script: Knda}        \p{Script=Kannada} (84)
\p{Script: Kthi}        \p{Script=Kaithi} (66)
\p{Script: Lana}        \p{Script=Tai_Tham} (127)
\p{Script: Lao}         (Short: \p{Sc=Lao}, \p{Lao}) (65)
\p{Script: Laoo}        \p{Script=Lao} (65)
\p{Script: Latin}       (Short: \p{Sc=Latn}, \p{Latn}) (1244)
\p{Script: Latn}        \p{Script=Latin} (1244)
\p{Script: Lepc}        \p{Script=Lepcha} (74)
\p{Script: Lepcha}      (Short: \p{Sc=Lepc}, \p{Lepc}) (74)
\p{Script: Limb}        \p{Script=Limbu} (66)
\p{Script: Limbu}       (Short: \p{Sc=Limb}, \p{Limb}) (66)
\p{Script: Linb}        \p{Script=Linear_B} (211)
\p{Script: Linear_B}    (Short: \p{Sc=Linb}, \p{Linb}) (211)
\p{Script: Lisu}        (Short: \p{Sc=Lisu}, \p{Lisu}) (48)
\p{Script: Lyci}        \p{Script=Lycian} (29)
\p{Script: Lycian}      (Short: \p{Sc=Lyci}, \p{Lyci}) (29)
\p{Script: Lydi}        \p{Script=Lydian} (27)
\p{Script: Lydian}      (Short: \p{Sc=Lydi}, \p{Lydi}) (27)
\p{Script: Malayalam}   (Short: \p{Sc=Mlym}, \p{Mlym}) (95)
\p{Script: Meetei_Mayek} (Short: \p{Sc=Mtei}, \p{Mtei}) (56)
\p{Script: Mlym}        \p{Script=Malayalam} (95)
\p{Script: Mong}        \p{Script=Mongolian} (153)
\p{Script: Mongolian}   (Short: \p{Sc=Mong}, \p{Mong}) (153)
\p{Script: Mtei}        \p{Script=Meetei_Mayek} (56)
\p{Script: Myanmar}     (Short: \p{Sc=Mymr}, \p{Mymr}) (188)
\p{Script: Mymr}        \p{Script=Myanmar} (188)
\p{Script: New_Tai_Lue} (Short: \p{Sc=Talu}, \p{Talu}) (83)
\p{Script: Nko}         (Short: \p{Sc=Nko}, \p{Nko}) (59)
\p{Script: Nkoo}        \p{Script=Nko} (59)
\p{Script: Ogam}        \p{Script=Ogham} (29)
\p{Script: Ogham}       (Short: \p{Sc=Ogam}, \p{Ogam}) (29)
\p{Script: Ol_Chiki}    (Short: \p{Sc=Olck}, \p{Olck}) (48)
\p{Script: Olck}        \p{Script=Ol_Chiki} (48)
\p{Script: Old_Italic}  (Short: \p{Sc=Ital}, \p{Ital}) (35)
\p{Script: Old_Persian} (Short: \p{Sc=Xpeo}, \p{Xpeo}) (50)
\p{Script: Old_South_Arabian} (Short: \p{Sc=Sarb}, \p{Sarb}) (32)
\p{Script: Old_Turkic}  (Short: \p{Sc=Orkh}, \p{Orkh}) (73)
```

```
\p{Script: Oriya}       (Short: \p{Sc=Orya}, \p{Orya}) (84)
\p{Script: Orkh}        \p{Script=Old_Turkic} (73)
\p{Script: Orya}        \p{Script=Oriya} (84)
\p{Script: Osma}        \p{Script=Osmanya} (40)
\p{Script: Osmanya}     (Short: \p{Sc=Osma}, \p{Osma}) (40)
\p{Script: Phag}        \p{Script=Phags_Pa} (56)
\p{Script: Phags_Pa}    (Short: \p{Sc=Phag}, \p{Phag}) (56)
\p{Script: Phli}        \p{Script=Inscriptional_Pahlavi} (27)
\p{Script: Phnx}        \p{Script=Phoenician} (29)
\p{Script: Phoenician}  (Short: \p{Sc=Phnx}, \p{Phnx}) (29)
\p{Script: Prti}        \p{Script=Inscriptional_Parthian} (30)
\p{Script: Qaac}        \p{Script=Coptic} (135)
\p{Script: Qaai}        \p{Script=Inherited} (523)
\p{Script: Rejang}      (Short: \p{Sc=Rjng}, \p{Rjng}) (37)
\p{Script: Rjng}        \p{Script=Rejang} (37)
\p{Script: Runic}       (Short: \p{Sc=Runr}, \p{Runr}) (78)
\p{Script: Runr}        \p{Script=Runic} (78)
\p{Script: Samaritan}   (Short: \p{Sc=Samr}, \p{Samr}) (61)
\p{Script: Samr}        \p{Script=Samaritan} (61)
\p{Script: Sarb}        \p{Script=Old_South_Arabian} (32)
\p{Script: Saur}        \p{Script=Saurashtra} (81)
\p{Script: Saurashtra}  (Short: \p{Sc=Saur}, \p{Saur}) (81)
\p{Script: Shavian}     (Short: \p{Sc=Shaw}, \p{Shaw}) (48)
\p{Script: Shaw}        \p{Script=Shavian} (48)
\p{Script: Sinh}        \p{Script=Sinhala} (80)
\p{Script: Sinhala}     (Short: \p{Sc=Sinh}, \p{Sinh}) (80)
\p{Script: Sund}        \p{Script=Sundanese} (55)
\p{Script: Sundanese}   (Short: \p{Sc=Sund}, \p{Sund}) (55)
\p{Script: Sylo}        \p{Script=Syloti_Nagri} (44)
\p{Script: Syloti_Nagri} (Short: \p{Sc=Sylo}, \p{Sylo}) (44)
\p{Script: Syrc}        \p{Script=Syriac} (77)
\p{Script: Syriac}      (Short: \p{Sc=Syrc}, \p{Syrc}) (77)
\p{Script: Tagalog}     (Short: \p{Sc=Tglg}, \p{Tglg}) (20)
\p{Script: Tagb}        \p{Script=Tagbanwa} (18)
\p{Script: Tagbanwa}    (Short: \p{Sc=Tagb}, \p{Tagb}) (18)
\p{Script: Tai_Le}      (Short: \p{Sc=Tale}, \p{Tale}) (35)
\p{Script: Tai_Tham}    (Short: \p{Sc=Lana}, \p{Lana}) (127)
\p{Script: Tai_Viet}    (Short: \p{Sc=Tavt}, \p{Tavt}) (72)
\p{Script: Tale}        \p{Script=Tai_Le} (35)
\p{Script: Talu}        \p{Script=New_Tai_Lue} (83)
\p{Script: Tamil}       (Short: \p{Sc=Taml}, \p{Taml}) (72)
\p{Script: Taml}        \p{Script=Tamil} (72)
\p{Script: Tavt}        \p{Script=Tai_Viet} (72)
\p{Script: Telu}        \p{Script=Telugu} (93)
\p{Script: Telugu}      (Short: \p{Sc=Telu}, \p{Telu}) (93)
\p{Script: Tfng}        \p{Script=Tifinagh} (55)
\p{Script: Tglg}        \p{Script=Tagalog} (20)
\p{Script: Thaa}        \p{Script=Thaana} (50)
\p{Script: Thaana}      (Short: \p{Sc=Thaa}, \p{Thaa}) (50)
\p{Script: Thai}        (Short: \p{Sc=Thai}, \p{Thai}) (86)
\p{Script: Tibetan}     (Short: \p{Sc=Tibt}, \p{Tibt}) (201)
\p{Script: Tibt}        \p{Script=Tibetan} (201)
\p{Script: Tifinagh}    (Short: \p{Sc=Tfng}, \p{Tfng}) (55)
\p{Script: Ugar}        \p{Script=Ugaritic} (31)
\p{Script: Ugaritic}    (Short: \p{Sc=Ugar}, \p{Ugar}) (31)
\p{Script: Unknown}     (Short: \p{Sc=Zzzz}, \p{Zzzz}) (1_006_751)
```

```
    \p{Script: Vai}         (Short: \p{Sc=Vai}, \p{Vai}) (300)
    \p{Script: Vaii}        \p{Script=Vai} (300)
    \p{Script: Xpeo}        \p{Script=Old_Persian} (50)
    \p{Script: Xsux}        \p{Script=Cuneiform} (982)
    \p{Script: Yi}          (Short: \p{Sc=Yi}, \p{Yi}) (1220)
    \p{Script: Yiii}        \p{Script=Yi} (1220)
    \p{Script: Zinh}        \p{Script=Inherited} (523)
    \p{Script: Zyyy}        \p{Script=Common} (5395)
    \p{Script: Zzzz}        \p{Script=Unknown} (1_006_751)
    \p{SD}                  \p{Soft_Dotted} (= \p{Soft_Dotted=Y}) (46)
    \p{SD: *}               \p{Soft_Dotted: *}
    \p{Sentence_Break: AT}  \p{Sentence_Break=ATerm} (4)
    \p{Sentence_Break: ATerm} (Short: \p{SB=AT}) (4)
    \p{Sentence_Break: CL}  \p{Sentence_Break=Close} (177)
    \p{Sentence_Break: Close} (Short: \p{SB=CL}) (177)
    \p{Sentence_Break: CR}  (Short: \p{SB=CR}) (1)
    \p{Sentence_Break: EX}  \p{Sentence_Break=Extend} (1455)
    \p{Sentence_Break: Extend} (Short: \p{SB=EX}) (1455)
    \p{Sentence_Break: FO}  \p{Sentence_Break=Format} (138)
    \p{Sentence_Break: Format} (Short: \p{SB=FO}) (138)
    \p{Sentence_Break: LE}  \p{Sentence_Break=OLetter} (96_405)
    \p{Sentence_Break: LF}  (Short: \p{SB=LF}) (1)
    \p{Sentence_Break: LO}  \p{Sentence_Break=Lower} (1907)
    \p{Sentence_Break: Lower} (Short: \p{SB=LO}) (1907)
    \p{Sentence_Break: NU}  \p{Sentence_Break=Numeric} (403)
    \p{Sentence_Break: Numeric} (Short: \p{SB=NU}) (403)
    \p{Sentence_Break: OLetter} (Short: \p{SB=LE}) (96_405)
    \p{Sentence_Break: Other} (Short: \p{SB=XX}) (1_012_008)
    \p{Sentence_Break: SC}  \p{Sentence_Break=SContinue} (26)
    \p{Sentence_Break: SContinue} (Short: \p{SB=SC}) (26)
    \p{Sentence_Break: SE}  \p{Sentence_Break=Sep} (3)
    \p{Sentence_Break: Sep} (Short: \p{SB=SE}) (3)
    \p{Sentence_Break: Sp}  (Short: \p{SB=Sp}) (21)
    \p{Sentence_Break: ST}  \p{Sentence_Break=STerm} (63)
    \p{Sentence_Break: STerm} (Short: \p{SB=ST}) (63)
    \p{Sentence_Break: UP}  \p{Sentence_Break=Upper} (1500)
    \p{Sentence_Break: Upper} (Short: \p{SB=UP}) (1500)
    \p{Sentence_Break: XX}  \p{Sentence_Break=Other} (1_012_008)
    \p{Separator}           \p{General_Category=Separator} (Short:
                              \p{Z}) (20)
    \p{Shavian}             \p{Script=Shavian} (Short: \p{Shaw}) (48)
    \p{Shaw}                \p{Shavian} (= \p{Script=Shavian}) (48)
    \p{Sinh}                \p{Sinhala} (= \p{Script=Sinhala}) (NOT
                              \p{Block=Sinhala}) (80)
    \p{Sinhala}             \p{Script=Sinhala} (Short: \p{Sinh}; NOT
                              \p{Block=Sinhala}) (80)
    \p{Sk}                  \p{Modifier_Symbol} (=
                              \p{General_Category=Modifier_Symbol})
                              (99)
    \p{Sm}                  \p{Math_Symbol} (= \p{General_Category=
                              Math_Symbol}) (945)
  X \p{Small_Form_Variants} \p{Block=Small_Form_Variants} (32)
    \p{So}                  \p{Other_Symbol} (= \p{General_Category=
                              Other_Symbol}) (3409)
    \p{Soft_Dotted}         \p{Soft_Dotted=Y} (Short: \p{SD}) (46)
    \p{Soft_Dotted: N*}     (Short: \p{SD=N}, \P{SD}) (1_114_066)
```

```
  \p{Soft_Dotted: Y*}     (Short: \p{SD=Y}, \p{SD}) (46)
  \p{Space}               \p{White_Space=Y} \s including beyond
                            ASCII plus vertical tab (26)
  \p{Space: *}            \p{White_Space: *}
  \p{Space_Separator}     \p{General_Category=Space_Separator}
                            (Short: \p{Zs}) (18)
  \p{SpacePerl}           \s, including beyond ASCII (25)
  \p{Spacing_Mark}        \p{General_Category=Spacing_Mark} (Short:
                            \p{Mc}) (276)
X \p{Spacing_Modifier_Letters} \p{Block=Spacing_Modifier_Letters}
                            (80)
X \p{Specials}            \p{Block=Specials} (16)
  \p{STerm}               \p{STerm=Y} (66)
  \p{STerm: N*}           (Single: \P{STerm}) (1_114_046)
  \p{STerm: Y*}           (Single: \p{STerm}) (66)
  \p{Sund}                \p{Sundanese} (= \p{Script=Sundanese})
                            (NOT \p{Block=Sundanese}) (55)
  \p{Sundanese}           \p{Script=Sundanese} (Short: \p{Sund}; NOT
                            \p{Block=Sundanese}) (55)
X \p{Superscripts_And_Subscripts} \p{Block=
                            Superscripts_And_Subscripts} (48)
X \p{Supplemental_Arrows_A} \p{Block=Supplemental_Arrows_A} (16)
X \p{Supplemental_Arrows_B} \p{Block=Supplemental_Arrows_B} (128)
X \p{Supplemental_Mathematical_Operators} \p{Block=
                            Supplemental_Mathematical_Operators}
                            (256)
X \p{Supplemental_Punctuation} \p{Block=Supplemental_Punctuation}
                            (128)
X \p{Supplementary_Private_Use_Area_A} \p{Block=
                            Supplementary_Private_Use_Area_A}
                            (65_536)
X \p{Supplementary_Private_Use_Area_B} \p{Block=
                            Supplementary_Private_Use_Area_B}
                            (65_536)
  \p{Surrogate}           \p{General_Category=Surrogate} (Short:
                            \p{Cs}) (2048)
  \p{Sylo}                \p{Syloti_Nagri} (= \p{Script=
                            Syloti_Nagri}) (NOT \p{Block=
                            Syloti_Nagri}) (44)
  \p{Syloti_Nagri}        \p{Script=Syloti_Nagri} (Short: \p{Sylo};
                            NOT \p{Block=Syloti_Nagri}) (44)
  \p{Symbol}              \p{General_Category=Symbol} (Short: \p{S})
                            (4499)
  \p{Syrc}                \p{Syriac} (= \p{Script=Syriac}) (NOT
                            \p{Block=Syriac}) (77)
  \p{Syriac}              \p{Script=Syriac} (Short: \p{Syrc}; NOT
                            \p{Block=Syriac}) (77)
  \p{Tagalog}             \p{Script=Tagalog} (Short: \p{Tglg}; NOT
                            \p{Block=Tagalog}) (20)
  \p{Tagb}                \p{Tagbanwa} (= \p{Script=Tagbanwa}) (NOT
                            \p{Block=Tagbanwa}) (18)
  \p{Tagbanwa}            \p{Script=Tagbanwa} (Short: \p{Tagb}; NOT
                            \p{Block=Tagbanwa}) (18)
X \p{Tags}                \p{Block=Tags} (128)
  \p{Tai_Le}              \p{Script=Tai_Le} (Short: \p{Tale}; NOT
                            \p{Block=Tai_Le}) (35)
```

```
    \p{Tai_Tham}            \p{Script=Tai_Tham} (Short: \p{Lana}; NOT
                              \p{Block=Tai_Tham}) (127)
    \p{Tai_Viet}            \p{Script=Tai_Viet} (Short: \p{Tavt}; NOT
                              \p{Block=Tai_Viet}) (72)
  X \p{Tai_Xuan_Jing_Symbols} \p{Block=Tai_Xuan_Jing_Symbols} (96)
    \p{Tale}                \p{Tai_Le} (= \p{Script=Tai_Le}) (NOT
                              \p{Block=Tai_Le}) (35)
    \p{Talu}                \p{New_Tai_Lue} (= \p{Script=New_Tai_Lue})
                              (NOT \p{Block=New_Tai_Lue}) (83)
    \p{Tamil}               \p{Script=Tamil} (Short: \p{Taml}; NOT
                              \p{Block=Tamil}) (72)
    \p{Taml}                \p{Tamil} (= \p{Script=Tamil}) (NOT
                              \p{Block=Tamil}) (72)
    \p{Tavt}                \p{Tai_Viet} (= \p{Script=Tai_Viet}) (NOT
                              \p{Block=Tai_Viet}) (72)
    \p{Telu}                \p{Telugu} (= \p{Script=Telugu}) (NOT
                              \p{Block=Telugu}) (93)
    \p{Telugu}              \p{Script=Telugu} (Short: \p{Telu}; NOT
                              \p{Block=Telugu}) (93)
    \p{Term}                \p{Terminal_Punctuation} (=
                              \p{Terminal_Punctuation=Y}) (161)
    \p{Term: *}             \p{Terminal_Punctuation: *}
    \p{Terminal_Punctuation} \p{Terminal_Punctuation=Y} (Short:
                              \p{Term}) (161)
    \p{Terminal_Punctuation: N*} (Short: \p{Term=N}, \P{Term})
                              (1_113_951)
    \p{Terminal_Punctuation: Y*} (Short: \p{Term=Y}, \p{Term}) (161)
    \p{Tfng}                \p{Tifinagh} (= \p{Script=Tifinagh}) (NOT
                              \p{Block=Tifinagh}) (55)
    \p{Tglg}                \p{Tagalog} (= \p{Script=Tagalog}) (NOT
                              \p{Block=Tagalog}) (20)
    \p{Thaa}                \p{Thaana} (= \p{Script=Thaana}) (NOT
                              \p{Block=Thaana}) (50)
    \p{Thaana}              \p{Script=Thaana} (Short: \p{Thaa}; NOT
                              \p{Block=Thaana}) (50)
    \p{Thai}                \p{Script=Thai} (NOT \p{Block=Thai}) (86)
    \p{Tibetan}             \p{Script=Tibetan} (Short: \p{Tibt}; NOT
                              \p{Block=Tibetan}) (201)
    \p{Tibt}                \p{Tibetan} (= \p{Script=Tibetan}) (NOT
                              \p{Block=Tibetan}) (201)
    \p{Tifinagh}            \p{Script=Tifinagh} (Short: \p{Tfng}; NOT
                              \p{Block=Tifinagh}) (55)
    \p{Title}               \p{General_Category=Titlecase_Letter}
                              (Short: \p{Lt}) (31)
    \p{Titlecase_Letter}    \p{Title} (= \p{General_Category=
                              Titlecase_Letter}) (31)
    \p{Ugar}                \p{Ugaritic} (= \p{Script=Ugaritic}) (NOT
                              \p{Block=Ugaritic}) (31)
    \p{Ugaritic}            \p{Script=Ugaritic} (Short: \p{Ugar}; NOT
                              \p{Block=Ugaritic}) (31)
    \p{UIdeo}               \p{Unified_Ideograph} (=
                              \p{Unified_Ideograph=Y}) (74_394)
    \p{UIdeo: *}            \p{Unified_Ideograph: *}
    \p{Unassigned}          \p{General_Category=Unassigned} (Short:
                              \p{Cn}) (867_235)
  X \p{Unified_Canadian_Aboriginal_Syllabics} \p{Block=
```

```
                              Unified_Canadian_Aboriginal_Syllabics}
                              (Short: \p{InCanadianSyllabics}) (640)
X \p{Unified_Canadian_Aboriginal_Syllabics_Extended} \p{Block=
                              Unified_Canadian_Aboriginal_Syllabics_-
                              Extended} (80)
  \p{Unified_Ideograph}   \p{Unified_Ideograph=Y} (Short: \p{UIdeo})
                              (74_394)
  \p{Unified_Ideograph: N*} (Short: \p{UIdeo=N}, \P{UIdeo})
                              (1_039_718)
  \p{Unified_Ideograph: Y*} (Short: \p{UIdeo=Y}, \p{UIdeo}) (74_394)
  \p{Unknown}             \p{Script=Unknown} (Short: \p{Zzzz})
                              (1_006_751)
  \p{Upper}               \p{Uppercase=Y} (1469)
  \p{Upper: *}            \p{Uppercase: *}
  \p{Uppercase}           \p{Upper} (= \p{Uppercase=Y}) (1469)
  \p{Uppercase: N*}       (Short: \p{Upper=N}, \P{Upper}) (1_112_643)
  \p{Uppercase: Y*}       (Short: \p{Upper=Y}, \p{Upper}) (1469)
  \p{Uppercase_Letter}    \p{General_Category=Uppercase_Letter}
                              (Short: \p{Lu}) (1427)
  \p{Vai}                 \p{Script=Vai} (NOT \p{Block=Vai}) (300)
  \p{Vaii}                \p{Vai} (= \p{Script=Vai}) (NOT \p{Block=
                              Vai}) (300)
  \p{Variation_Selector}  \p{Variation_Selector=Y} (Short: \p{VS})
                              (259)
  \p{Variation_Selector: N*} (Short: \p{VS=N}, \P{VS}) (1_113_853)
  \p{Variation_Selector: Y*} (Short: \p{VS=Y}, \p{VS}) (259)
X \p{Variation_Selectors} \p{Block=Variation_Selectors} (16)
X \p{Variation_Selectors_Supplement} \p{Block=
                              Variation_Selectors_Supplement} (240)
X \p{Vedic_Extensions}    \p{Block=Vedic_Extensions} (48)
X \p{Vertical_Forms}      \p{Block=Vertical_Forms} (16)
  \p{VertSpace}           \v (7)
  \p{VS}                  \p{Variation_Selector} (=
                              \p{Variation_Selector=Y}) (259)
  \p{VS: *}               \p{Variation_Selector: *}
  \p{WB: *}               \p{Word_Break: *}
  \p{White_Space}         \p{White_Space=Y} (Short: \p{WSpace}) (26)
  \p{White_Space: N*}     (Short: \p{Space=N}, \P{WSpace})
                              (1_114_086)
  \p{White_Space: Y*}     (Short: \p{Space=Y}, \p{WSpace}) (26)
  \p{Word}                \w, including beyond ASCII (101_685)
  \p{Word_Break: ALetter} (Short: \p{WB=LE}) (23_694)
  \p{Word_Break: CR}      (Short: \p{WB=CR}) (1)
  \p{Word_Break: EX}      \p{Word_Break=ExtendNumLet} (10)
  \p{Word_Break: Extend}  (Short: \p{WB=Extend}) (1455)
  \p{Word_Break: ExtendNumLet} (Short: \p{WB=EX}) (10)
  \p{Word_Break: FO}      \p{Word_Break=Format} (137)
  \p{Word_Break: Format}  (Short: \p{WB=FO}) (137)
  \p{Word_Break: KA}      \p{Word_Break=Katakana} (309)
  \p{Word_Break: Katakana} (Short: \p{WB=KA}) (309)
  \p{Word_Break: LE}      \p{Word_Break=ALetter} (23_694)
  \p{Word_Break: LF}      (Short: \p{WB=LF}) (1)
  \p{Word_Break: MB}      \p{Word_Break=MidNumLet} (8)
  \p{Word_Break: MidLetter} (Short: \p{WB=ML}) (8)
  \p{Word_Break: MidNum}  (Short: \p{WB=MN}) (15)
  \p{Word_Break: MidNumLet} (Short: \p{WB=MB}) (8)
```

```
    \p{Word_Break: ML}      \p{Word_Break=MidLetter} (8)
    \p{Word_Break: MN}      \p{Word_Break=MidNum} (15)
    \p{Word_Break: Newline} (Short: \p{WB=NL}) (5)
    \p{Word_Break: NL}      \p{Word_Break=Newline} (5)
    \p{Word_Break: NU}      \p{Word_Break=Numeric} (402)
    \p{Word_Break: Numeric} (Short: \p{WB=NU}) (402)
    \p{Word_Break: Other}   (Short: \p{WB=XX}) (1_088_067)
    \p{Word_Break: XX}      \p{Word_Break=Other} (1_088_067)
    \p{WSpace}              \p{White_Space} (= \p{White_Space=Y}) (26)
    \p{WSpace: *}           \p{White_Space: *}
    \p{XDigit}              \p{Hex_Digit=Y} (Short: \p{Hex}) (44)
    \p{XID_Continue}        \p{XID_Continue=Y} (Short: \p{XIDC})
                              (101_615)
    \p{XID_Continue: N*}    (Short: \p{XIDC=N}, \P{XIDC}) (1_012_497)
    \p{XID_Continue: Y*}    (Short: \p{XIDC=Y}, \p{XIDC}) (101_615)
    \p{XID_Start}           \p{XID_Start=Y} (Short: \p{XIDS}) (99_741)
    \p{XID_Start: N*}       (Short: \p{XIDS=N}, \P{XIDS}) (1_014_371)
    \p{XID_Start: Y*}       (Short: \p{XIDS=Y}, \p{XIDS}) (99_741)
    \p{XIDC}                \p{XID_Continue} (= \p{XID_Continue=Y})
                              (101_615)
    \p{XIDC: *}             \p{XID_Continue: *}
    \p{XIDS}                \p{XID_Start} (= \p{XID_Start=Y}) (99_741)
    \p{XIDS: *}             \p{XID_Start: *}
    \p{Xpeo}                \p{Old_Persian} (= \p{Script=Old_Persian})
                              (NOT \p{Block=Old_Persian}) (50)
    \p{Xsux}                \p{Cuneiform} (= \p{Script=Cuneiform})
                              (NOT \p{Block=Cuneiform}) (982)
    \p{Yi}                  \p{Script=Yi} (1220)
X   \p{Yi_Radicals}         \p{Block=Yi_Radicals} (64)
X   \p{Yi_Syllables}        \p{Block=Yi_Syllables} (1168)
    \p{Yiii}                \p{Yi} (= \p{Script=Yi}) (1220)
X   \p{Yijing_Hexagram_Symbols} \p{Block=Yijing_Hexagram_Symbols} (64)
    \p{Z}                   \p{Separator} (= \p{General_Category=
                              Separator}) (20)
    \p{Zinh}                \p{Inherited} (= \p{Script=Inherited})
                              (523)
    \p{Zl}                  \p{Line_Separator} (= \p{General_Category=
                              Line_Separator}) (1)
    \p{Zp}                  \p{Paragraph_Separator} (=
                              \p{General_Category=
                              Paragraph_Separator}) (1)
    \p{Zs}                  \p{Space_Separator} (=
                              \p{General_Category=Space_Separator})
                              (18)
    \p{Zyyy}                \p{Common} (= \p{Script=Common}) (5395)
    \p{Zzzz}                \p{Unknown} (= \p{Script=Unknown})
                              (1_006_751)
T   \p{_CanonDCIJ}          (For internal use by Perl, not necessarily
                              stable) (= \p{Soft_Dotted=Y}) (46)
T   \p{_Case_Ignorable}     (For internal use by Perl, not necessarily
                              stable) (= \p{Case_Ignorable=Y}) (1632)
T   \p{_CombAbove}          (For internal use by Perl, not necessarily
                              stable) (= \p{Canonical_Combining_Class=
                              Above}) (318)
T   \p{_X_Begin}            (For internal use by Perl, not necessarily
                              stable) (1_113_907)
```

```
T \p{_X_Extend}          (For internal use by Perl, not necessarily
                           stable) (1462)
T \p{_X_LV_LVT_V}        (For internal use by Perl, not necessarily
                           stable) (11_267)
```

Legal \p{} and \P{} constructs that match no characters

Unicode has some property-value pairs that currently don't match anything. This happens generally either because they are obsolete, or for symmetry with other forms, but no language has yet been encoded that uses them. In this version of Unicode, the following match zero code points:

\p{Canonical_Combining_Class=Attached_Below_Left}

\p{Joining_Type=Left_Joining}

Properties not accessible through \p{} and \P{}

A few properties are accessible in Perl via various function calls only. These are: Lowercase_Mapping lc() and lcfirst() Titlecase_Mapping ucfirst() Uppercase_Mapping uc()

Case_Folding is accessible through the /i modifier in regular expressions.

The Name property is accessible through the \N{} interpolation in double-quoted strings and regular expressions, but both usages require a `use charnames;` to be specified, which also contains related functions viacode() and vianame().

Unicode regular expression properties that are NOT accepted by Perl

Perl will generate an error for a few character properties in Unicode when used in a regular expression. The non-Unihan ones are listed below, with the reasons they are not accepted, perhaps with work-arounds. The short names for the properties are listed enclosed in (parentheses).

Expands_On_NFC (XO_NFC)

Expands_On_NFD (XO_NFD)

Expands_On_NFKC (XO_NFKC)

Expands_On_NFKD (XO_NFKD)

Easily computed, and yet doesn't cover the common encoding forms (UTF-16/8)

Grapheme_Link (Gr_Link)

Deprecated by Unicode. Use ccc=vr (Canonical_Combining_Class=Virama) instead

Jamo_Short_Name (JSN)

Used by Unicode internally for generating other properties and not intended to be used stand-alone

Script=Katakana_Or_Hiragana (sc=Hrkt)

Obsolete. All code points previously matched by this have been moved to "Script=Common"

An installation can choose to allow any of these to be matched by changing the controlling lists

contained in the program `$Config{privlib}`/*unicore/lib/unicore/mktables* and then re-running *lib/unicore/mktables*. (`%Config` is available from the Config module).

Files in the To directory (for serious hackers only)

All Unicode properties are really mappings (in the mathematical sense) from code points to their respective values. As part of its build process, Perl constructs tables containing these mappings for all properties that it deals with. But only a few of these are written out into files. Those written out are in the directory `$Config{privlib}`/*unicore/To/* (%Config is available from the Config module).

Those ones written are ones needed by Perl internally during execution, or for which there is some demand, and those for which there is no access through the Perl core. Generally, properties that can be used in regular expression matching do not have their map tables written, like Script. Nor are the simplistic properties that have a better, more complete version, such as Simple_Uppercase_Mapping (Uppercase_Mapping is written instead).

None of the properties in the *To* directory are currently directly accessible through the Perl core, although some may be accessed indirectly. For example, the uc() function implements the Uppercase_Mapping property and uses the *Upper.pl* file found in this directory.

The available files with their properties (short names in parentheses), and any flags or comments about them, are:

```
Bmg.pl          Bidi_Mirroring_Glyph (bmg)
Digit.pl        Perl_Decimal_Digit
Fold.pl         Case_Folding (cf)
Lower.pl        Lowercase_Mapping (lc)
NFKCCF.pl       NFKC_Casefold (NFKC_CF)
Title.pl        Titlecase_Mapping (tc)
Upper.pl        Uppercase_Mapping (uc)
```

An installation can choose to change which files are generated by changing the controlling lists contained in the program `$Config{privlib}`/*unicore/lib/unicore/mktables* and then re-running *lib/unicore/mktables*.

Each of these files defines two hash entries to help reading programs decipher it. One of them looks like this:

```
$utf8::SwashInfo{'ToNAME'}{'format'} = 's';
```

where 'NAME' is a name to indicate the property. For backwards compatibility, this is not necessarily the property's official Unicode name. (The 'To' is also for backwards compatibility.) The hash entry gives the format of the mapping fields of the table, currently one of the following:

```
b    binary
d    single decimal digit
f    floating point number
i    integer
r    rational: an integer or a fraction
s    arbitrary string
x    positive hex whole number; a code point
```

This format applies only to the entries in the main body of the table. Entries defined in hashes or ones that are missing from the list can have a different format.

The value that the missing entries have is given by the other SwashInfo hash entry line; it looks like this:

```
$utf8::SwashInfo{'ToNAME'}{'missing'} = 'NaN';
```

This example line says that any Unicode code points not explicitly listed in the file have the value 'NaN' under the property indicated by NAME. If the value is the special string `<code point>`, it means that the value for any missing code point is the code point itself. This happens, for example, in the file for Uppercase_Mapping (To/Upper.pl), in which code points like the character 'A', are missing because the uppercase of 'A' is itself.

SEE ALSO

http://www.unicode.org/reports/tr44/

perlrecharclass

perlunicode