Chapter 3 – Unicode – boolean – identifiers

12/6/22

3.3.4

code point = code value associated with a character

Unicode

Unicode has 17 code planes. 1st plane is Basic Multilingual Plane. Other 16 planes hold "supplementary characters".

BMP w/ surrogates area ⇒ supplementary characters

BMP = U+0000 to U+FFFF

Supp = U+10000 to U+10FFFF

Each character in BMP represented by a single 16-bit code unit (char)

surrogates: BMP has a range of unused values called the surrogates area:

See encoding details for UTF-16

0xD800 to 0xDBFF, 0xDC00 to 0xDFFF (2048 values)

1st code unit — 2nd code unit

supplementary characters are created by combining surrogates from each subrange as shown above

Benefit

Easy to tell if value encodes character in a single code unit or part of a supplementary character (and which surrogate)

Advice

Avoid working with chars. Work w/ code points.

3.3.5

to bool ⇔ int conversion

boolean: false or true

cannot convert to integer & vice versa (not directly)

3.4

identifiers { letters, digits ← cannot start w/ digit, currency symbols, punctuation connectors }

ANY LANGUAGE, eg. π is a letter

identifier restrictions

Don't use $ ← intended for compiler-generated names

Use Character.isJavaIdentifierStart() & isJavaIdentifierPart() to determine what's allowed

can't use Java keywords as names.

can't use a single underscore '_' as a name.

Write a program to take an int in the range 0x10000 to 0x10ffff and return the high & low surrogate pairs to encode it in UTF-16. Iterate over the range & print the character, the code point value in hex, and the high & low surrogate values in hex. Iterate over all BMP & supplementary characters & determine if they can be used in Java identifiers (& whether they can be used at the beginning of the identifier).