Swift Standard Library Type Aliases

This chapter describes the global type aliases defined in the Swift standard library.

Language

Swift

Symbols

Type Aliases

`AnyClass`

The protocol to which all class types implicitly conform.

`ArrayLiteralConvertible`

`BooleanLiteralConvertible`

`BooleanLiteralType`

The default type for an otherwise-unconstrained Boolean literal.

`CBool`

The C '_Bool' and C++ 'bool' type.

`CChar`

The C 'char' type.

`CChar16`

The C++11 'char16_t' type, which has UTF-16 encoding.

`CChar32`

The C++11 'char32_t' type, which has UTF-32 encoding.

`CDouble`

The C 'double' type.

`CFloat`

The C 'float' type.

`CInt`

The C 'int' type.

`CLong`

The C 'long' type.

`CLongLong`

The C ‘long long’ type.

`CShort`

The C ‘short’ type.

`CSignedChar`

The C ‘signed char’ type.

`CUnsignedChar`

The C ‘unsigned char’ type.

`CUnsignedInt`

The C ‘unsigned int’ type.

`CUnsignedLong`

The C ‘unsigned long’ type.

`CUnsignedLongLong`

The C ‘unsigned long long’ type.

`CUnsignedShort`

The C ‘unsigned short’ type.

`CWideChar`

The C++ ‘wchar_t’ type.

`DictionaryLiteralConvertible`

`ExtendedGraphemeClusterLiteralConvertible`

`ExtendedGraphemeClusterType`

The default type for an otherwise-unconstrained Unicode extended grapheme cluster literal.

`Float32`

A 32-bit floating point type.

`Float64`

A 64-bit floating point type.

`FloatLiteralConvertible`

`FloatLiteralType`

The default type for an otherwise-unconstrained floating point literal.

`IntMax`

The largest native signed integer type.

`IntegerLiteralConvertible`

`IntegerLiteralType`

The default type for an otherwise-unconstrained integer literal.

`NilLiteralConvertible`

`StringInterpolationConvertible`

`StringLiteralConvertible`

`StringLiteralType`

The default type for an otherwise-unconstrained string literal.

`UIntMax`

The largest native unsigned integer type.

`UnfoldFirstSequence`

The return type of `sequence(first:next:)`.

`UnicodeScalarLiteralConvertible`

`UnicodeScalarType`

The default type for an otherwise-unconstrained unicode scalar literal.

`Void`

The return type of functions that don't explicitly specify a return type; an empty tuple (i.e., `()`).