Swift

About the Language Reference

This part of the book describes the formal grammar of the Swift programming language. The grammar described here is intended to help you understand the language in more detail, rather than to allow you to directly implement a parser or compiler.

The Swift language is relatively small, because many common types, functions, and operators that appear virtually everywhere in Swift code are actually defined in the Swift standard library. Although these types, functions, and operators are not part of the Swift language itself, they are used extensively in the discussions and code examples in this part of the book.

How to Read the Grammar

The notation used to describe the formal grammar of the Swift programming language follows a few conventions:

- An arrow (→) is used to mark grammar productions and can be read as “can consist of.”
- Syntactic categories are indicated by *italic* text and appear on both sides of a grammar production rule.
- Literal words and punctuation are indicated by boldface `constant width` text and appear only on the right-hand side of a grammar production rule.
- Alternative grammar productions are separated by vertical bars (|). When alternative productions are too long to read easily, they are broken into

multiple grammar production rules on new lines.

- In a few cases, regular font text is used to describe the right-hand side of a grammar production rule.
- Optional syntactic categories and literals are marked by a trailing subscript, *opt*.

As an example, the grammar of a getter-setter block is defined as follows:

> GRAMMAR OF A GETTER-SETTER BLOCK
>
> *getter-setter-block* → **{** getter-clause setter-clause$_{opt}$ **}** | **{** setter-clause getter-clause **}**

This definition indicates that a getter-setter block can consist of a getter clause followed by an optional setter clause, enclosed in braces, *or* a setter clause followed by a getter clause, enclosed in braces. The grammar production above is equivalent to the following two productions, where the alternatives are spelled out explicitly:

> GRAMMAR OF A GETTER-SETTER BLOCK
>
> *getter-setter-block* → **{** getter-clause setter-clause$_{opt}$ **}**
> *getter-setter-block* → **{** setter-clause getter-clause **}**

‹ Advanced Operators Lexical Structure ›