

[Scala 3 Reference](#) / [Other Changed Features](#) / [Imports](#)**LEARN**

INSTALL

PLAYGROUND

FIND A LIBRARY

COMMUNITY

BLOG

Imports

[✎ Edit this page on GitHub](#)

The syntax of wildcard and renaming imports (and exports) has changed.

Wildcard Imports

Wildcard imports are now expressed with `*` instead of underscore. Example:

```
import scala.annotation.* // imports everything in the annotation package
```

If you want to import a member named `*` specifically, you can use backticks around it.

```
object A:
  def * = ...
  def min = ...

object B:
  import A.`*` // imports just `*`

object C:
  import A.* // imports everything in A
```

Renaming Imports

To rename or exclude an import, we now use `as` instead of `⇒`. A single renaming import no longer needs to be enclosed in braces. Examples:

```
import A.{min as minimum, `*` as multiply}
import Predef.{augmentString as _, *} // imports everything except augmentString
import scala.annotation as ann
import java as j
```

Migration



To support cross-building, Scala 3.0 supports the old import syntax with `_` for wildcards and `⇒` for renamings in addition to the new one. The old syntax will be dropped in a future versions. Automatic rewritings from old to new syntax are offered under settings `-source 3.1-migration -rewrite .`

Syntax

```
Import      ::= 'import' ImportExpr {',' ImportExpr}
ImportExpr  ::= SimpleRef {'.' id} '.' ImportSpec
ImportSpec  ::= NamedSelector
              | WildcardSelector
              | '{' ImportSelectors '}'
NamedSelector ::= id ['as' (id | '_')]
WildcardSelector ::= '*' | 'given' [InfixType]
ImportSelectors ::= NamedSelector [',' ImportSelectors]
                  | WildCardSelector {',' WildCardSelector}
```

[< Wildca...](#)
[Chang... >](#)