



TOUR OF SCALA

PACKAGES AND IMPORTS

Packages and Imports

Scala uses packages to create namespaces which allow you to modularize programs.

Creating a package

Packages are created by declaring one or more package names at the top of a Scala file.

```
package users

class User
```

One convention is to name the package the same as the directory containing the Scala file. However, Scala is agnostic to file layout. The directory structure of an sbt project for `package users` might look like this:

```
- ExampleProject
- build.sbt
- project
- src
  - main
    - scala
      - users
        User.scala
        UserProfile.scala
        UserPreferences.scala
  - test
```

Notice how the `users` directory is within the `scala` directory and how there are multiple Scala files within the package. Each Scala file in the package could have the same package declaration. The other way to declare packages is by using braces:

```
package users {
  package administrators {
    class NormalUser
  }
  package normalusers {
    class NormalUser
  }
}
```

As you can see, this allows for package nesting and provides greater control for scope and encapsulation.

The package name should be all lower case and if the code is being developed within an organization which has a website, it should be the following format convention: `<top-level-domain>.<domain-name>.<project-name>` . For example, if Google had a project called `SelfDrivingCar` , the package name would look like this:

```
package com.google.selfdrivingcar.camera
```

```
class Lens
```

This could correspond to the following directory structure:

```
SelfDrivingCar/src/main/scala/com/google/selfdrivingcar/camera/Lens.scala .
```

Imports

`import` clauses are for accessing members (classes, traits, functions, etc.) in other packages. An `import` clause is not required for accessing members of the same package. Import clauses are selective:

```
import users._ // import everything from the users package
import users.User // import the class User
import users.{User, UserPreferences} // Only imports selected members
import users.{UserPreferences => UPrefs} // import and rename for convenience
```

One way in which Scala is different from Java is that imports can be used anywhere:

```
def sqrtplus1(x: Int) = {
  import scala.math.sqrt
  sqrt(x) + 1.0
}
```

In the event there is a naming conflict and you need to import something from the root of the project, prefix the package name with `_root_`:

```
package accounts

import _root_.users._
```

Note: The `scala` and `java.lang` packages as well as `object Predef` are imported by default.

[← previous](#)

[next →](#)

Contributors to this page:

 ckipp01

 mlachkar

 ashawley

 SethTisue

 julienroubieu

 coryveilleux

DOCUMENTATION

- Getting Started
- API
- Overviews/Guides
- Language Specification

DOWNLOAD

- Current Version
- All versions

COMMUNITY

- Community
- Mailing Lists
- Chat Rooms & More
- Libraries and Tools
- The Scala Center

CONTRIBUTE

- How to help
- Report an Issue

SCALA

- Blog
- Code of Conduct

SOCIAL

- GitHub
- Twitter

