

[← Kotlin Eclipse 环境搭建](#)[Kotlin Android 环境搭建 →](#)

Kotlin 使用命令行编译

Kotlin 命令行编译工具下载地址：<https://github.com/JetBrains/kotlin/releases/tag/v1.1.2-2>，目前最新为 1.1.2-2。

你可以选择一个最新的稳定版下载。

下载完成后，解压到指定目录，然后将 **bin** 目录添加到系统环境变量。bin 目录包含编译和运行 Kotlin 所需的脚本。

SDKMAN!

在 OS X、Linux、Cygwin、FreeBSD 和 Solaris 系统上也可以使用更简单的安装方法，命令如下：

```
$ curl -s https://get.sdkman.io | bash

$ sdk install kotlin
```

Homebrew

在 OS X 下，你可以使用 Homebrew 安装：

```
$ brew update
$ brew install kotlin
```

MacPorts

如果你是 MacPorts 用户，可以使用以下命令安装：

```
$ sudo port install kotlin
```

创建和运行第一个程序

创建一个名为 hello.kt 文件，代码如下：

```
hello.kt

fun main(args: Array<String>) {
    println("Hello, World!")
}
```

使用 Kotlin 编译器编译应用：

```
$ kotlinc hello.kt -include-runtime -d hello.jar
```

- **-d**: 用来设置编译输出的名称，可以是 class 或 .jar 文件，也可以是目录。
- **-include-runtime** : 让 .jar 文件包含 Kotlin 运行库，从而可以直接运行。

如果你想看所有的可用选项，运行：

```
$ kotlinc -help
```

运行应用

```
$ java -jar hello.jar  
Hello, World!
```

编译成库

若需要将生成的 jar 包供其他 Kotlin 程序使用，可无需包含 Kotlin 的运行库：

```
$ kotlinc hello.kt -d hello.jar
```

由于这样生成的 .jar 文件不包含 Kotlin 运行库，所以你应该确保当它被使用时，运行时在你的 classpath 上。

你也可以使用 kotlin 命令来运行 Kotlin 编译器生成的 .jar 文件

```
$ kotlin -classpath hello.jar HelloKt
```

HelloKt 为编译器为 hello.kt 文件生成的默认类名。

运行 REPL（交互式解释器）

我们可以运行如下命令得到一个可交互的 shell，然后输入任何有效的 Kotlin 代码，并立即看到结果

```
[Ocean] ~/tutorials/kotlin/command_line/kotlinc$ bin/kotlinc-jvm  
Kotlin interactive shell  
Type :help for help, :quit for quit  
>>> 2+2  
4  
>>> println("Welcome to the Kotlin Shell")  
Welcome to the Kotlin Shell  
>>>
```

使用命令行执行脚本

Kotlin 也可以作为一个脚本语言使用，文件后缀名为 .kts。

例如我们创建一个名为 list_folders.kts，代码如下：

```
import java.io.File  
  
val folders = File(args[0]).listFiles { file -> file.isDirectory() }  
folders?.forEach { folder -> println(folder) }
```

执行时通过 `-script` 选项设置相应的脚本文件。

```
$ kotlinc -script list_folders.kts <path_to_folder>
```

```
$ kotlinc -script list_folders.kts
```

[← Kotlin Eclipse 环境搭建](#)[Kotlin Android 环境搭建 →](#)[✎ 点我分享笔记](#)