

How to Install Kotlin on Linux?

Last Updated : 29 Oct, 2022

Kotlin is a programming language introduced by JetBrains in 2011, the official designer of the most intelligent Java IDE, named IntelliJ IDEA. This is a strongly statically typed general-purpose programming language that runs on JVM. In 2017, Kotlin is sponsored by Google, announced as one of the official languages for Android Development.

In this article, we will learn How can we install the Kotlin programming language in Linux.

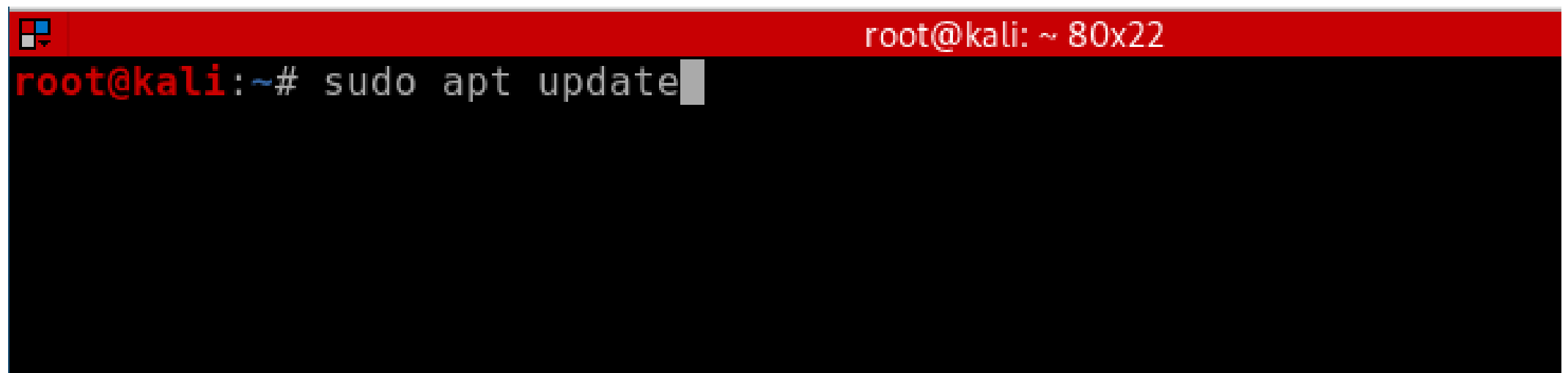
Installing Kotlin on Linux:

Follow the below steps to install Kotlin on your Linux system.

Step 1: Upgrade Packages

Open your Linux terminal and update your package information from all configured sources.

```
# sudo apt update
```



Step 2: Install JDK

Install the Java Software Development Kit (JDK). Kotlin is based on JDK. JDK is used to compile Kotlin programs. When you run the following command you will be asked to enter **Y** or **N** enter **Y** and click enter.

```
# sudo apt install default-jdk
```



```
root@kali: ~ 80x22
root@kali:~# sudo apt install default-jdk
```

Step 3: Install Kotlin using SDKMAN

SDKMAN is an easier way to install Kotlin on UNIX-based systems. After Installing SDKMAN restart your computer.

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



```
root@kali: ~ 81x23
root@kali:~# curl -s https://get.sdkman.io | bash
```

Now we will install Kotlin using the following command.

```
# sdk install kotlin
```



```
root@kali: ~ 82x25
root@kali:~# sdk install kotlin
```

Try: `$ kotlin -version;` And: `$ kotlin -help;`

After executing the above command Kotlin will be installed in your system. Now we will create a Kotlin file and check it by doing compile and run.

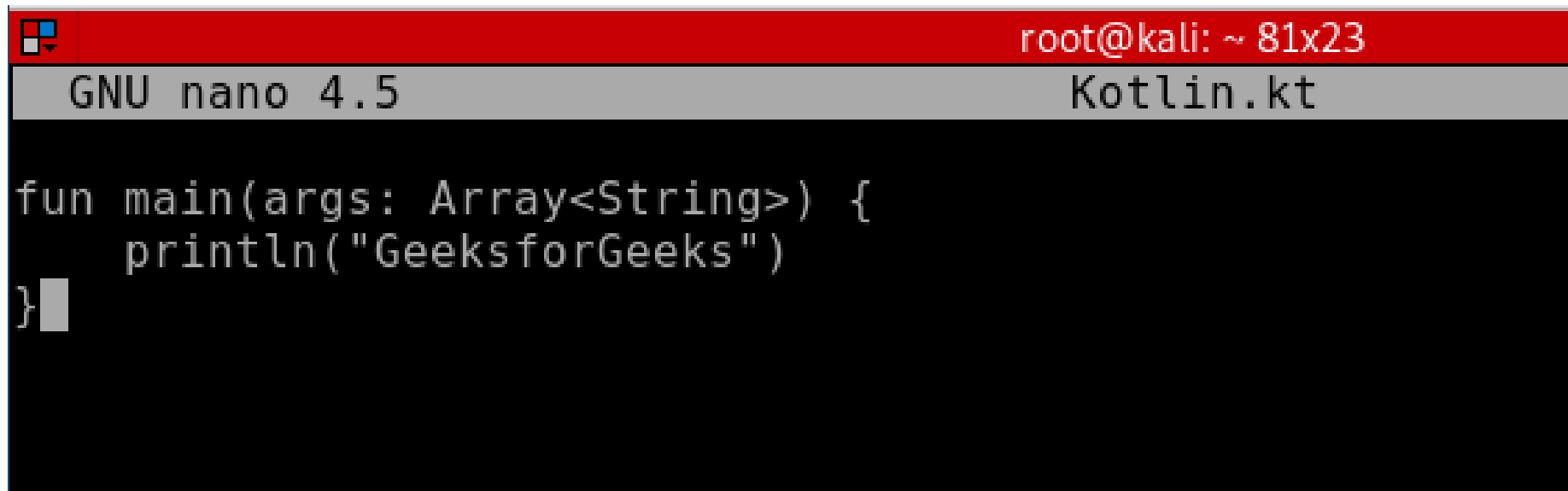
Step 4: Create a Kotlin file

Open any editor and write the following code there and save that file with the .kt extension. Here I'm creating a file with Kotlin.kt name.



```
root@kali: ~ 81x23
root@kali:~# nano Kotlin.kt
```

For creating a file write **nano fileName.kt** in your terminal. Then write your Kotlin code inside that file and save it using ctrl + x and then press y.



```
root@kali: ~ 81x23
GNU nano 4.5 Kotlin.kt

fun main(args: Array<String>) {
    println("GeeksforGeeks")
}
```

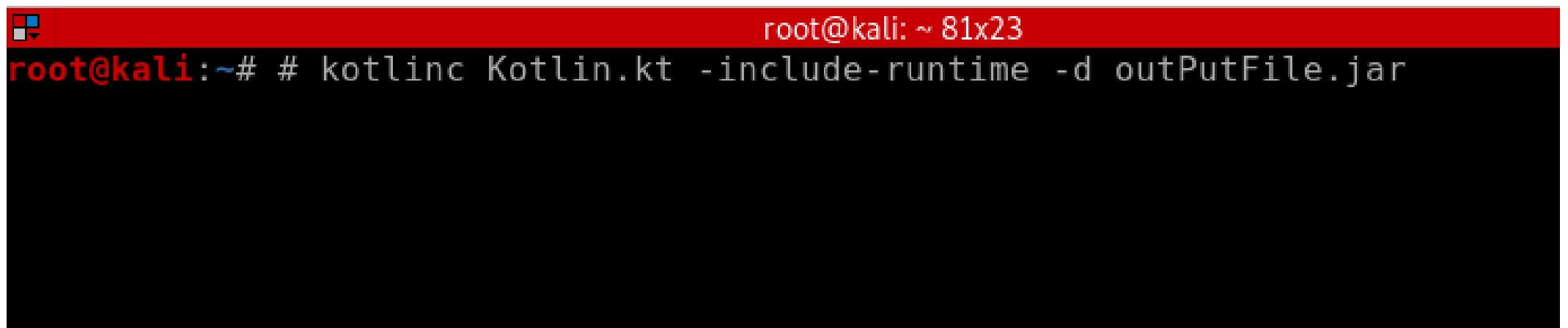
```
// Function in Kotlin
fun main(args: Array<String>) {

    // Print statement
    println("GeeksforGeeks")
}
```

Step 5: Compile Kotlin file

Navigate to that folder in which you have created your Kotlin file and then compile that file using the following command. Replace **Kotlin.kt** with your file name and The value of the **-d** option specifies the name of the output file.

```
# kotlinc Kotlin.kt -include-runtime -d outPutFile.jar
```

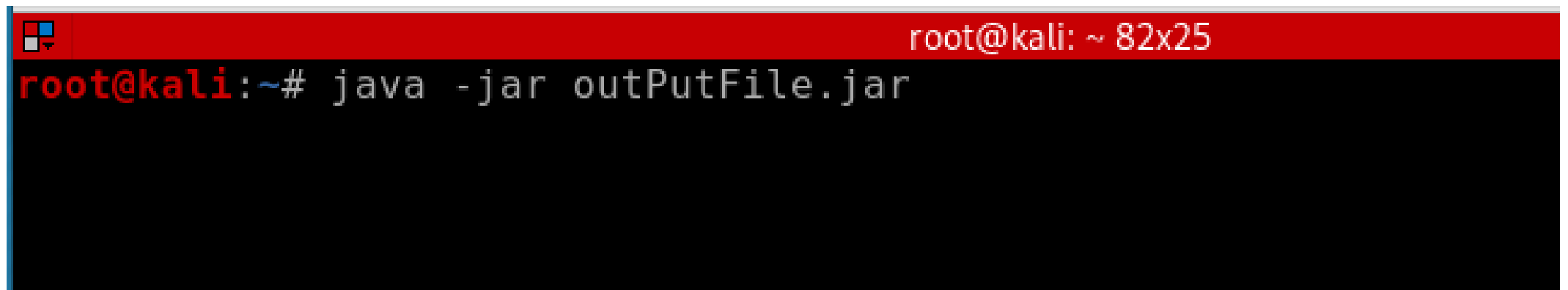


```
root@kali: ~ 81x23
root@kali:~# # kotlinc Kotlin.kt -include-runtime -d outPutFile.jar
```

Step 6: Run the Kotlin Program

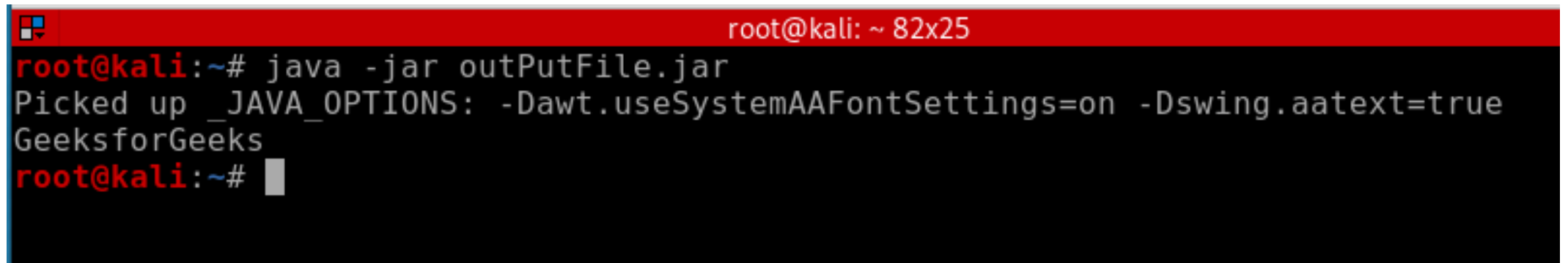
Now we will check the output by running the following command.

```
# java -jar outPutFile.jar
```

A terminal window with a red title bar containing the text 'root@kali: ~ 82x25'. The terminal content shows the command 'root@kali:~# java -jar outPutFile.jar' entered at the prompt. The background is black, and the text is white and red.

Output:

```
GeeksforGeeks
```

A terminal window with a red title bar containing the text 'root@kali: ~ 82x25'. The terminal content shows the command 'root@kali:~# java -jar outPutFile.jar' followed by the output 'Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true' and 'GeeksforGeeks'. The prompt 'root@kali:~#' is shown again with a cursor. The background is black, and the text is white and red.