Kotlin – Installation Procedure

Tushar B. Kute,
http://tusharkute.com

Installing Kotlin

- Installing Kotlin on Ubuntu is quite straightforward. Since Kotlin runs on the Java Virtual Machine (JVM), you'll first need to ensure Java is installed.
- After that, you can install the Kotlin compiler using a few different methods.

Installing Kotlin: Linux / Ubuntu

- **Step 1: Install Java Development Kit (JDK)**
- Kotlin requires a Java Development Kit (JDK) to run.
- Update package lists:
  - `sudo apt update`
  - `sudo apt upgrade -y`
- Install the default JDK:
  - `sudo apt install default-jdk -y`
- Verify Java installation:
  - `java -version`
- You should see output similar to this, indicating your Java version:
  - `openjdk version "11.0.12" 2021-07-20`
  - `OpenJDK Runtime Environment (build 11.0.12+7-Ubuntu-0ubuntu220.04)`
  - `OpenJDK 64-Bit Server VM (build 11.0.12+7-Ubuntu-0ubuntu220.04, mixed mode, sharing)`

Installing Kotlin

- **Step 2: Install the Kotlin Compiler**
- Snap is a universal packaging system developed by Canonical (the creators of Ubuntu). It's a very simple way to install applications.
- Ensure snapd is installed (usually pre-installed on modern Ubuntu):
  - `sudo apt update`
  - `sudo apt install snapd -y`
- Install Kotlin via Snap:
  - `sudo snap install --classic kotlin`
- The --classic flag is needed because Kotlin requires broader system access.
- Verify Kotlin installation:
  - `kotlinc -version`

Installing Kotlin on Windows

- **Step 1: Install Java Development Kit (JDK)**
- Kotlin runs on the Java Virtual Machine (JVM), so a JDK is a prerequisite.
  - Download JDK:
  - Go to the official Oracle JDK download page or OpenJDK distribution sites (like Adoptium, Amazon Corretto, etc.). Download the installer for Windows (e.g., .exe or .msi).
  - Oracle JDK: https://www.oracle.com/java/technologies/downloads/
  - Adoptium (OpenJDK): https://adoptium.net/temurin/releases/
- Run the Installer:
- Execute the downloaded installer and follow the on-screen prompts. It's generally recommended to install it in the default location.

Installing Kotlin on Windows

- Set Environment Variables (if not set by installer):
  - JAVA_HOME:
    - Search for "Environment Variables" in the Windows search bar and select "Edit the system environment variables."
    - Click "Environment Variables..."
    - Under "System variables," click "New..."
    - Variable name: JAVA_HOME
    - Variable value: The path to your JDK installation directory (e.g., C:\Program Files\Java\jdk-17 or C:\Program Files\Eclipse Adoptium\jdk-17.0.x.y-hotspot).
    - Click "OK."
  - Add to Path:
    - In the "System variables" section, find the Path variable and click "Edit..."
    - Click "New" and add %JAVA_HOME%\bin.
    - Click "OK" on all open windows.

Installing Kotlin on Windows

- Step 2: Install the Kotlin Command-Line Compiler : Manual Download
- This is a straightforward way to get the Kotlin compiler.
  - Download Kotlin Compiler:
  - Go to the official Kotlin GitHub releases page: https://github.com/JetBrains/kotlin/releases
  - Find the latest stable release and download the kotlin-compiler-<version>.zip file.
  - Extract the Archive:
  - Extract the contents of the downloaded ZIP file to a convenient location, for example, C:\kotlin\kotlinc.
  - Add Kotlin to your System Path:
    - Search for "Environment Variables" in the Windows search bar and select "Edit the system environment variables."
    - Click "Environment Variables..."
    - Under "System variables," find the Path variable and click "Edit..."
    - Click "New" and add the path to the bin directory inside your extracted Kotlin compiler folder (e.g., C:\kotlin\kotlinc\bin).
    - Click "OK" on all open windows.

Testing Kotlin

- Now that Kotlin is installed, you can try running a simple "Hello, World!" program.
  - Create a Kotlin file:
  - `nano hello.kt`
- Add the following code:

```
fun main() {
    println("Hello, World from Kotlin on Ubuntu!")
}
```

- Save and exit the file.

Testing Kotlin

- Compile the Kotlin file:
  - `kotlinc hello.kt -include-runtime -d hello.jar`
- This compiles your Kotlin code into a runnable JAR file, including the Kotlin runtime.
- Run the compiled JAR:
  - `java -jar hello.jar`
- You should see the output:
  - `Hello, World from Kotlin on Ubuntu!`

Testing Kotlin on Windows

- Open Command Prompt/PowerShell:
- Navigate to the directory where you saved hello.kt:
  - `cd C:\Users\YourUser\KotlinProjects`
- Compile the Kotlin file:
  - `kotlinc hello.kt -include-runtime -d hello.jar`
- This command compiles your Kotlin source file (hello.kt) into a runnable JAR file named hello.jar. The -include-runtime flag bundles the Kotlin runtime library into the JAR, making it self-contained.

Testing Kotlin on Windows

- Run the compiled JAR:
  - `java -jar hello.jar`
- You should see the output:
  - `Hello, World from Kotlin on Windows!`

Set Up an IDE

- **IntelliJ IDEA:** IntelliJ IDEA, developed by JetBrains (the creators of Kotlin), offers the best support for Kotlin development. The Kotlin plugin is bundled and enabled by default.
  - Download IntelliJ IDEA Community Edition: https://www.jetbrains.com/idea/download/
  - Run the installer and follow the prompts. Once installed, you can create a new Kotlin project directly from the IDE's welcome screen.
- **Android Studio:** If you plan to develop Android applications, Android Studio is the official IDE and also comes with excellent Kotlin support.
  - Download Android Studio: https://developer.android.com/studio
  - Run the installer and follow the prompts.

Thank you

@mitu_skillologies

@mITuSkillologies

@mitu_group

@mitu-skillologies

@MITUSkillologies

kaggle

@mituskillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

@mituskillologies

contact@mitu.co.in

tushar@tusharkute.com