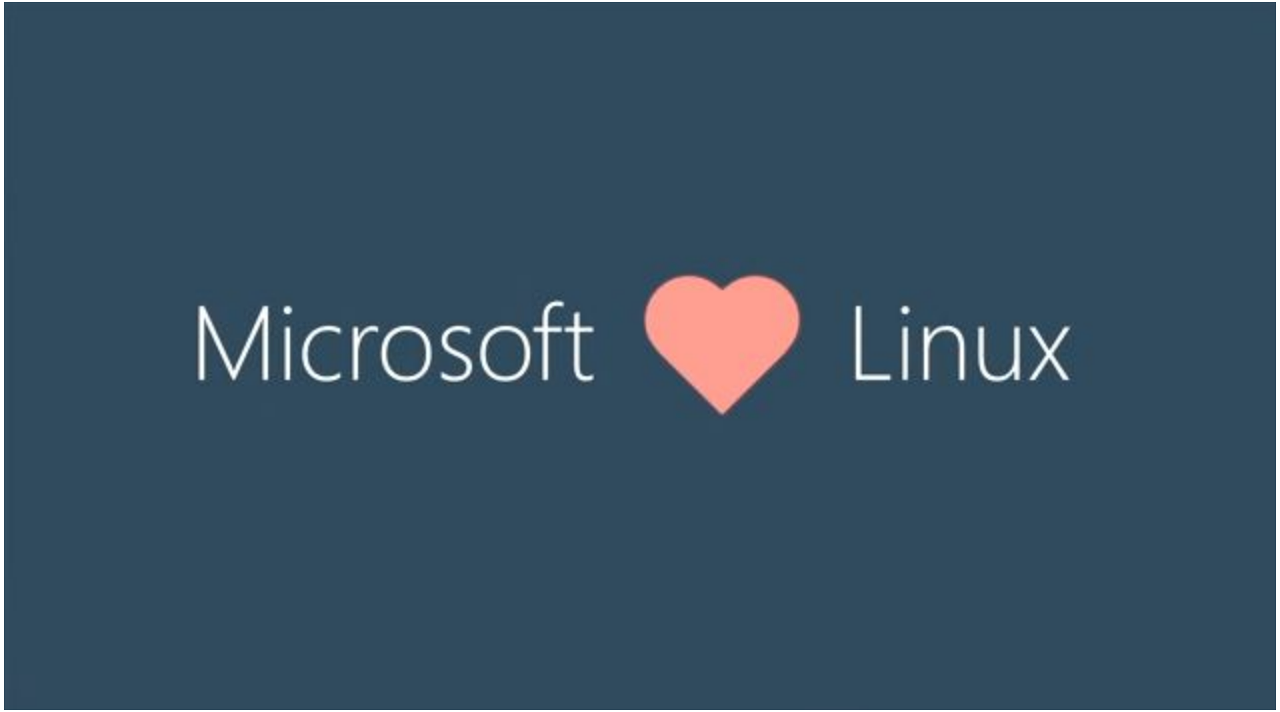
WSL 使用经验分享



概念区分

物理机：

物理机指的是一台实际的硬件设备。

物理机是实实在在存在的计算机系统，能够直接被操作系统和软件程序使用。传统的个人电脑、服务器以及笔记本电脑都属于物理机的范畴。

虚拟机：

虚拟机是在一台物理机上创建的逻辑计算机系统。

它在硬件上并不存在，但通过虚拟化技术可以模拟出完整的计算机环境。在一个物理机上可以同时运行多个虚拟机，每个虚拟机都被独立分配了一部分物理资源，如内存、处理器核心、硬盘空间等

容器：

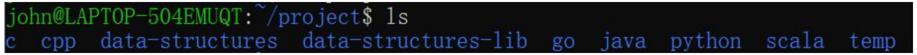
容器是一种虚拟化技术，类似于虚拟机，但有所不同。

容器技术允许开发人员打包应用程序及其所有依赖项（例如库、配置文件等）到一个称为容器的独立单元中。与虚拟机不同，容器并不需要模拟完整的操作系统，而是共享主机操作系统的内核。这使得容器更加轻量级和快速启动。

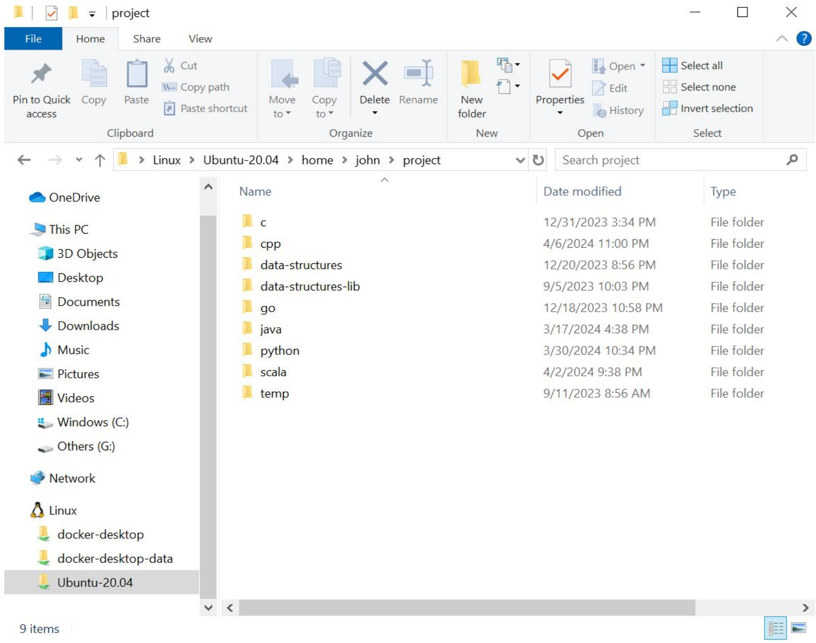
WSL 介绍及使用

适用于 Linux 的 Windows 子系统文档 | Microsoft Learn

命令行界面 CLI（Command-line interface）
图形用户界面 GUI（Graphical User Interface）



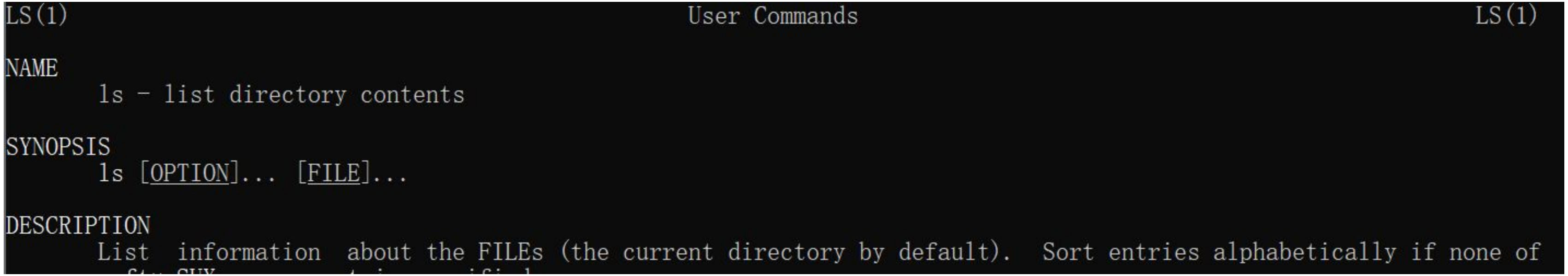
```
john@LAPTOP-504EMUQT:~/project$ ls
c  cpp  data-structures  data-structures-lib  go  java  python  scala  temp
```



Name	Date modified	Type
c	12/31/2023 3:34 PM	File folder
cpp	4/6/2024 11:00 PM	File folder
data-structures	12/20/2023 8:56 PM	File folder
data-structures-lib	9/5/2023 10:03 PM	File folder
go	12/18/2023 10:58 PM	File folder
java	3/17/2024 4:38 PM	File folder
python	3/30/2024 10:34 PM	File folder
scala	4/2/2024 9:38 PM	File folder
temp	9/11/2023 8:56 AM	File folder

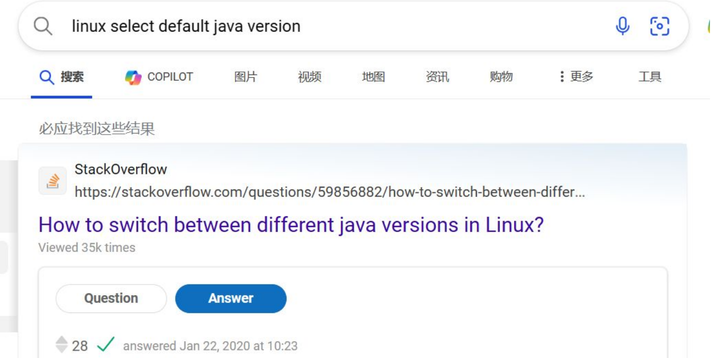
熟悉命令

1. 认识命令：不必死记硬背，了解命令的含义（man命令 or 搜索）



```
LS(1)                              User Commands                              LS(1)

NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about the FILEs (the current directory by default).  Sort entries alphabetically if none of
```

2. 不会就搜：



linux select default java version

必应找到这些结果

StackOverflow
https://stackoverflow.com/questions/59856882/how-to-switch-between-differ...

How to switch between different java versions in Linux?

Viewed 35k times

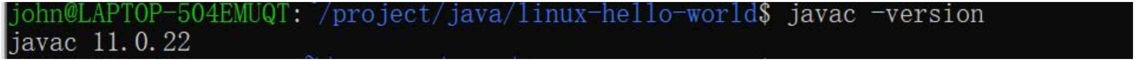
28 ✓ answered Jan 22, 2020 at 10:23

WSL 开发环境

- 依赖
- 命令
- IDE

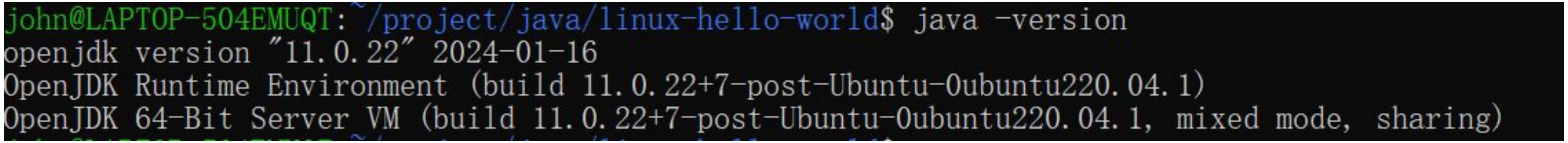
依赖

1. Java 编译器



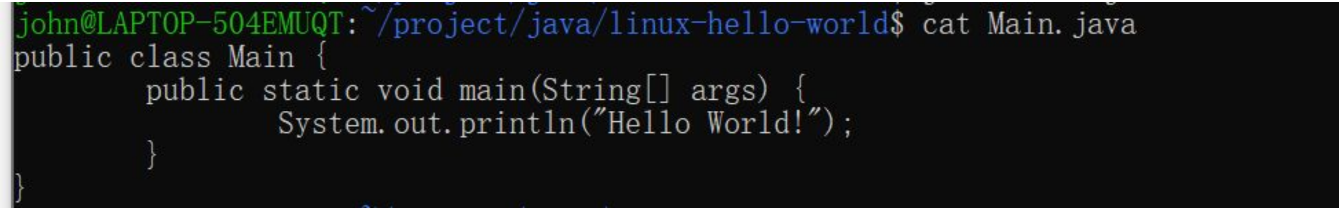
```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ javac -version
javac 11.0.22
```

2. JRE



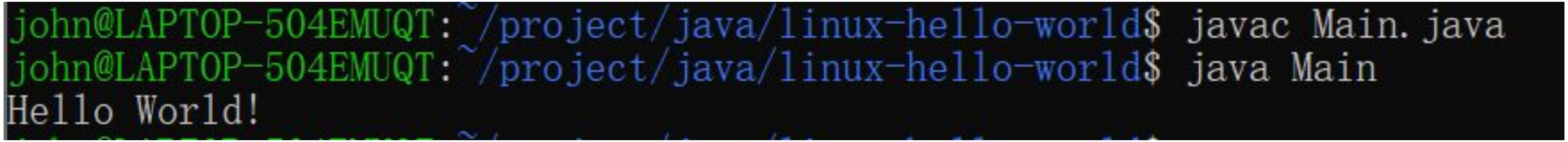
```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ java -version
openjdk version "11.0.22" 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu220.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu220.04.1, mixed mode, sharing)
```

3. 源代码



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ cat Main.java
public class Main {
        public static void main(String[] args) {
                System.out.println("Hello World!");
        }
}
```

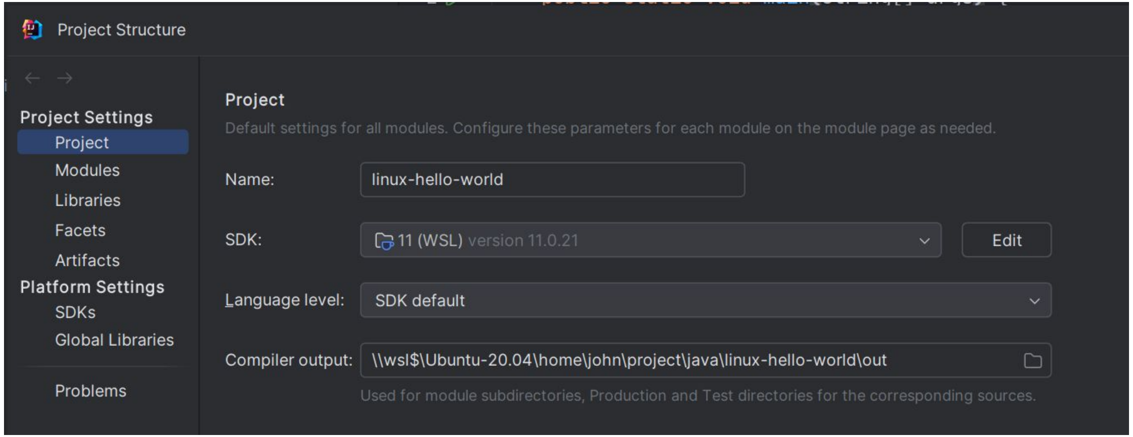
命令



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ javac Main.java
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ java Main
Hello World!
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$
```

IDE(以 IntelliJ IDEA 为例)

[WSL | IntelliJ IDEA Documentation (jetbrains.com)]



Project

Default settings for all modules. Configure these parameters for each module on the module page as needed.

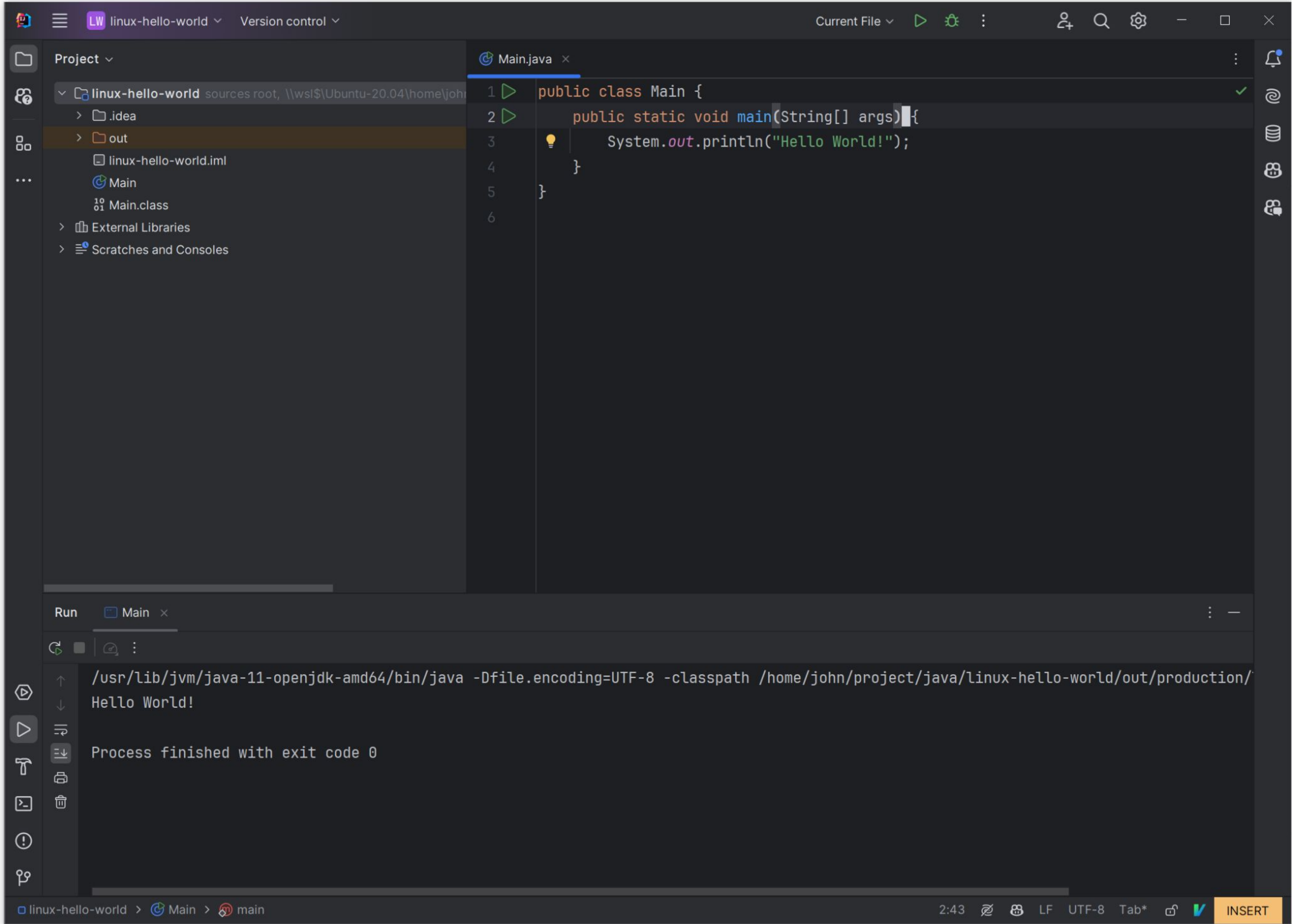
Name: linux-hello-world

SDK: 11 (WSL) version 11.0.21

Language level: SDK default

Compiler output: \\wsl$\Ubuntu-20.04\home\john\project\java\linux-hello-world\out

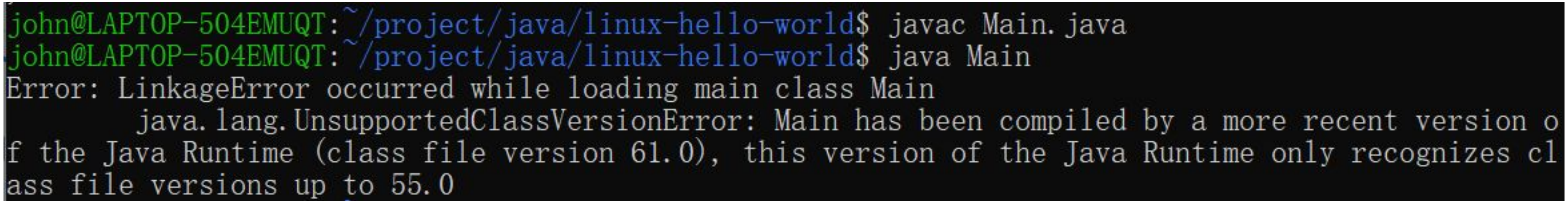
Used for module subdirectories, Production and Test directories for the corresponding sources.



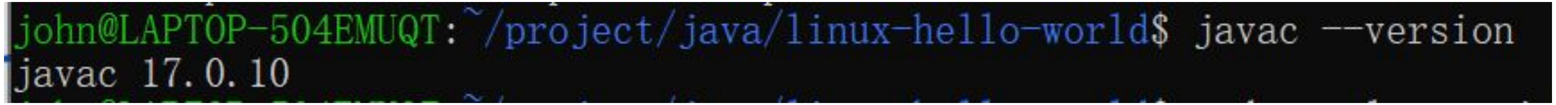
```java
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}

```

```
/usr/lib/jvm/java-11-openjdk-amd64/bin/java -Dfile.encoding=UTF-8 -classpath /home/john/project/java/linux-hello-world/out/production/
Hello World!

Process finished with exit code 0
```

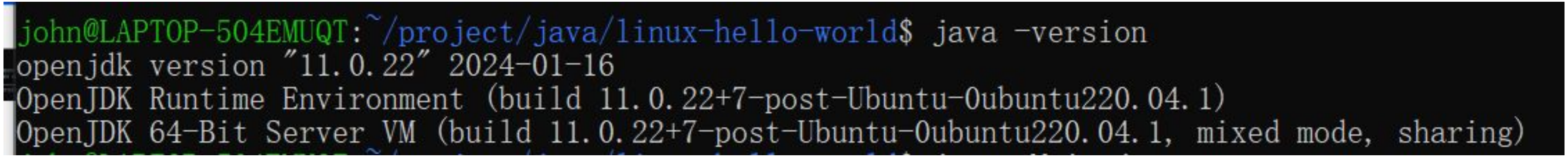
一个真实的小插曲



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ javac Main.java
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ java Main
Error: LinkageError occurred while loading main class Main
        java.lang.UnsupportedClassVersionError: Main has been compiled by a more recent version o
f the Java Runtime (class file version 61.0), this version of the Java Runtime only recognizes cl
ass file versions up to 55.0
```



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ javac --version
javac 17.0.10
```



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ java -version
openjdk version "11.0.22" 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Ubuntu-0ubuntu220.04.1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Ubuntu-0ubuntu220.04.1, mixed mode, sharing)
```

How to change your Java and javac version on Ubuntu / Linux

#linux #ubuntu #java #tutorial

In this short post i will show you how to easily change your java version and your java compiler version on your Ubuntu / Linux machine.

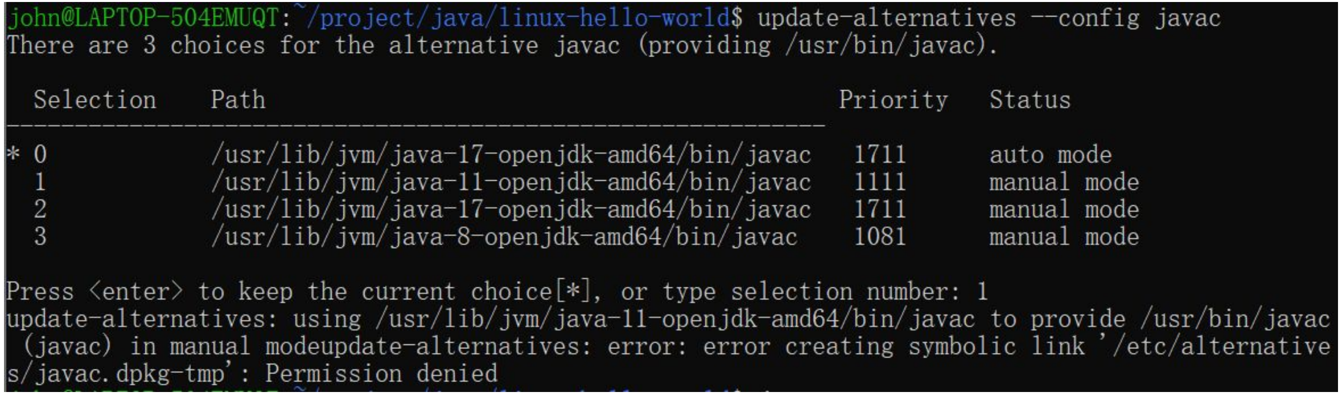
If you have installed more than one version of java on your system, you still have the opportunity to switch between different versions. Maybe you wanted to check out the newest features of the new java version or you have some kind of old project that only works for some older version (yes i know java ist a wora language but anyway).

If that is the case try the following commands:

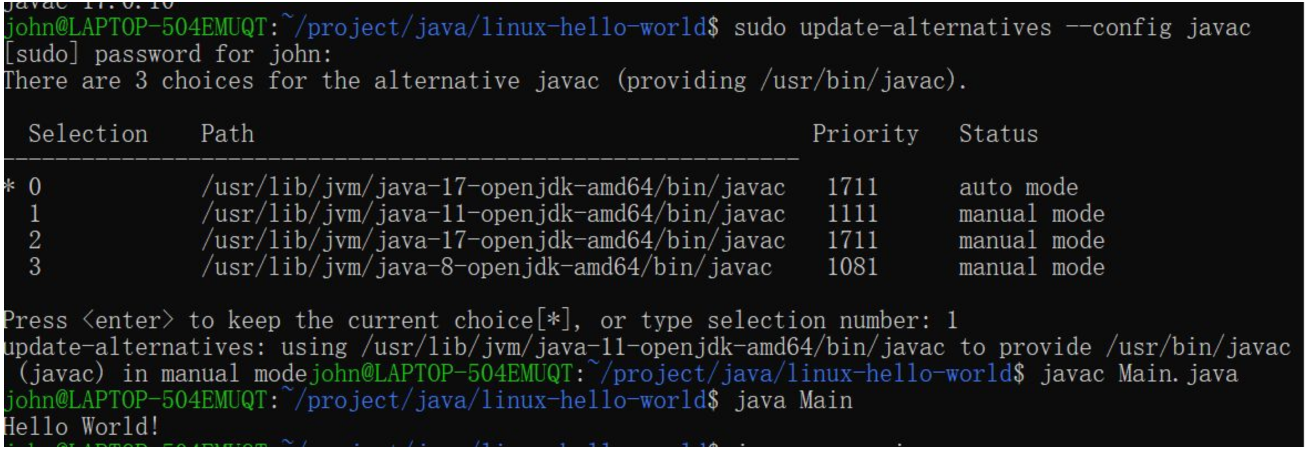
```
$ update-alternatives --config java
```

```
$ update-alternatives --config javac
```

```
There are 2 choices for the alternative java (providing /usr/bin/java).

  Selection    Path                                            Priority
------------------------------------------------------------
* 0            /usr/lib/jvm/java-11-openjdk-armhf/bin/java    1061
  1            /usr/lib/jvm/java-11-openjdk-armhf/bin/java    1044
  2            /usr/lib/jvm/java-8-openjdk-armhf/jre/bin/java 1061

Press enter to keep the current choice[*], or type selection number:
```



```
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ update-alternatives --config javac
There are 3 choices for the alternative javac (providing /usr/bin/javac).

  Selection    Path                                          Priority   Status
------------------------------------------------------------
* 0            /usr/lib/jvm/java-17-openjdk-amd64/bin/javac   1711      auto mode
  1            /usr/lib/jvm/java-11-openjdk-amd64/bin/javac   1111      manual mode
  2            /usr/lib/jvm/java-17-openjdk-amd64/bin/javac   1711      manual mode
  3            /usr/lib/jvm/java-8-openjdk-amd64/bin/javac    1081      manual mode

Press <enter> to keep the current choice[*], or type selection number: 1
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/javac to provide /usr/bin/javac
 (javac) in manual modeupdate-alternatives: error: error creating symbolic link '/etc/alternative
s/javac.dpkg-tmp': Permission denied
```



```
javac 17.0.10
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ sudo update-alternatives --config javac
[sudo] password for john:
There are 3 choices for the alternative javac (providing /usr/bin/javac).

  Selection    Path                                          Priority   Status
------------------------------------------------------------
* 0            /usr/lib/jvm/java-17-openjdk-amd64/bin/javac   1711      auto mode
  1            /usr/lib/jvm/java-11-openjdk-amd64/bin/javac   1111      manual mode
  2            /usr/lib/jvm/java-17-openjdk-amd64/bin/javac   1711      manual mode
  3            /usr/lib/jvm/java-8-openjdk-amd64/bin/javac    1081      manual mode

Press <enter> to keep the current choice[*], or type selection number: 1
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/javac to provide /usr/bin/javac
 (javac) in manual modejohn@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ javac Main.java
john@LAPTOP-504EMUQT:~/project/java/linux-hello-world$ java Main
Hello World!
```