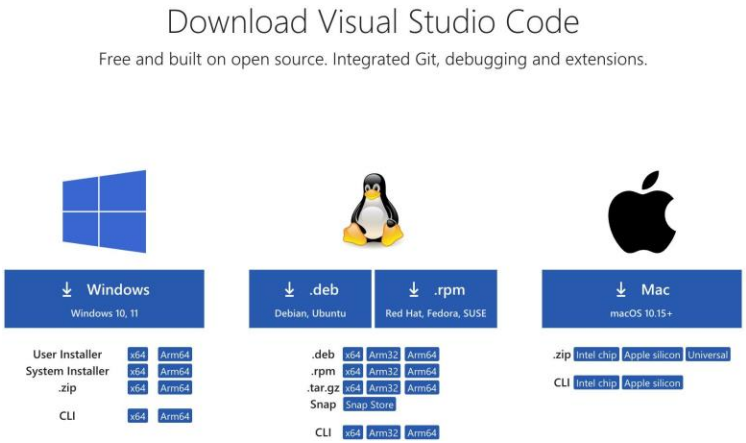
VSCode安裝與測試

PDSA-2025-spring

Step 1: 安裝VSCode

前往Visual Studio Code官網, 依照自己的作業系統下載並且安裝。



https://code.visualstudio.com/Download

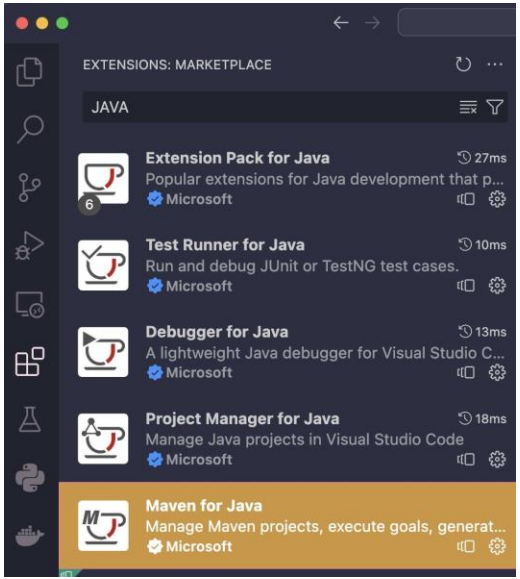
Step 2: 安裝JAVA&VSCode JAVA插件

A.安裝Coding Pack: https://code.visualstudio.com/docs/java/java-tutorial

Windows: https://aka.ms/vscode-java-installer-win
MacOS: https://aka.ms/vscode-java-installer-mac

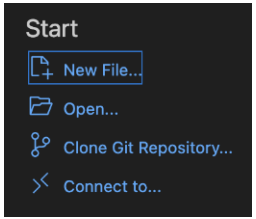
B.從終端機安裝（之後要進VSCode安裝插件）
Ubuntu: sudo apt install openjdk-21-jdk
MacOS: brew install openjdk

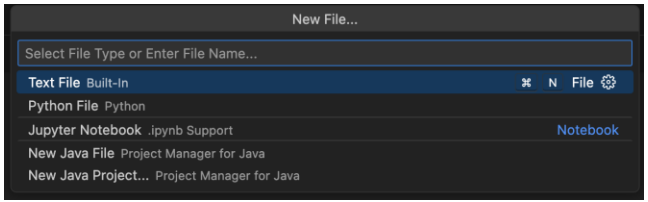
確認插件已安裝

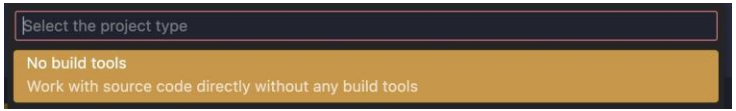

EXTENSIONS: MARKETPLACE
JAVA
Extension Pack for Java
27ms
Popular extensions for Java development that p...
Microsoft
Test Runner for Java
10ms
Run and debug JUnit or TestNG test cases.
Microsoft
Debugger for Java
13ms
A lightweight Java debugger for Visual Studio C...
Microsoft
Project Manager for Java
18ms
Manage Java projects in Visual Studio Code
Microsoft
Maven for Java
Manage Maven projects, execute goals, generat...
Microsoft

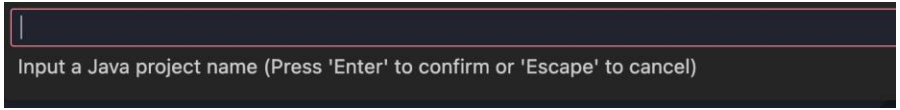
Step 3: 創建專案

1. 在首頁點選New File
2. 點選New Java Project
3. 點選No build tools
4. 選擇Java專案要放的位置
5. 輸入專案名稱按Enter

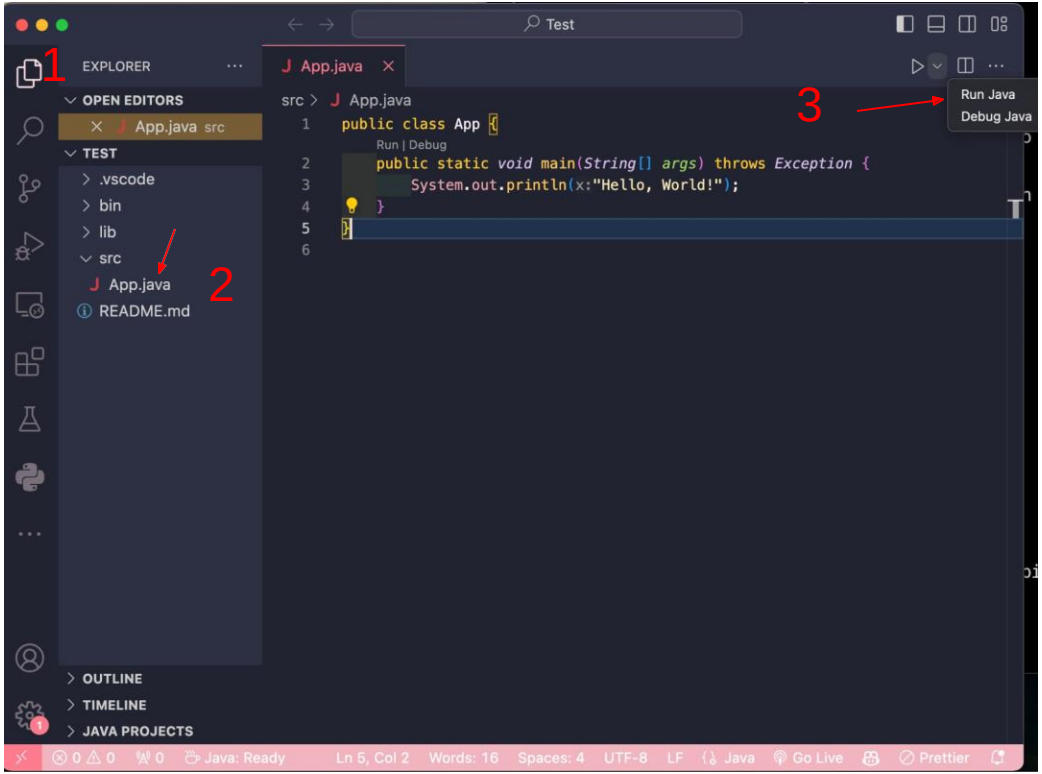


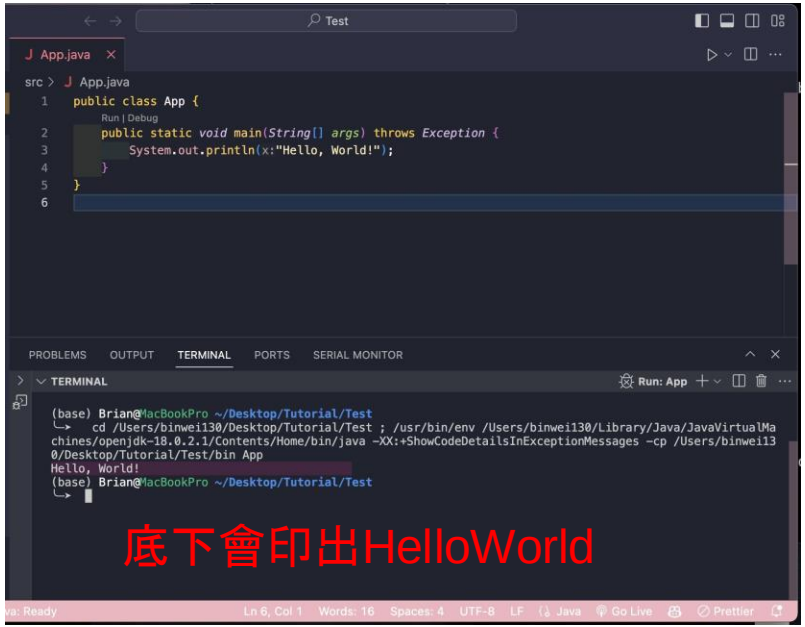




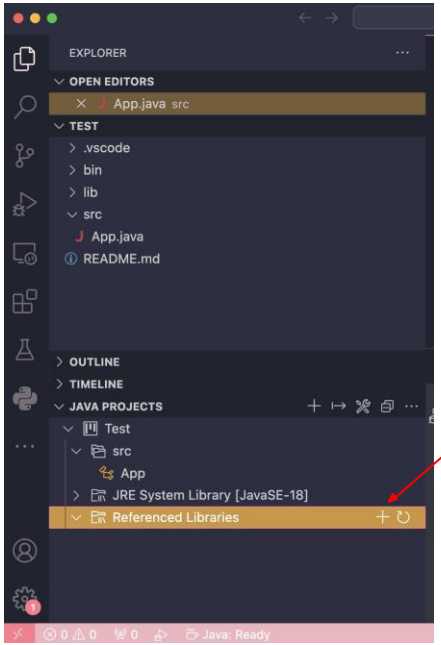


Step 4: 測試是否可以正常運作

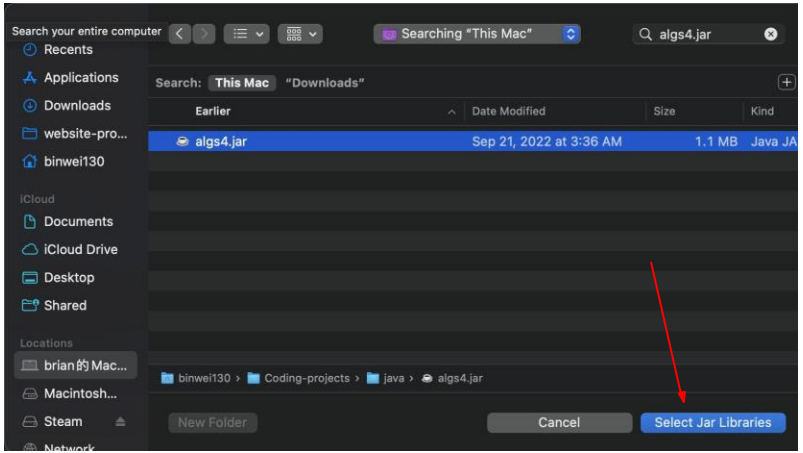




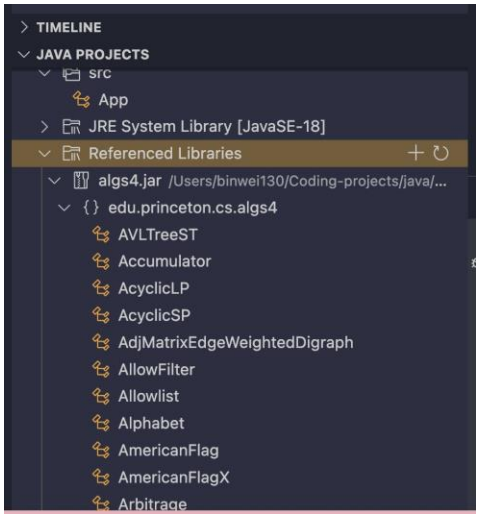
Step 5: 導入課程所需插件(algs4.jar)



在Referenced Libraries上面按 +
選擇下載好的algs4.jar後按確定



預期結果



Step 6: 更改App.java測試是否有導入成功

```java
import edu.princeton.cs.algs4.Stack;
public class App {
   public static void main(String[] args) throws Exception {
       Stack<Integer> stack = new Stack<Integer>();
       stack.push(1);
       System.out.println(stack.peek());
   }
}
```

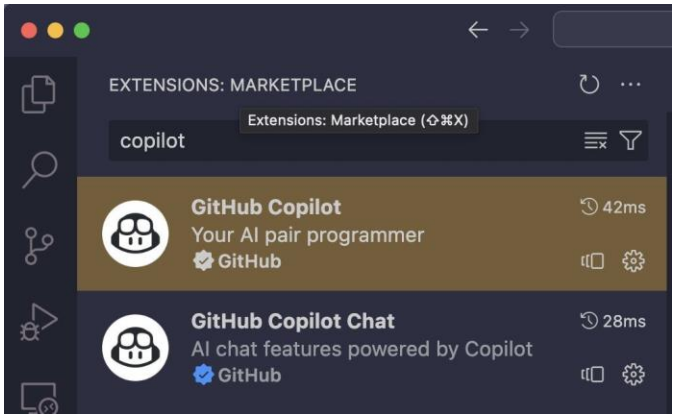
若Terminal可以印出“1”的話，代表導入成功，之後作業可以照此步驟進行。

```
(base) Brian@MacBookPro ~/Desktop/Tutorial/Test
└➤   cd /Users/binwei130/Desktop/Tutorial/Test ; /usr/bin/env /Users/binwei1
30/Library/Java/JavaVirtualMachines/openjdk-18.0.2.1/Contents/Home/bin/java @
/var/folders/7x/w1g1jwqx48z8yzslrm3bz0m40000gn/T/cp_9mrumcd8ciwhu2sdkslub0lcu
.argfile App
1
```

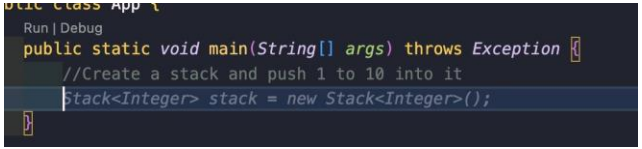
Bonus: 用@ntu.edu.tw信箱申請Copilot

可參考此文章

：https://medium.com/%E5%BD%BC%E5%BE%97%E6%BD%98%E7%9A%84-swift-ios-app-%E9%96%8B%E7%99%BC%E5%95%8F%E9%A1%8C%E8%A7%A3%E7%AD%94%E9%9B%86/%E7%94%A8%E8%80%81%E5%B8%AB-%E5%AD%B8%E7%94%9F%E8%BA%AB%E4%BB%BD%E5%85%8D%E8%B2%BB%E4%BD%BF%E7%94%A8-github-copilot-223236e0e0e8

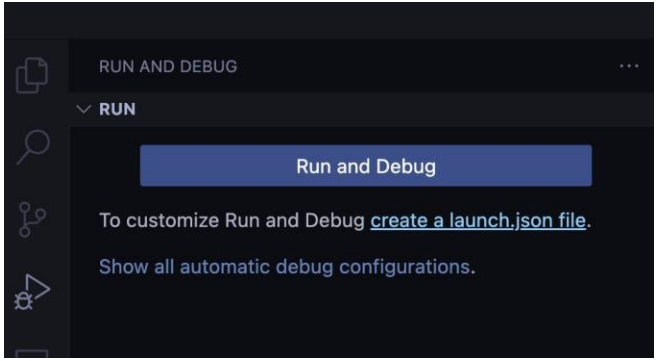


用學校信箱申請GitHub以後並且提交認證文件，審核成功後就可以在Vscode安裝Copilot，AI幫你建議接下來要打什麼程式碼.



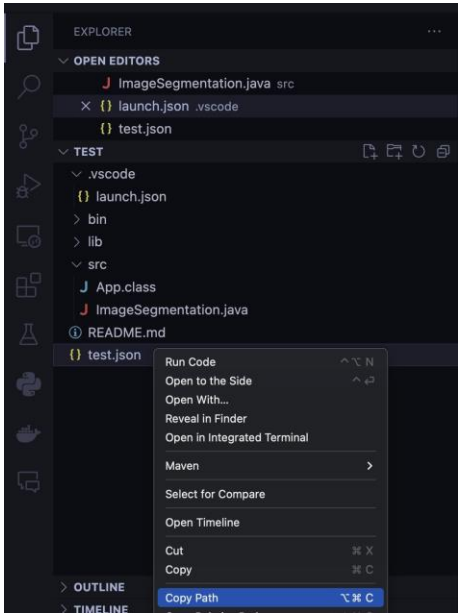
使用測試資料:

1. 使用如同前面提到的導入方式導入lib中的json-simple-1.1.1.jar
2. 將作業連結中的Test code貼上Vscode
3. 把你的程式碼填進去
4. 到左方的Run&Debug按下create a launch.json file.



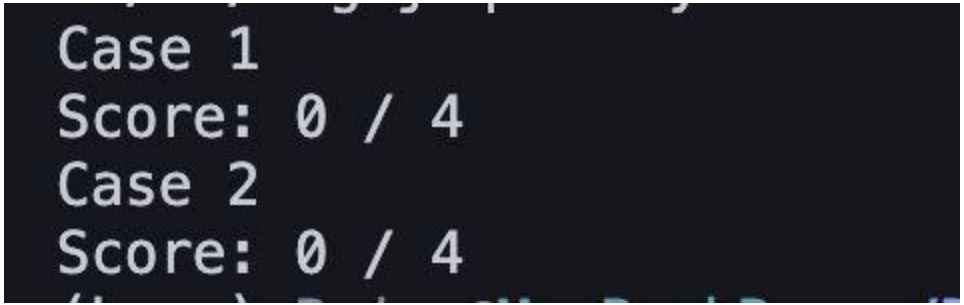
使用測試資料

複製下載好的test.json路徑, 加入.vscode/launch.json檔案中的args選項



```jsonc
{
    // Use IntelliSense to learn about possible attributes.
    // Hover to view descriptions of existing attributes.
    // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "type": "java",
            "name": "Current File",
            "request": "launch",
            "mainClass": "${file}"
        },
        {
            "type": "java",
            "name": "ImageSegmentation",
            "request": "launch",
            "mainClass": "ImageSegmentation",
            "projectName": "ImageSegmentation_6c507772",
            "args": ["test.json"]
        }
    ]
}
```

再如前面一樣正常跑就可以有以下輸出



有問題可以在Discord發問