**Assisted Practice: 1.2 Access Modifiers**

This section will guide you to:

* Create a Java project in your IDE
* Write a program in Java to verify the working of access modifiers

This lab has three subsections, namely:

* + 1. Writing a program in Java to implement access modifiers
    2. Executing the program and verifying how the access modifiers work
    3. Pushing the code to your GitHub repositories

**Step 1.2.1:** Writing a program in Java to implement access modifiers

There are two ways you can perform this step; you can create a new Java project, or you can create a new Java class in the existing project. It is preferable to create a new Java class in the existing project but feel free to explore the first option. The steps mentioned below will work once you create a project in Java.

* Open Eclipse
* *[Right click]* on the **src** folder of the project
* Select *New* -> *Java Class* -> Enter the filename (follow camelCasing)
* Execute the code below resolving the warning and errors due compatibility-related issues

//1. Class is having Default access modifier

**class** defAccessSpecifier

{

**void** display()

{

System.***out***.println("You are using defalut access specifier");

}

}

**public** **class** accessSpecifiers1 {

**public** **static** **void** main(String[] args) {

//default

System.***out***.println("Dafault Access Specifier");

defAccessSpecifier obj = **new** defAccessSpecifier();

obj.display();

}

}

//2. using private access specifiers

**class** priaccessspecifier

{

**private** **void** display()

{

System.***out***.println("You are using private access specifier");

}

}

**public** **class** accessSpecifiers2 {

**public** **static** **void** main(String[] args) {

//private

System.***out***.println("Private Access Specifier");

priaccessspecifier obj = **new** priaccessspecifier();

//trying to access private method of another class

//obj.display();

}

}

//3. using protected access specifiers

**package** pack1;

**public** **class** proaccessspecifiers {

**protected** **void** display()

{

System.***out***.println("This is protected access specifier");

}

}

//create another package

**package** pack2;

**import** pack1.\*;

**public** **class** accessSpecifiers3 **extends** proaccessspecifiers {

**public** **static** **void** main(String[] args) {

accessSpecifiers3 obj = **new** accessSpecifiers3 ();

obj.display();

}

}

//4. using public access specifiers

**package** pack1;

**public** **class** pubaccessspecifiers {

**public** **void** display()

{

System.***out***.println("This is Public Access Specifiers");

}

}

//create another package

**package** pack2;

**import** pack1.\*;

**public** **class** accessSpecifiers4 {

**public** **static** **void** main(String[] args) {

pubaccessspecifiers obj = **new** pubaccessspecifiers();

obj.display();

}

}

**Step 1.2.2:** Executing the program and verifying how the access modifiers work

Before you execute the program, check for syntactical corrections. If no errors are found, follow the steps mentioned below:

* ***[Right click]*** in the program space
* Select *Run As Java Application*







**Step 1.2.3:** Pushing the code to your GitHub repositories

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**