**Assisted Practice: 2.2 Constructors**

This section will guide you to:

* Create a Java project in your IDE
* Write a program in Java to create constructors

This lab has three subsections, namely:

* + 1. Writing a program in Java to verify the implementations of constructor types
    2. Executing the program and verifying constructors
    3. Pushing the code to your GitHub repositories

**Step 2.2.1:** Write a program in Java to verify the implementations of constructor types

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

//default constructor

**class** EmpInfo{

**int** id;

String name;

**void** display() {

System.***out***.println(id+" "+name);

}

}

**public** **class** constructorDemo {

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

EmpInfo emp1=**new** EmpInfo();

EmpInfo emp2=**new** EmpInfo();

emp1.display();

emp2.display();

}

}

//parameterized constructor

**class** Std{

**int** id;

String name;

Std(**int** i,String n)

{

id=i;

name=n;

}

**void** display() {

System.***out***.println(id+" "+name);

}

}

**public** **class** paramConstrDemo {

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

Std std1=**new** Std(2,"Alex");

Std std2=**new** Std(10,"Annie");

std1.display();

std2.display();

}

}

**Step 2.2.2:** Executing the program and verifying the constructors

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 2.2.3:** 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**