Core Java - Classes Assignment

Instructions:

Below you will find a series of assignments that you will complete in this exercise. You will need to create a new project folder for this task and name it **Java_Assignment_4_Classes**. You can use the default settings when creating the new project.

For each of the below assignments you will create new classes as specified below. Pay attention to the package name as you will use ***different*** packages in these exercises as indicated below in the assignment directions.

Most classes will NOT declare a main() method except for the Application.java file that you will write for that package. In your assignments, you will execute the main() method of the Application.java file.

NOTE A couple assignments will NOT compile. They are there to walk you through some common errors and observe the compiler message printed.

Assignments

1. Write a class named **Person.** The person class should define a name (String) and age (int) instance variables. Write another class, Application, in which you instantiate a new Person object. Set the person's age and name properties. Print the age and name properties of the created person. Place these files in a package named, "one".

   Note: The Person.java file should NOT declare a main() method.

2. Write a class named **PrivatePerson**. Declare instance variables name (String) and age (int) and mark them private. Write getter and setter methods to access these properties. In a Main class, create a new instance of Person and print its properties name and age. Place these files in a package named, "two".

3. Write a class named **TalkablePerson** that declares a public talk() method along with name (String) and age (int) properties. The talk() method should have a void return type and print the message, "Hello my name is: {name}". Instead of {name} the person's name property should be printed. In a Main class, create an instance of Person and use their talk method. Place these files in a package named, "three".

   Hint: Use the keyword **this** to access a member variable

4. Rewrite TalkablePerson from Assignment 3 to declare its talk() method private. Run the Main class. Notice the compiler error. Mark the talk() method public again.

5. Write a class named **ConstructorPerson** that declares name (String) and age (int) instance variables. Write a no-arg constructor[1] that initializes ConstructorPerson with default values. In a Main class, create an instance of a ConstructorPerson and print its properties name and age.

   Place these files in a package named, "four".

[1] Remember that a no-arg constructor is a constructor that declares zero parameters. For example, say we have the class Animal. A no-arg constructor would look like this:

```
public  Animal(){ }
```

Notice that to no parameters are declared. Note, that you can still place any number of statements with the curly braces.

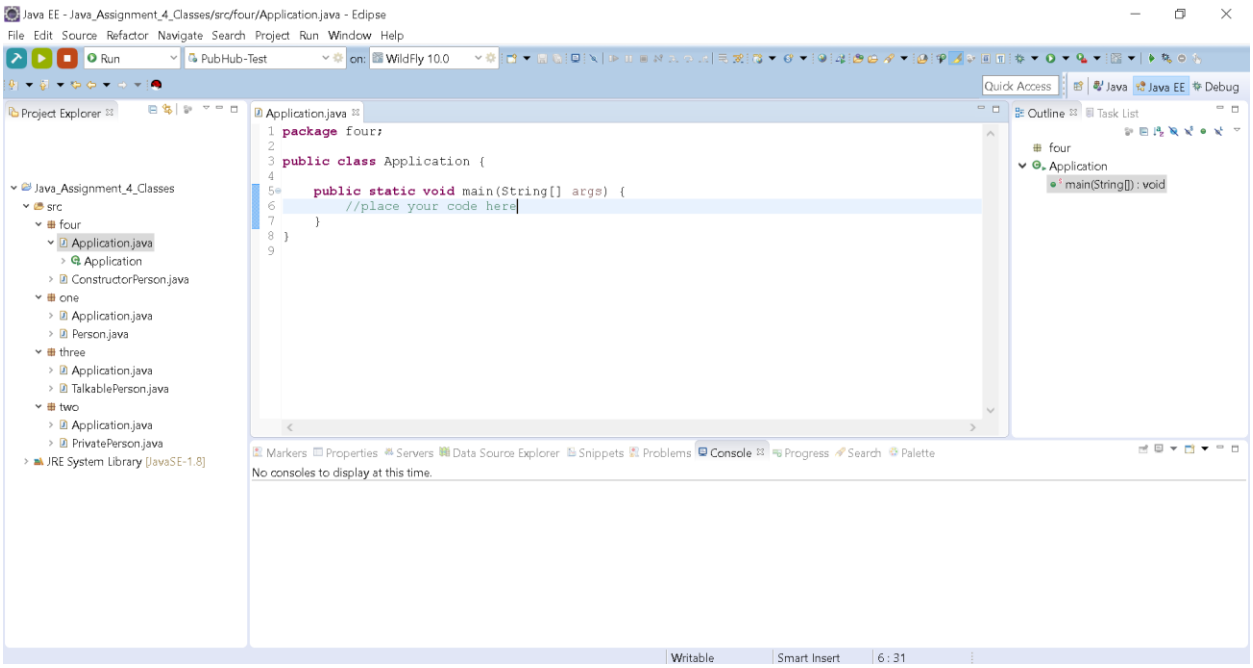
Goal:

The purpose of this assignment is to provide you practice with writing classes and instantiating instances of them. You will utilize the **main()** method of your classes to execute a particular goal of the assignment.

By the end of this assignment, you will have 1 new project folder with the associated classes that meet the requirements mentioned above.

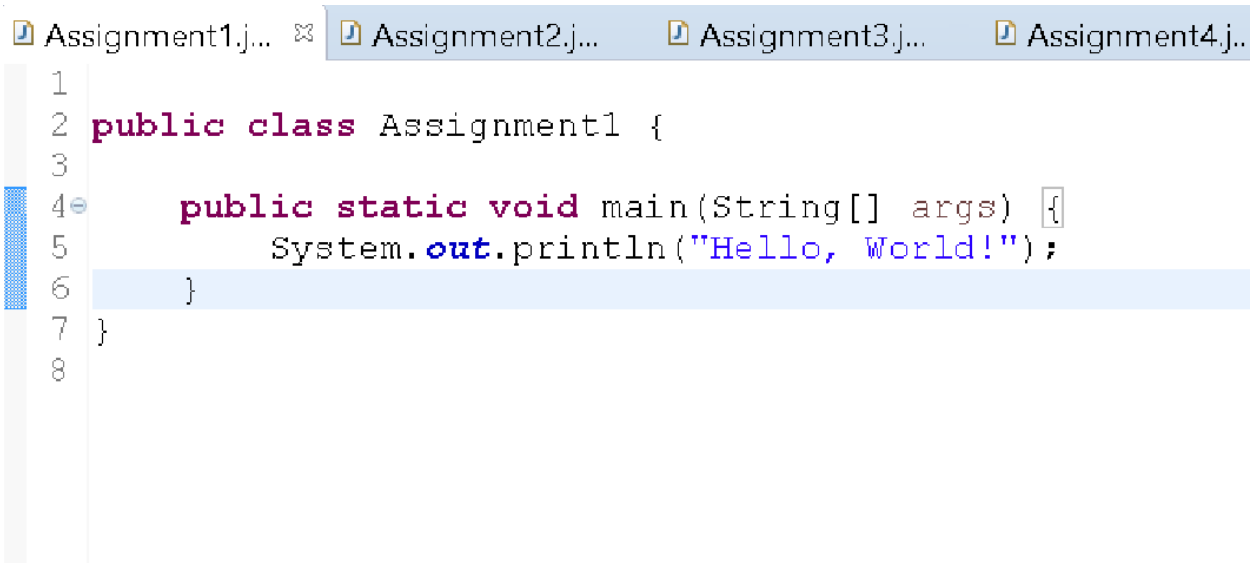
Helpful Notes

Project Structure:



- Your project structure should resemble the above. Note that the name of the project is **Java_Assignment_4_Classes** and all class files are named according to the assignment number.

Assignment Classes:



- Notice the indentation of the statement on line 5 to show that it is part of the main() method. Likewise, your code should follow this pattern.