LAB 3 Part 1 OBJECTS & CLASSES

Objective:

In the lecture you learnt about classes and objects. The main objective of this lab is to learn how to instantiate the class objects, set their properties and access their methods, how to access other classes stored on other directory by setting classpath. You will also play with String class in this lab.

String

The *String* class represents character strings. All string literals in Java programs, such as "abc", are implemented as instances of this class. The class *String* includes methods for examining individual characters of the sequence, for comparing strings, for searching strings, for extracting substrings, and for creating a copy of a string with all characters translated to uppercase or to lowercase
Here are some more examples of how strings can be used:

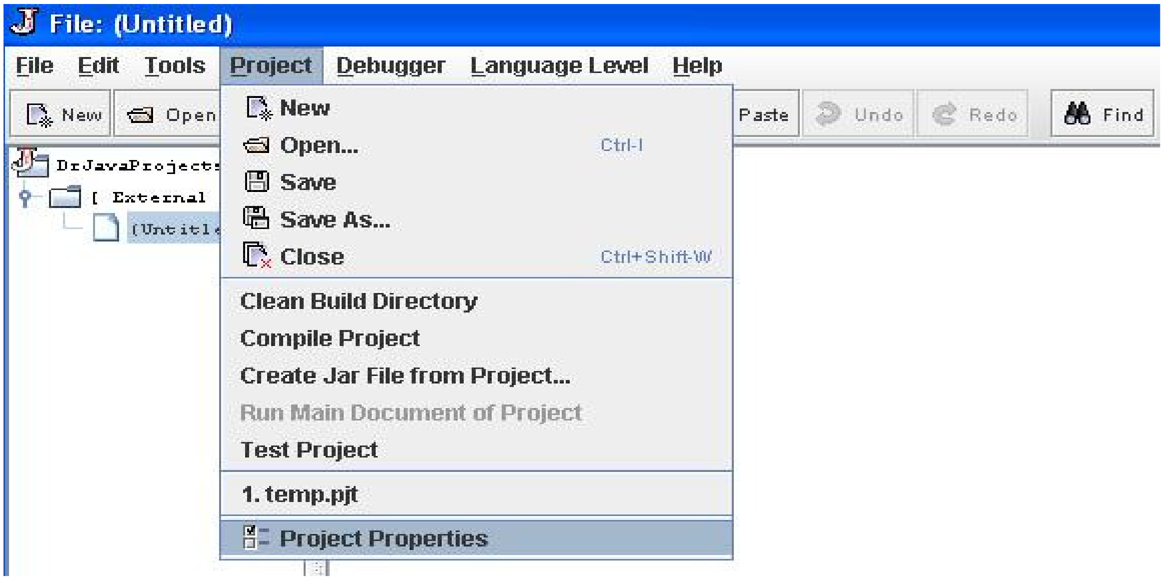
```
// Show String “abc” in Interaction Pane

  System.out.println(“abc”);
```

```
// Show String “abccde” in Interaction Pane

  String cde= “cde”;
  System.out.println(“abc” +cde);
```

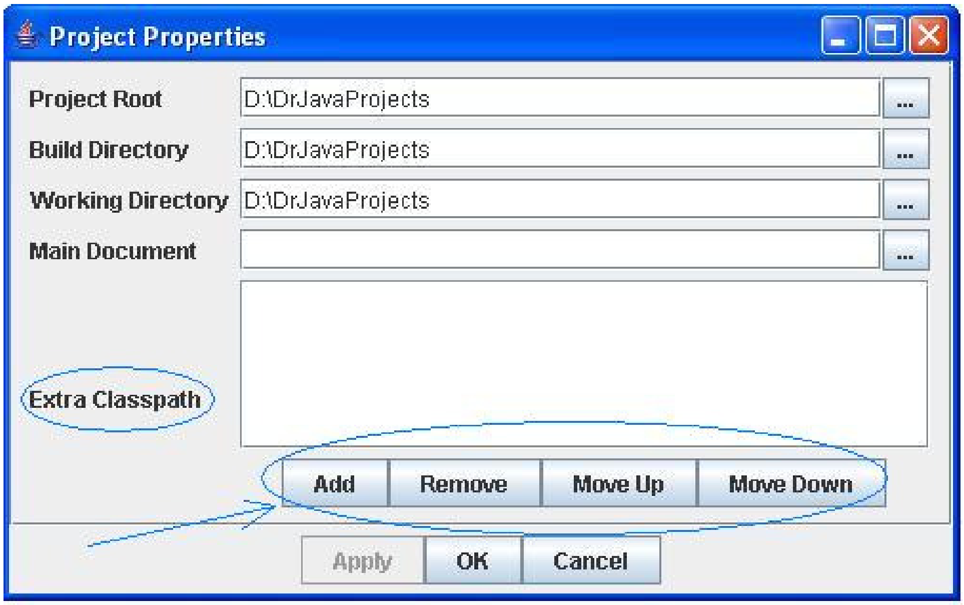
```
// Show String “c” in Interaction Pane

  String c = “abc”.substring(2,3);
  System.out.println(c);
```

ClassPath

ClassPath tells JDK tools and applications where to find third-party and user-defined classes i.e. classes that are not Java extensions or part of the Java platform. You only need to set the class path when you want to load a class that's not in the current directory or in any of its subdirectories, and not in a location specified by the extensions mechanism.
DrJava had made this thing very simple. In DrJava there is an option named *Project Properties* to set and remove the classpath in *Project menu*.



When we click this option, another form will open. In this form you will find various kinds of options available to set the classpath and to *remove* the existing classpath.



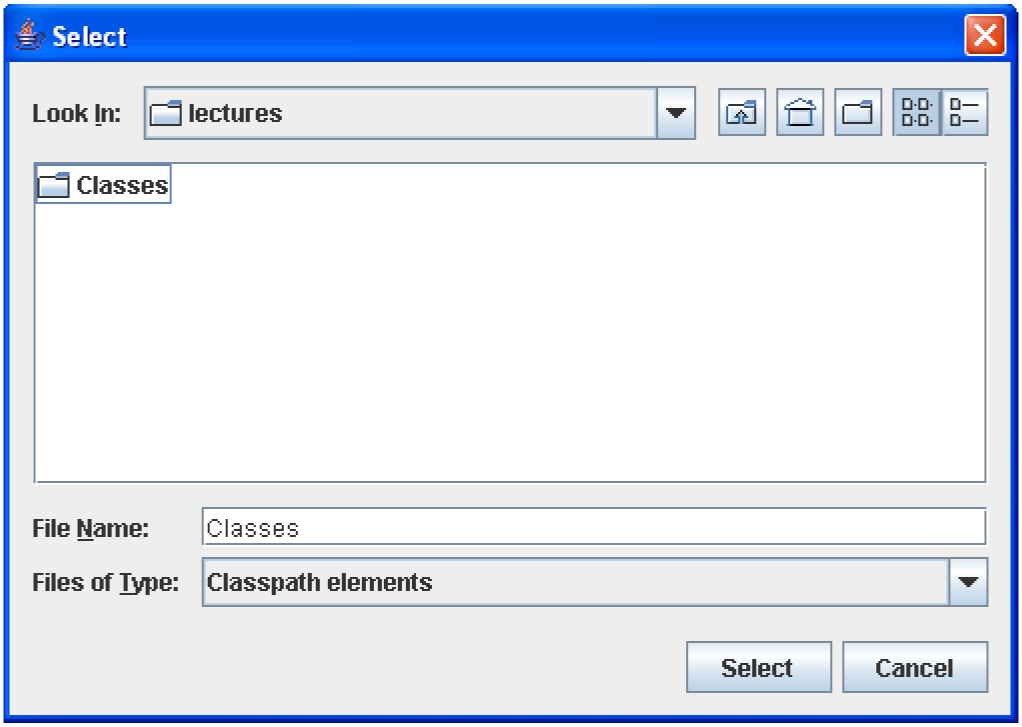
Project Root: D:\DrJavaProjects

Build Directory: D:\DrJavaProjects

Working Directory: D:\DrJavaProjects

Main Document:

Now to add classpath of an existing class click *Add* button.



Look In: lectures

File Name: Classes

Files of Type: Classpath elements

Now give the address of the folder which contains classes as shown above. Click select to select the folder. Now click OK .Now this class is available i.e. we can instantiate this class and use all its functionality. Not only this class is available to us but any class which is in this folder is accessible. We can instantiate it and use it.

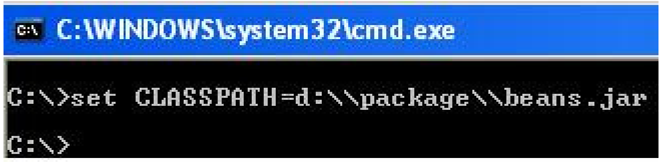
Alternate way of setting ClassPath

Hint:

This is an alternate way, while using DrJava don't have to follow these settings.

The class path can be set using either the -classpath option when calling a JDK tool (the preferred method) or by setting the CLASSPATH environment variable.

```
C :> set CLASSPATH=classpath1; classpath2...
```



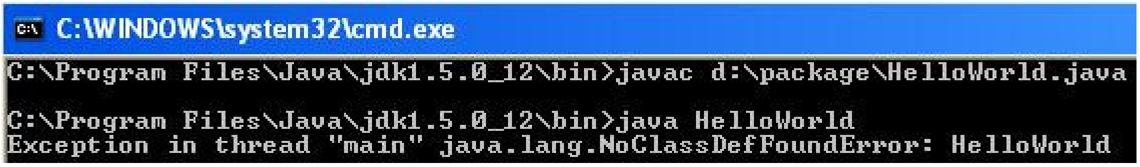
C:\WINDOWS\system32\cmd.exe

```
C:\>set CLASSPATH=d:\\package\\beans.jar
C:\>
```

Class paths to the .jar, .zip or .class files. Each *classpath* should end with a filename or directory depending on what you are setting the class path to:

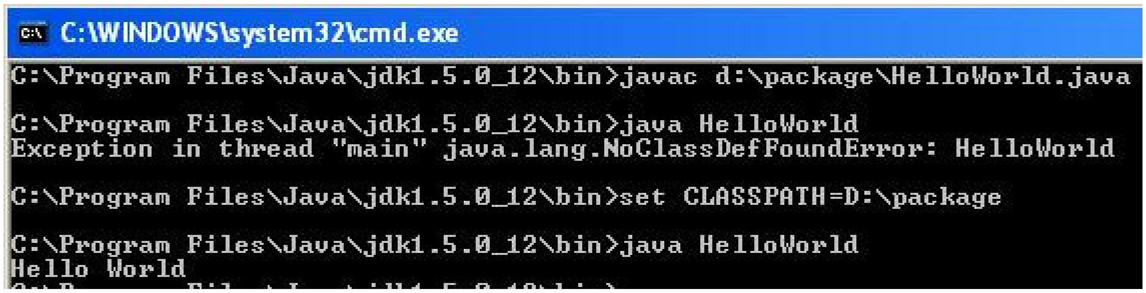
- For a .jar or .zip file that contains .class files, the class path ends with the name of the .zip or .jar file.
- For .class files in an unnamed package, the class path ends with the directory that contains the .class files.
- For .class files in a named package, the class path ends with the directory that contains the "root" package (the first package in the full package name).

Following Error will occur if we don’t set the classpath.



```
C:\Program Files\Java\jdk1.5.0_12\bin>javac d:\package\HelloWorld.java

C:\Program Files\Java\jdk1.5.0_12\bin>java HelloWorld
Exception in thread "main" java.lang.NoClassDefFoundError: HelloWorld
```

But after setting class path



```
C:\Program Files\Java\jdk1.5.0_12\bin>javac d:\package\HelloWorld.java

C:\Program Files\Java\jdk1.5.0_12\bin>java HelloWorld
Exception in thread "main" java.lang.NoClassDefFoundError: HelloWorld

C:\Program Files\Java\jdk1.5.0_12\bin>set CLASSPATH=D:\package

C:\Program Files\Java\jdk1.5.0_12\bin>java HelloWorld
Hello World
```

Unset CLASSPATH by using:

```
C :> set CLASSPATH=
```

Exercise

1. Display your Name using String class.

2. Create a String object named *country* and initialize it with “PAKISTAN”.

   a) Determine the length of this word using *length() method.*
   b) Determine the index of letter ‘*S*’ by using *indexOf()* method.
   c) Determine the index of “*IST”* by using *indexOf()* method.
   d) Determine the last index of letter *‘A’* by using *lastindexOf()* method.
   e) Convert this String into lowercase by using *toLowerCase().*
   f) Convert this lowercase String into uppercase by using *toUpperCase().*

3. Make a project by clicking Project Menu and selecting ‘New’ option. Enter a name for your project and click Save. DrJava will display Project Properties window, set classpath to *World.class* and *Turtle.class* as shown in the start of the tutorial.

Hint:

To make sure that you have made the project correctly and added the class paths of Turtle.class and World.class in the right way type the following in Interaction Pane:

```
System.out.println(new World());
```

OR simply make the object the the World class

```
World w = new World();
```

A window should appear as shown below, indicating that the classes have been added accurately.

4. Instantiate World class. Now make two objects named *sqr*, *rect* of Turtle class. Now draw square and, rectangle using these turtles. For square take x=200 and for rectangle length=180 and width=90.

Hint:

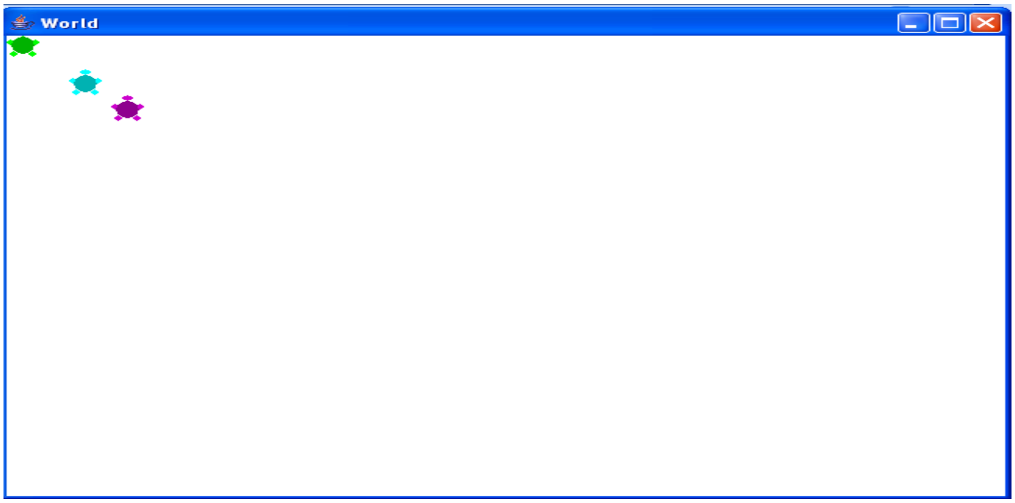
```
// Instantiate World object

                World world = new World();

//Instantiate Turtle class and pass object or World class as parameter.

                Turtle turtle1= new Turtle(world);

// You can also give coordinates to draw turtle at a specific position

                Turtle turtle2= new Turtle(a,b,world);
```

Where a and b are x and y coordinated on the window.

By entering the code:

```
World world= new World();
Turtle turtle1= new Turtle(10,10,world);
Turtle turtle2= new Turtle(50,50,world);
Turtle turtle3= new Turtle(77,77,world);
```

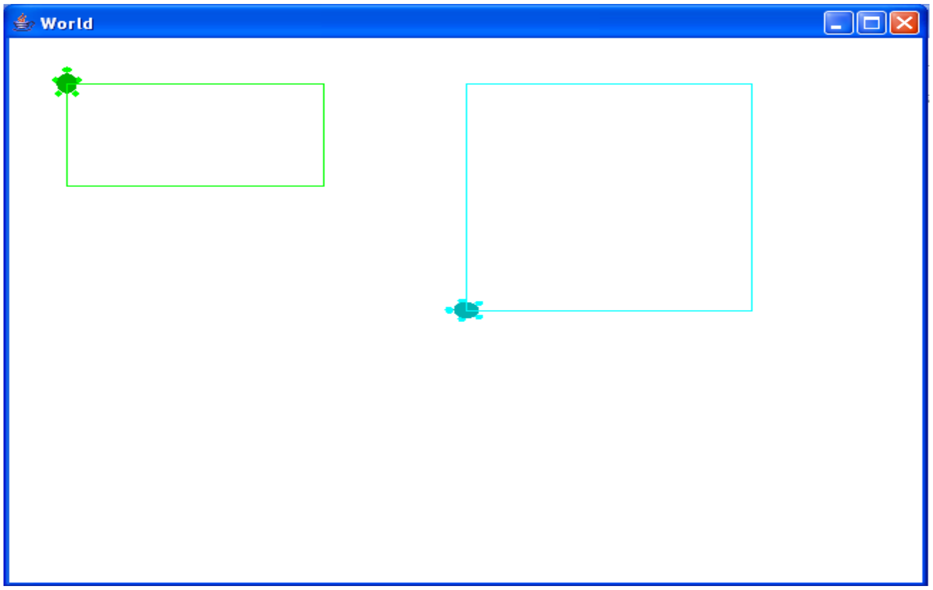
Following Window is displayed



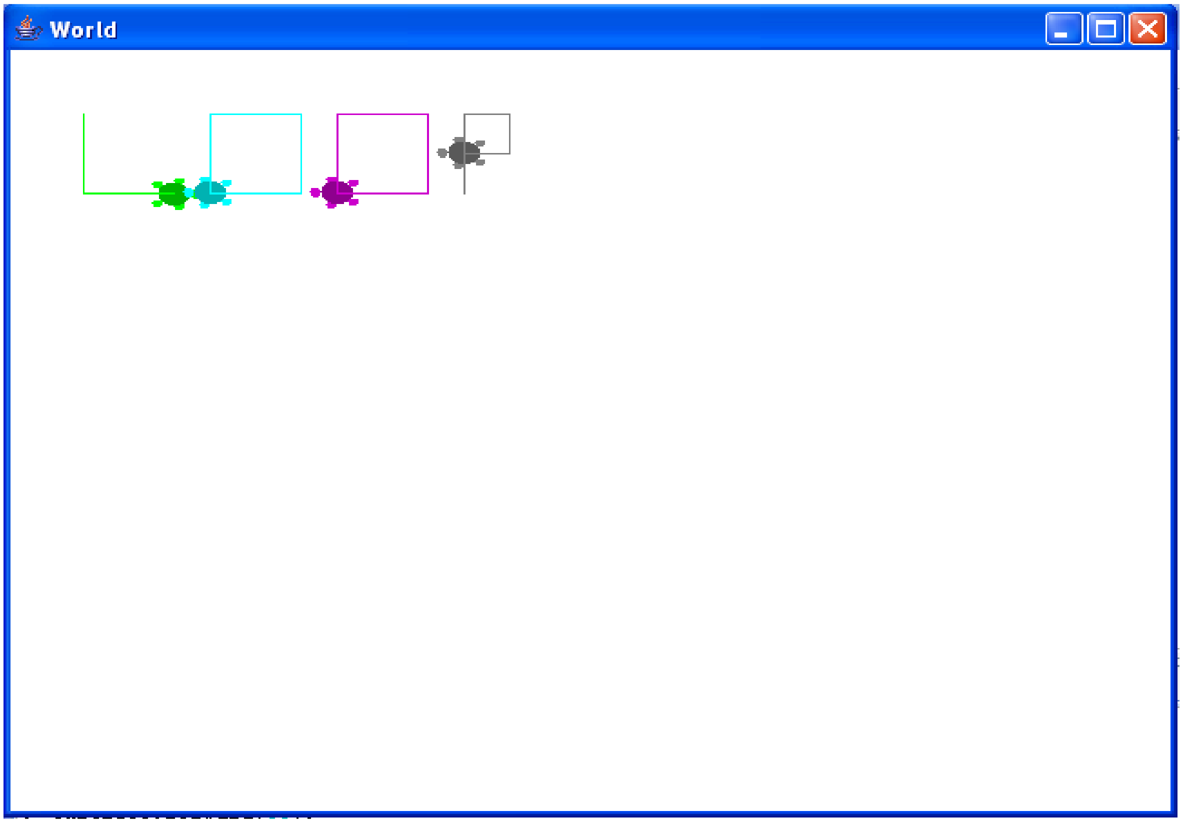
Use following methods to draw Rectangle and Square.

```
Turtle1.turnLeft();      // To Move Left.
Turtle1.turnRght();     // To Move Right
Turtle1.forward(x);      // To Move x steps
```

The final result should be something like this



5. Write the word " LOOP " with 4 different Turtles using methods learned in above question.


World