NMK20703 OBJECT ORIENTED PROGRAMMING LAB MODULE 2

BASICS OF JAVA

FACULTY OF ELECTRONIC ENGINEERING TECHNOLOGY

Universiti Malaysia Perlis

NMK20703 Object-oriented Programming: Lab Module 2

Learning Outcome:

After completing this lab module, students will be able to:

- Create a class in Java
- Declare member variables (attributes) using int, float, double and char, String class and other classes as its data type
- Display text to the monitor and get inputs from the keyboard.
- Use arithmetic operators namely +, -, * and /
- Declare member methods
- Create constructors

TASK 1

Answer these questions

1. Mark the statements below as TRUE or FALSE.

a. In Java, an object is created by using the keyword ***class*.**

b. In the statement **`temp = console.nextInt( );`** , temp must be a variable.

c. The statement ABC.xyz ( ) means that you are calling a method named ABC that belongs to class xyz ( ).

d. For one object or class, there can be many instances of the object.

e. Classes are general implementation of an object. To use a class, you must declare instances of that class.

2. What is the output of the following statements? Assume that a and b are floats, with a = 20 and b = 3.

```
a. System.out.println (“Enter your age” + “>>” );

b. System.out.println (“The value of 3 + 5 = “ + (3+5) + “.” );

   The value of 3 + 5 = 8.

c. System.out.println ( a + " + " + b + " = " + (a+b) );

d. System.out.println ( a + " X " + b + " = " + (a*b) );

e. System.out.println ( a + " / " + b + " = " + (a/b) );

f. System.out.printf ( "%.2f + %.2f = %.3f.", a,b,(a/b));

g. System.out.println( );

h. System.out.println (" a + b = " + (a+b));
```

3. The following statements will run but it has a logical error. Explain briefly what the error is, and suggest a fix.

```
float a = 20;
float b = 3;
System.out.print ( "Nilai a = " + a );
System.out.println(" dan nilai b = " + b + ". Maka a + b = " +
 (a/b));
```

4. Consider the following statements:

```
String str = "Today is better";
int ex;
```

```
char chr;
```

a. What is the value of `chr` by the following statement?

```
a. chr = str.charAt(2);


b. chr = str.charAt(8);
```

b. What is the value of `ex` by the following statement?

```
a. ex = str.length( );

b. ex = str.indexOf("better");

c. ex = str.indexOf(‘T’);

d. ex = str.indexOf(‘t’) + 9;
```

c. What is the value of the following statement?

```
a. System.out.println (str.substring(6,10));


b. System.out.println (str.startsWith ("day"));


c. System.out.println (str.startsWith ("T"));


 d. System.out.println (str.startsWith ("To"));


e. System.out.println (str.endsWith ("."));
```

5. Determine the output of the following code:

```
int x, y;
x = 1;
y = 2;
System.out.println("The output is " + x +
y); System.out.println("The output is " + (x + y) );
```

6. Write Java statements to do the following tasks:

a. Declare two char variables named ch1 and ch2; initiate their values to 1.

b. Display the following:

```
***************
*** SUCCESS ***
***************
```

c. Create an empty class named Gas.

d. Create a class named Bed with two member variables: height as centimetres and weight in kilograms.

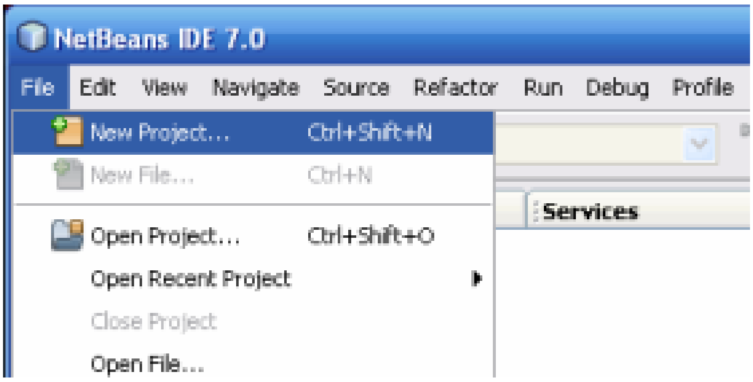
e. Instantiate an object of Bed and name it as myBed.

TASK 2:

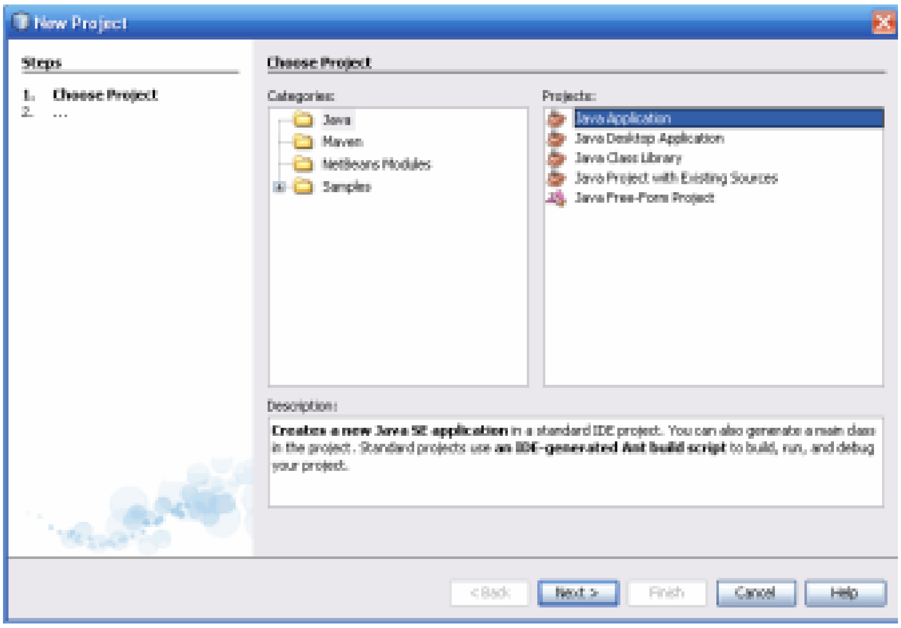
A Java project may consist of one or more classes. When you create a new project in NetBeans, it will start with one **main class** that contains a main method named as `public static void main( ).` A Java program will always **start from the main class**. Sometimes the main class is referred to as a pilot program or test program because all commands that manipulate other classes are defined here.

The following steps will guide you on how to start writing a pilot program in Java. Open your Netbeans IDE and follow the instructions provided.

1. Create a new Java Application in NetBeans.
    a. Launch the NetBeans IDE
    b. In the NetBeans IDE, choose File | New Project.



    c. In the New Project wizard, click Java category and select Java Application as shown in the following figure:



New Project

Steps

1. Choose Project
2. ...

Choose Project

Categories: Java, Maven, NetBeans Modules, Samples

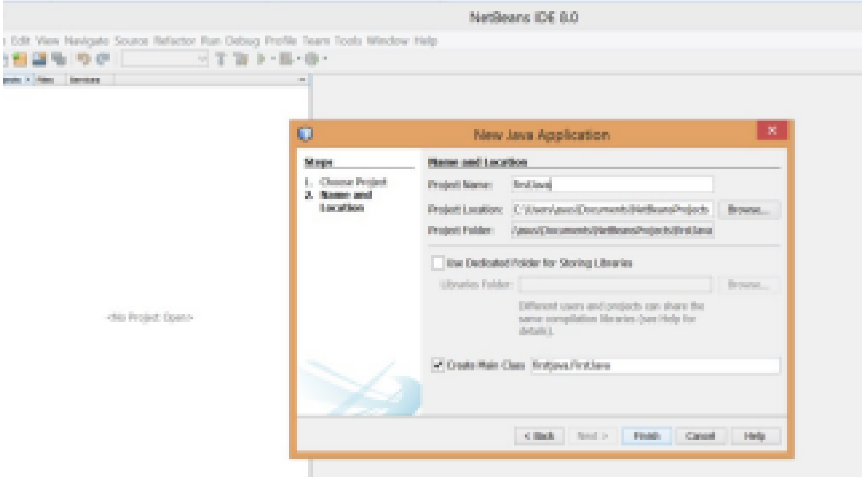
Projects: Java Application, Java Desktop Application, Java Class Library, Java Project with Existing Sources, Java Free-Form Project

Description:

Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

d. In the Name and Location page of the wizard, do the following

- In the Project Name field, type `firstJava`. This will be the package name.
- You can use the default location of your project.If you need to change the location of your project, click the ‘Browse’ button on the right.
- Make sure that the “Create Main Class” checkbox is selected. The main class is important because it is like a controller of your programme. A default name is provided for you.



NetBeans IDE 8.0

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: firstJava

Project Location: C:\Users\jaso\Documents\NetBeansProjects

Project Folder: \jaso\Documents\NetBeansProjects\firstJava

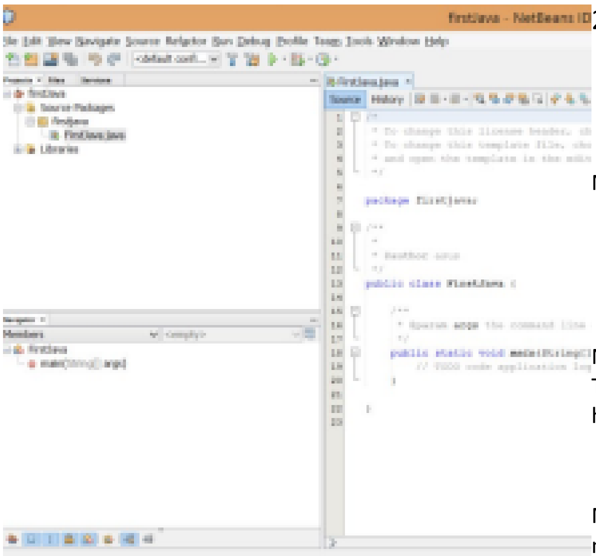
□ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☑ Create Main Class firstjava.FirstJava

e. Click ‘Finish’. A new space with the label FirstJava.java will appear. Here is where you will add all the necessary Java code to define the member variables and member methods.



```
/*
 * To change this license header, ch
 * To change this template file, cho
 * and open the template in the edit
 */

package firstjava;

/**
 *
 * @author [illegible]
 */
public class FirstJava {

    /**
     * @param args the command line
     */
    public static void main(String[]
        // TODO code application log
    }

}
```

2. Save your work. (Press ‘Ctrl’ + ‘S’)

Name of file

Name of class.
This is a main class because it
has a main method

Main
method

3. In the java project you just created, write an application that inputs temperature in degrees Celsius and prints out the temperature in degrees Fahrenheit. Use System.in for input and System.out for output. The formula to convert degrees Celsius to the equivalent degrees Fahrenheit is

Fahrenheit = 1.8 * Celsius + 32

4. Create a new project and write an application to convert centimeters (input) to feet and inches (output). Use JOptionPane for input and output. 1 inch = 2.54 centimeters.

5. Your weight is actually the amount of gravitational attraction exerted on you by the earth. Since the Moon’s gravity is only one-sixth of the earth’s gravity, on the Moon you would weigh only one-sixth of what you weigh on earth. Write an application that inputs the user’s earth weight and outputs her or his weight on Mercury, Venus, Jupiter, and Saturn. Use your preferred choice for input and output. Use the values in this table

Planet	Multiply the Earth Weight by
Mercury Venus Jupiter Saturn	0.4 0.9 2.5 1.1

TASK 3

Implement the following class in Java. Put the class in a package named mypackage.

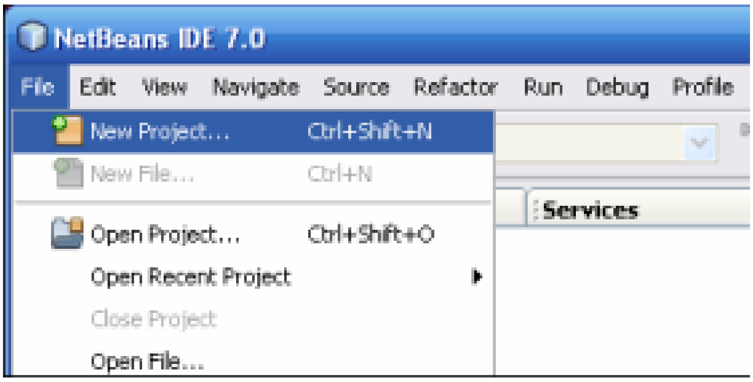
Coordinate

x : integer
y : integer

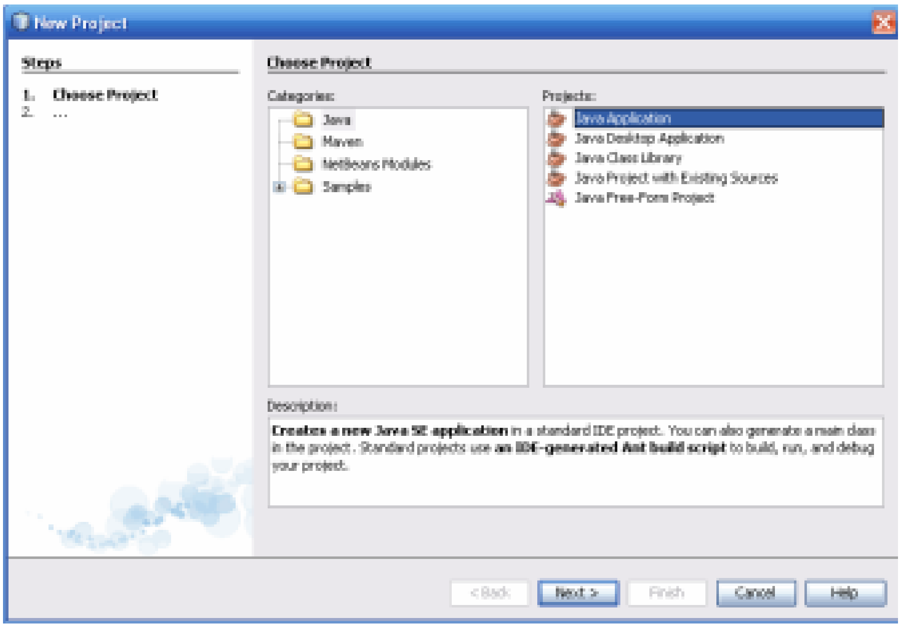
getX( )
getY( )
displayCoordinate( )
setX (int)
setY(int)
setToOrigin( )
Coordinate( )
Coordinate (int, int)

Solution:

6. Create a new Java Application in NetBeans.
    a. Launch the NetBeans IDE
    b. In the NetBeans IDE, choose File | New Project.



    c. In the New Project wizard, expand the Java category and select Java Application as shown in the following figure:



Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

d. In the Name and Location page of the wizard, do the following

- In the Project Name field, type `myPackage`

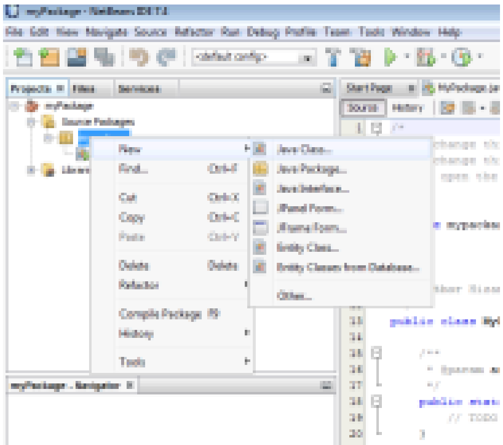
- In the Create Main Class field, type `mypackage.TestPackage`

- Leave the Set as Main Project checkbox selected.

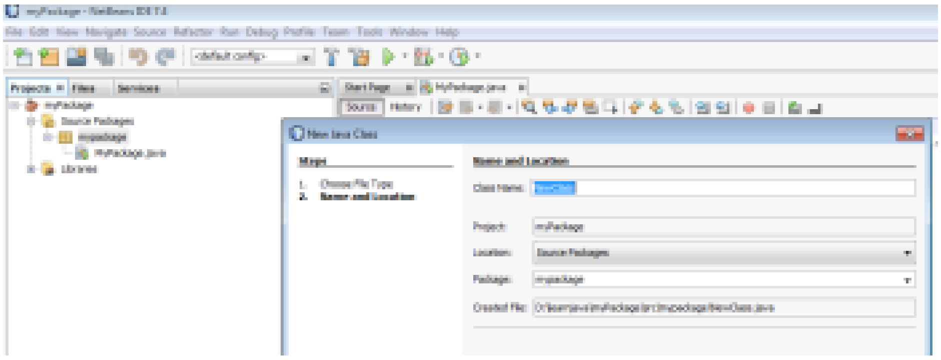
e. Click Finish.

7. Create the class

a. In the navigator pane on the upper left, right click on the name of your project (mypackage), click 'New' click 'Java Class..'



b. In the Class Name field, type `Coordinate`.



Class Name: NewClass

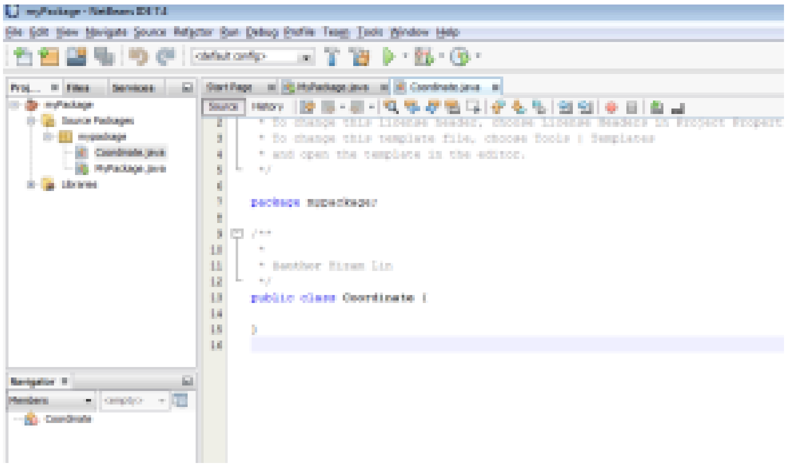
Project: myPackage

Location: Source Packages

Package: mypackage

Created File: [illegible]

c. Click 'Finish'. A new space with the label Coordinate.java will appear. Here is where you will add all the necessary Java code to define the member variables and member methods.



```
 * To change this license header, choose License Headers in Project Propert
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package mypackage;

/**
 *
 * @author [illegible]
 */
public class Coordinate {

}
```

8. Define member variables x and y as integers. Remember that variables are usually concealed/blocked from access by other methods outside this class.

```
private int x;
private int y ;
```

9. Define the member methods.

   a. getX ( )

      - this method returns the value of x that is held by the object
      - since x is an integer, getX will also return an integer

```
int getX ( )
{
    return x;
}
```

   b. getY ( )

      - this method returns the value of y that is held by the object
      - since x is an integer, getY will also return an integer

```
int getY ( )
{
    return y;
}
```

   c. displayCoordinate ( )

      - this method displays the coordinate (x, y).
      - Usually if a methods job is to display something, it will have a void return type. -

      ** Can you specify what is the difference of getX( ) and displayCoordinate() ?

```
void displayCoordinate ( )
{
    System.Out.println (“Koordinat untuk objek ini ialah: (“
+ x + “, “ + y + “) “);
}
```

   d. setX (int a)

      - Assigns the value of ‘a’ (given by user) into x in the object.
      - setX will do the task but does not give any feedback, so the return type is void.

```
void setX ( )
{
    x = a;
}
```

   e. setY (int a)

      - Assigns the value of ‘a’ (given by user) into y in the object.

- setY will do the task but does not give any feedback, so the return type is void.

```
void setY ( )
{
    y = a;
}
```

f. setToOrigin ( )

- Origin is located on (0,0), meaning that this method will assign x = 0 and y = 0; - setToOrigin will do the task but does not give any feedback, so the return type is void.

```
void setToOrigin ( )
{
    x = 0;
    y = 0;
}
```

g. Coordinate ( )

- This is the constructor method without any parameters. It is called when you create a new instance/objct of the class.
- Constructors do not have return type.
- This constructor create a new point *located on origin (0,0).*

```
Coordinate ( )
{
    x = 0;
    y = 0;
}
```

h. Coordinate (int, int)

- This is the constructor method with parameters. It is called when you create a new instance/objct of the class.
- Constructors do not have return type.
- This constructor create a new point *located on a point specified by the user by giving the value of a and b.*

```
Coordinate (int a, int b)
{
    x = a;
    y = b;
}
```

10. Save and Run the project. ▶

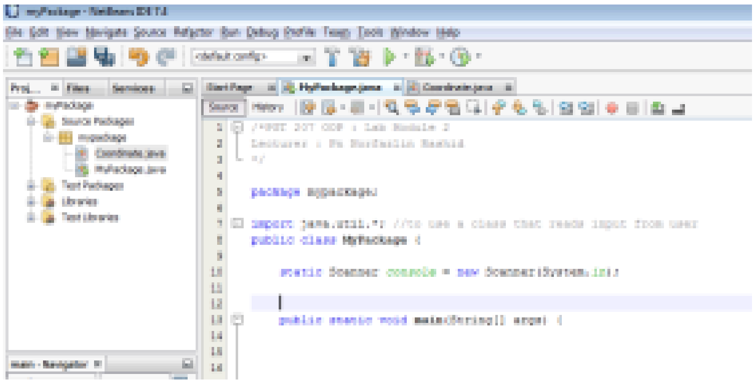
11. The complete class is as below:

```
package mypackage;


public class Coordinate {
 private int x;
 private int y ;


 public int getX ( )
{
    return x;
}

 int getY ( )
{
```

```
    return y;
```

```
}

 void displayCoordinate ( )
{
    System.out.println ("Koordinat untuk objek ini ialah: (" + x + ", " + y +
")"); }

 void setX (int a)
{
    x = a;
}

 void setY (int a)
{
    y = a;
}


 void setToOrigin ( )
{
    x = 0;
    y = 0;
}

 Coordinate ( )
{
    x = 0;
    y = 0;
}

 Coordinate (int a, int b)
{
    x = a;
    y = b;
}

}
```

12. Even if we have successfully build (run) the class, we do not see any output because a class is a blueprint or a concept in Java. To check whether our methods are functioning properly, we need to write a 'pilot program' or a 'test program'. In this program, we make sure that all methods in the class are called for. The best test program will call each method several times with different input. To create a test program, open the class that contains `public static void main( )`. In this lab module, the name of the class file is MyPackage.java



```
/*[illegible] OOP : Lab Module 2
Lecturer : [illegible]
*/

package mypackage;

import java.util.*; //to use a class that reads input from user
public class MyPackage {

    static Scanner console = new Scanner(System.in);

    |
    public static void main(String[] args) {
```

13. Rewrite the following pilot program in myPackage.java

```
package mypackage;

import java.util.*; //to use a class that reads input from user
public class MyPackage {

 static Scanner console = new Scanner(System.in);
 public static void main(String[] args) {

 int x,y;

 System.out.println("Let us create 3 coordinates.");
 System.out.println();
 //================Coordinate 1==================================
//Get the value of x and y for coordinate 1
        System.out.println("COORDINATE #1");
 System.out.println("Input the value of x: ");
 x = console.nextInt();
 System.out.println("Input the value of y: ");
 y = console.nextInt();

 //create a new instance for Coordinate 1
 Coordinate coord1 = new Coordinate (x, y);


 //================Coordinate 2==================================
//Get the value of x and y for coordinate 2
 System.out.println();
  System.out.println("COORDINATE #2");
 System.out.println("Input the value of x: ");
 x = console.nextInt();
 System.out.println("Input the value of y: ");
 y = console.nextInt();

 //create a new instance for Coordinate 2
 Coordinate coord2 = new Coordinate (x, y);

 //================titik 3==================================
//Get the value of x and y for coordinate 2
 System.out.println();
  System.out.println("COORDINATE #3");
 System.out.println("Input the value of x: ");
 x = console.nextInt();
 System.out.println("Input the value of y: ");
 y = console.nextInt();

 //create a new instance for Coordinate 3
 Coordinate coord3 = new Coordinate ();


 //Display initial values
 System.out.println();
 System.out.println("The coordinates values after creation.");
System.out.println("Coordinate 1 --> " + coord1.getX() + " " +
coord1.getY());
  System.out.println("Coordinate 2 --> " + coord2.getX() + " " +
coord2.getY());
  System.out.println("Coordinate 3 --> " + coord3.getX() + " " +
coord3.getY());

 //Using other methods
       System.out.println("****Other methods will be
implemented.***");  System.out.println("****Jeng Jeng Jeng***");

       try {
 Thread.sleep(1500);
       }
 catch (InterruptedException ex)
       {
       // do nothing
       }

 coord1.setToOrigin();
 coord2.setX(5);
 coord3.setY(coord3.getY() + 10);

 //display current values
 System.out.println(); //print a blank line (langkau baris)
System.out.println();
 System.out.println();

 System.out.print("COORDINATE 1:");
 coord1.displayCoordinate();

 System.out.print("COORDINATE 2:");
 coord2.displayCoordinate();

 System.out.print("COORDINATE 3:");
 coord3.displayCoordinate();


 }
```

```
}
```

14. Save and run the project.

Exercise: Based on what you have done, answer the following questions

1. What is the purpose of having two constructors? Is it compulsary?

2. getX() and getY() are similar to displayCoordinate( ), which is enabling the user to know the value of x and y. However, they are actually doing two different task- getX() and getY( ) returns the value of x and y, whereas displayCoordinate() prints a message to the monitor. Evaluate the situations that are best for each task.
   [maknanya: masa bila sesuai guna ‘return a value’ macam getX() dan getY(), dan apakah pula situasi yang sesuai untuk guna ‘print a message’]

3. From the pilot program (as in 8), determine ALL statements that are used to read user input from the keyboard.
   [senaraikan: salin saja line of code yang berkenaan]

4. When you first create titik3 in the pilot program, is the coordinate displayed as you expect it? Explain what happens to titik3.

5. What is the difference in the output for System.out.println() and System.out.print() ?

6. Add another java class in myPackage project (yang awak buat tadi) based on the class diagram below:
   (You may need to refer to your textbook or the online Java Tutorial for some of the syntax)

Car

licensePlate : String
maxSpeed: integer
currentSpeed: integer = 0
Coordinate currentLocation

getLicensePlate( )
displayLicensePlate( )
Car (String, int, x, y) //new object will be given license plate, max speed and location
drive (int) //drives at a constant speed (halaju malar)
stop ( )
getMaxSpeed ( )
getCurrentSpeed ( )
displayCurrentLocation ( )
speedUp (int) //increase the current speed by a given value
slowDown (int) //decrease the current speed by a given value

7. Modify the pilot program so that it tests the Car class.