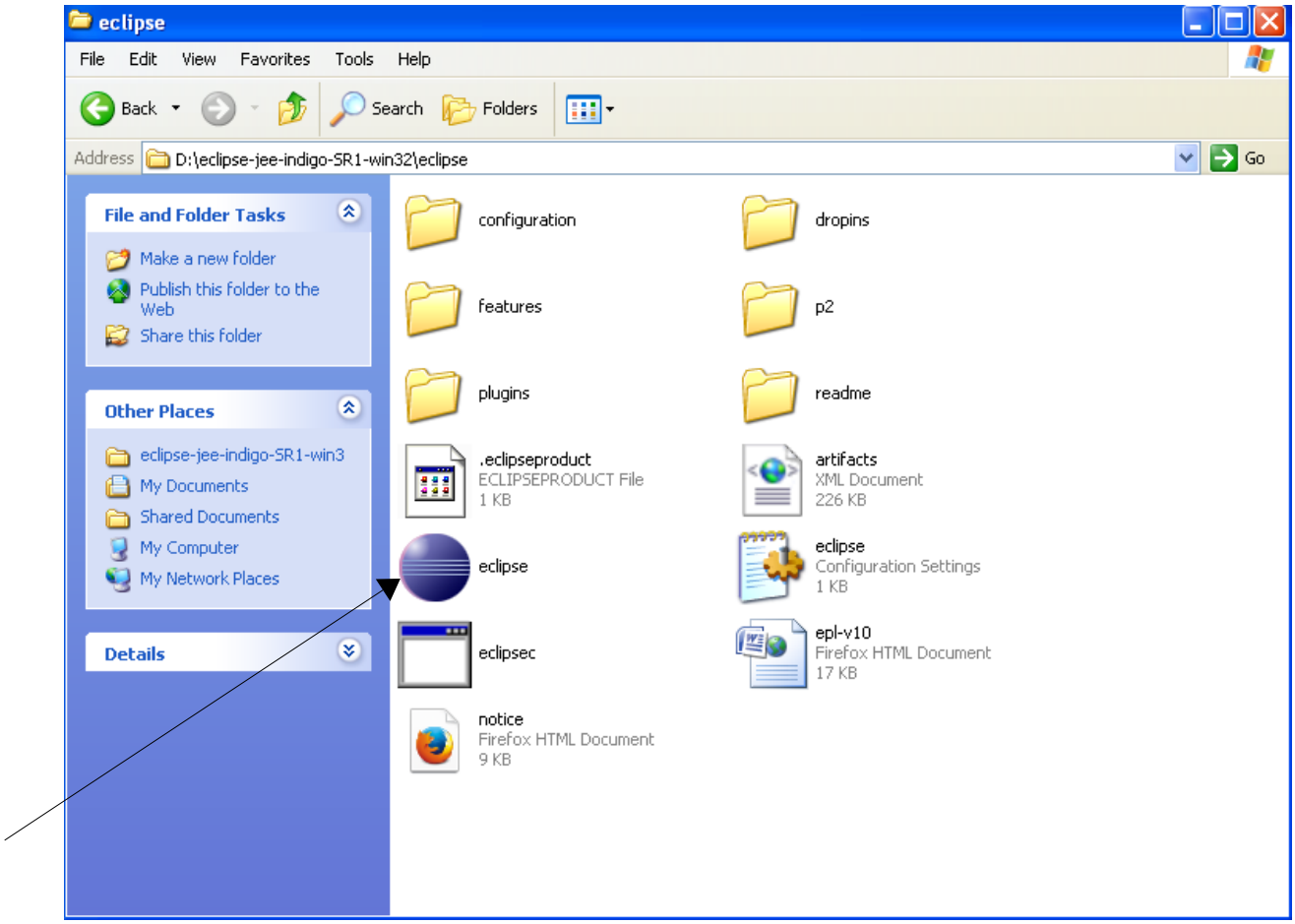
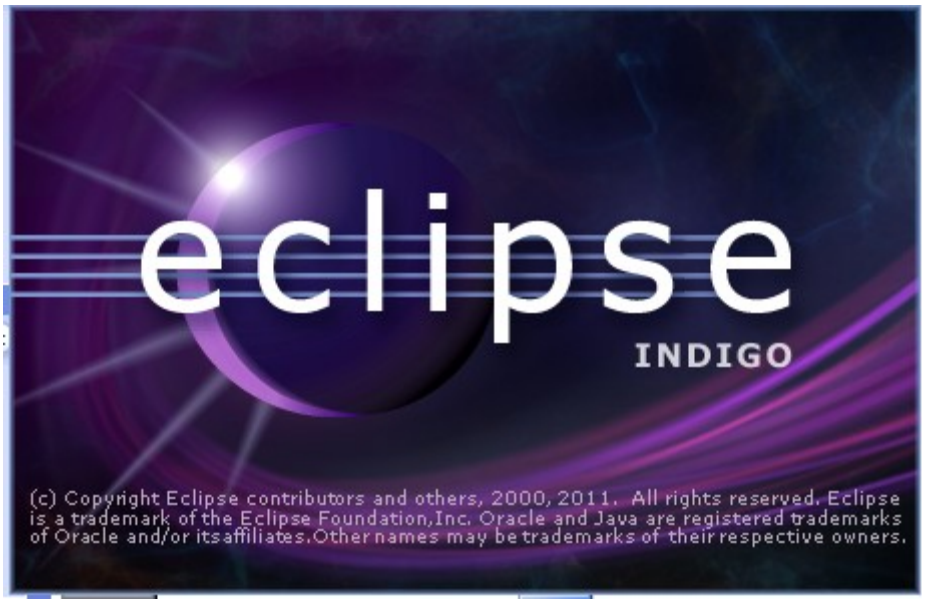
Introducing Eclipse ID

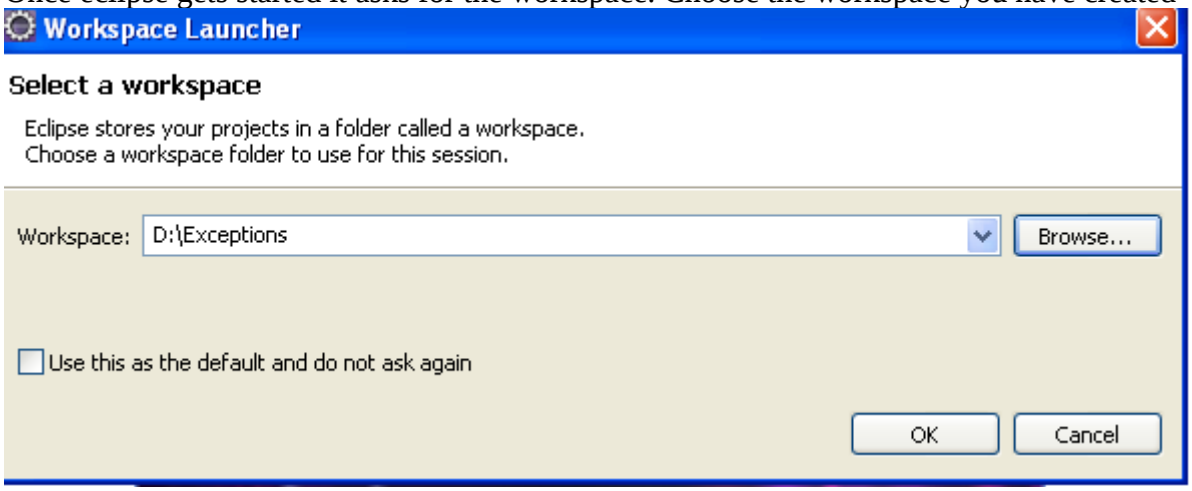
Steps:

1. Create a workspace (folder)(for eg in : d drive)
2. Open Eclipse
   Go to eclipse folder and double click on eclipse icon

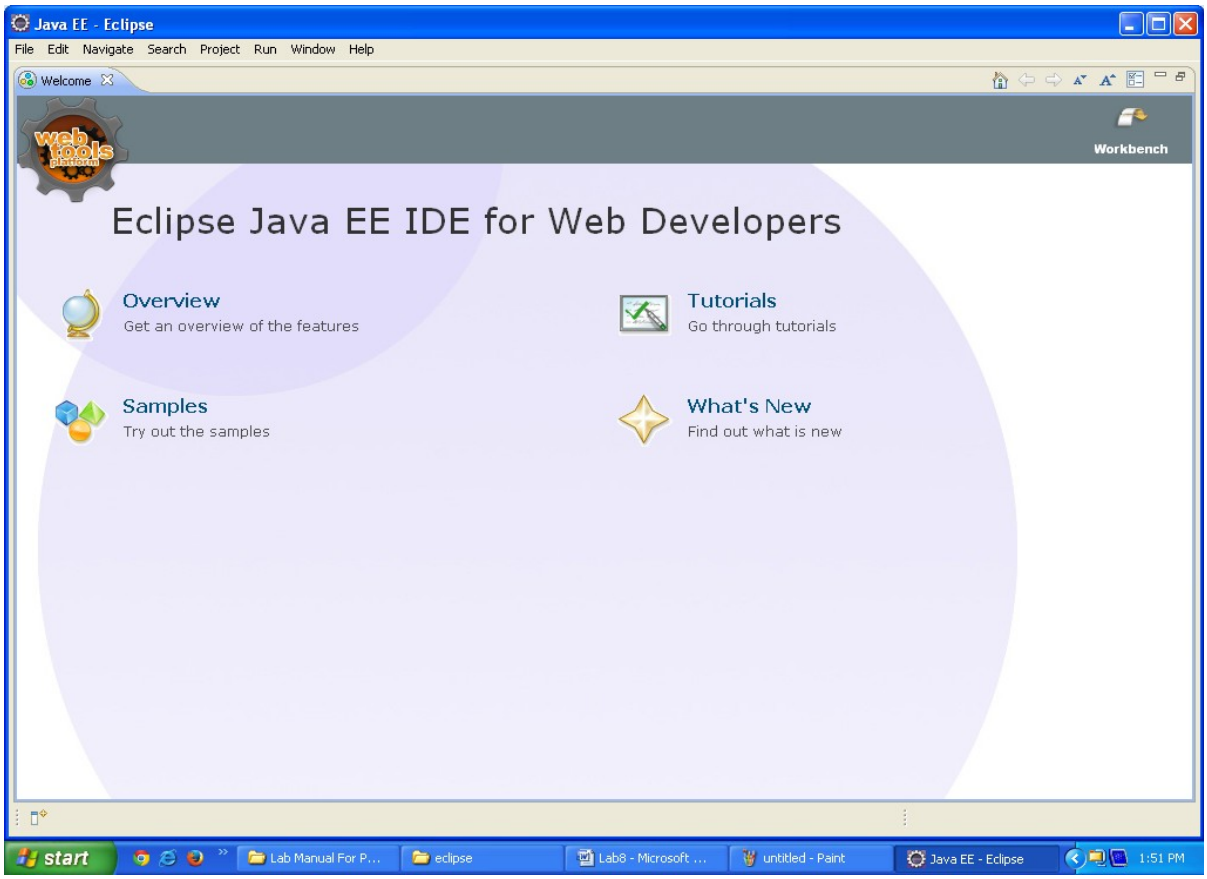




Once eclipse gets started it asks for the workspace. Choose the workspace you have created

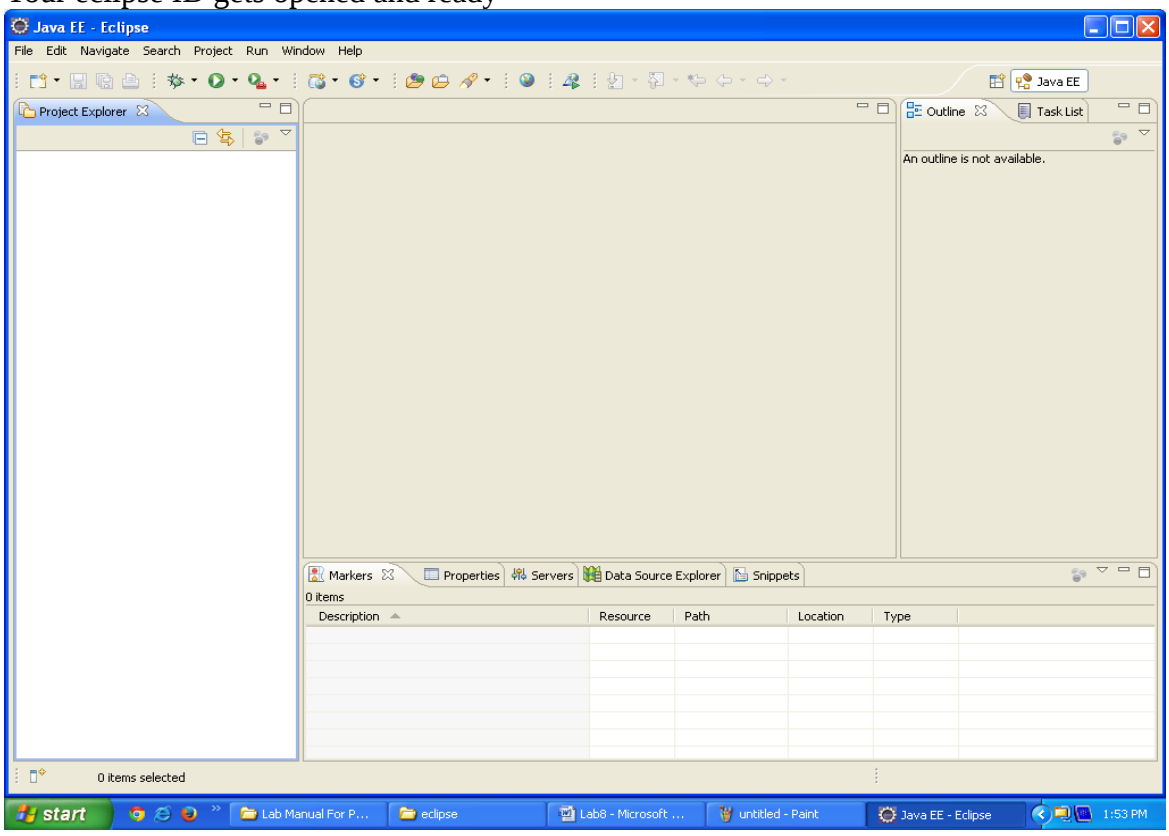


Click ok …

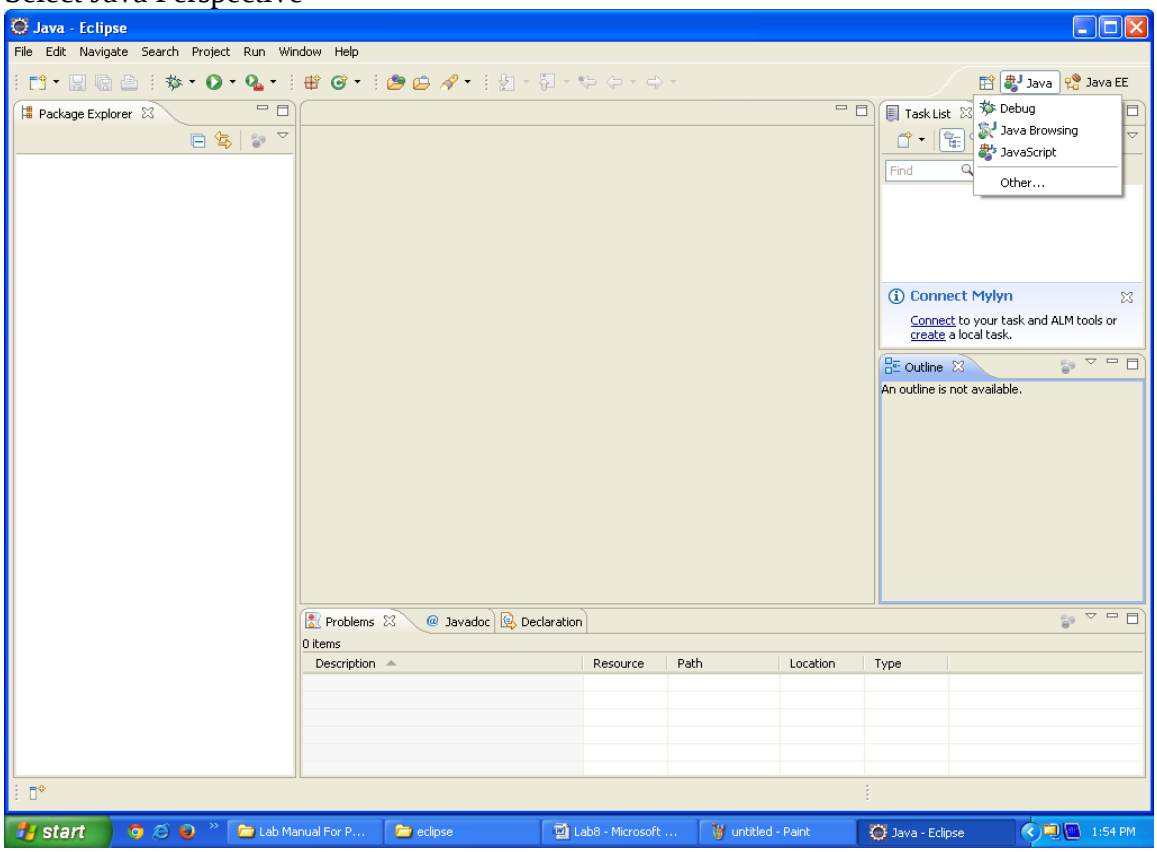


Click on cross button

Your eclipse ID gets opened and ready

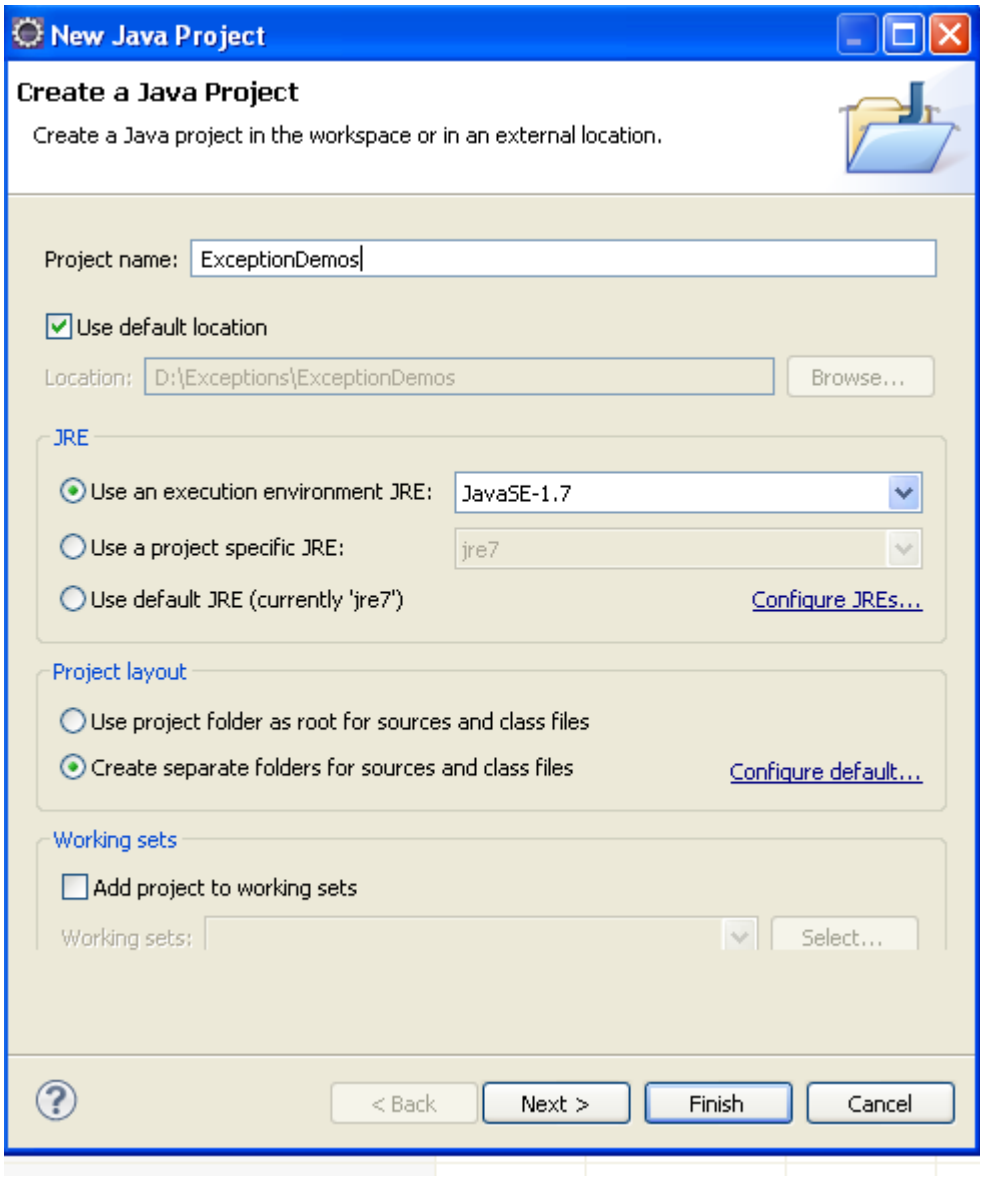


Select Java Perspective

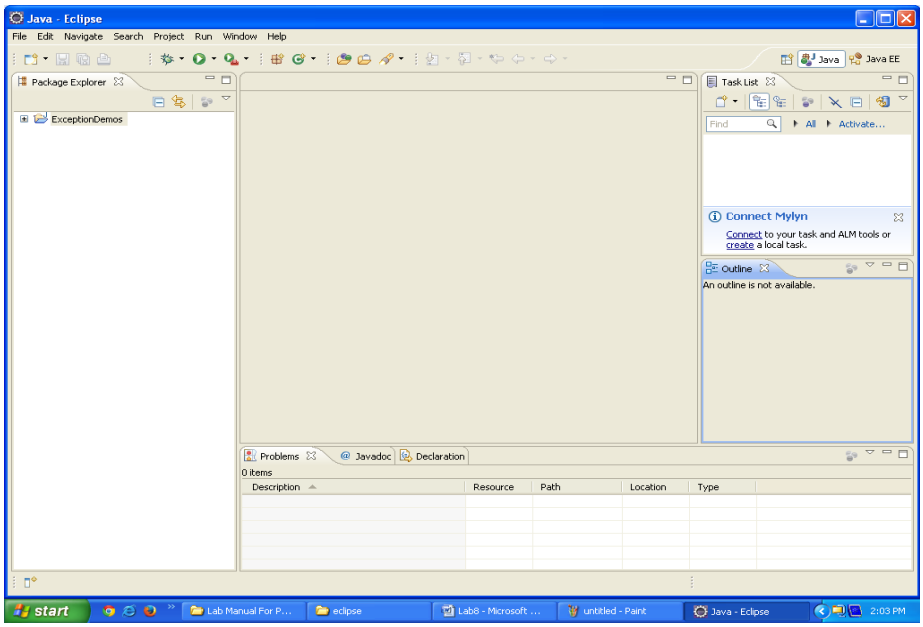


Create a new Project

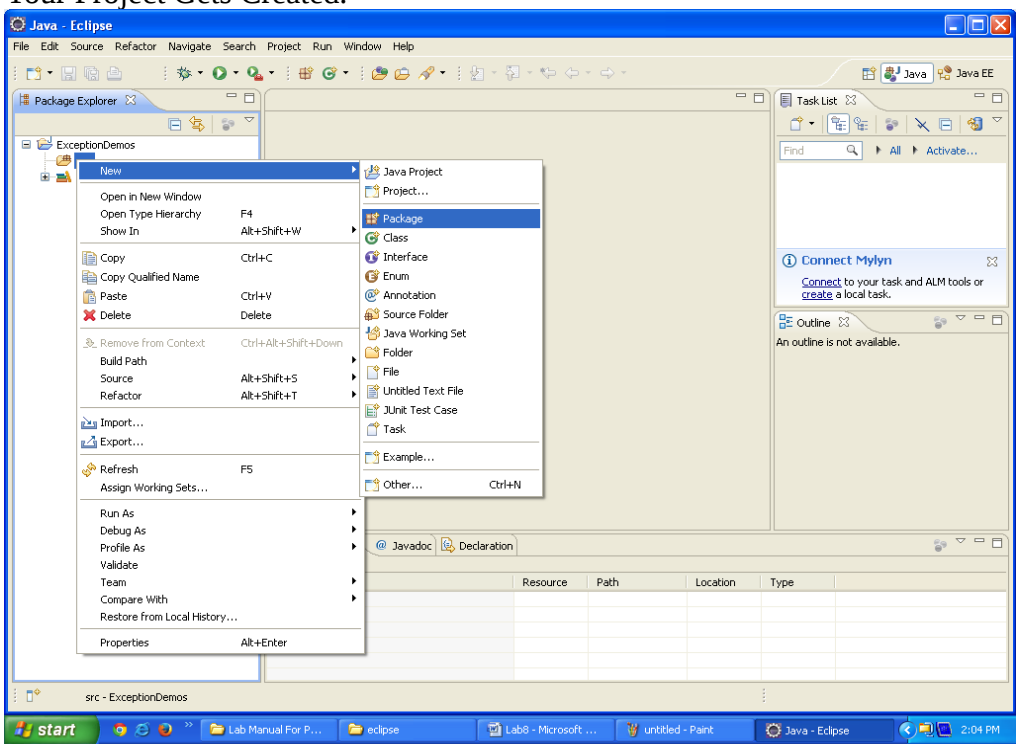
File -> New ->Java project



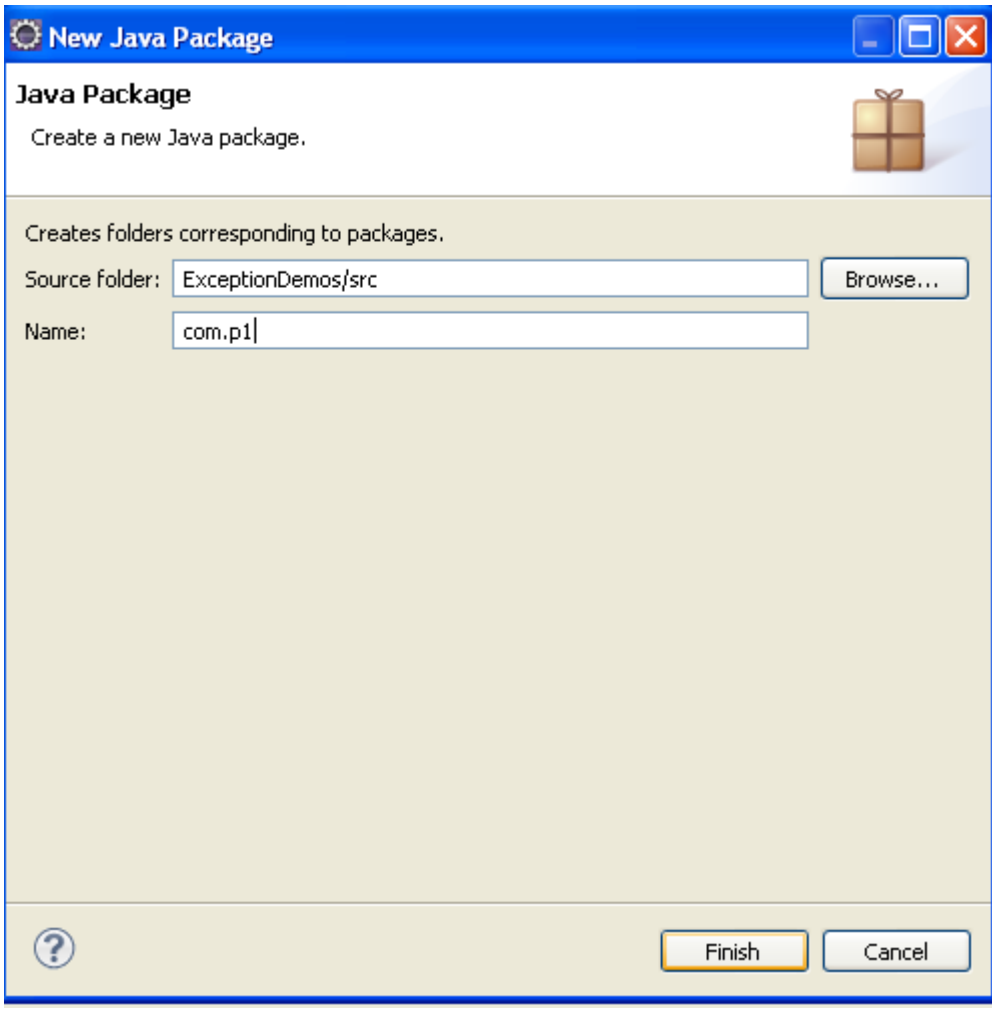
Click on Next- > Finish



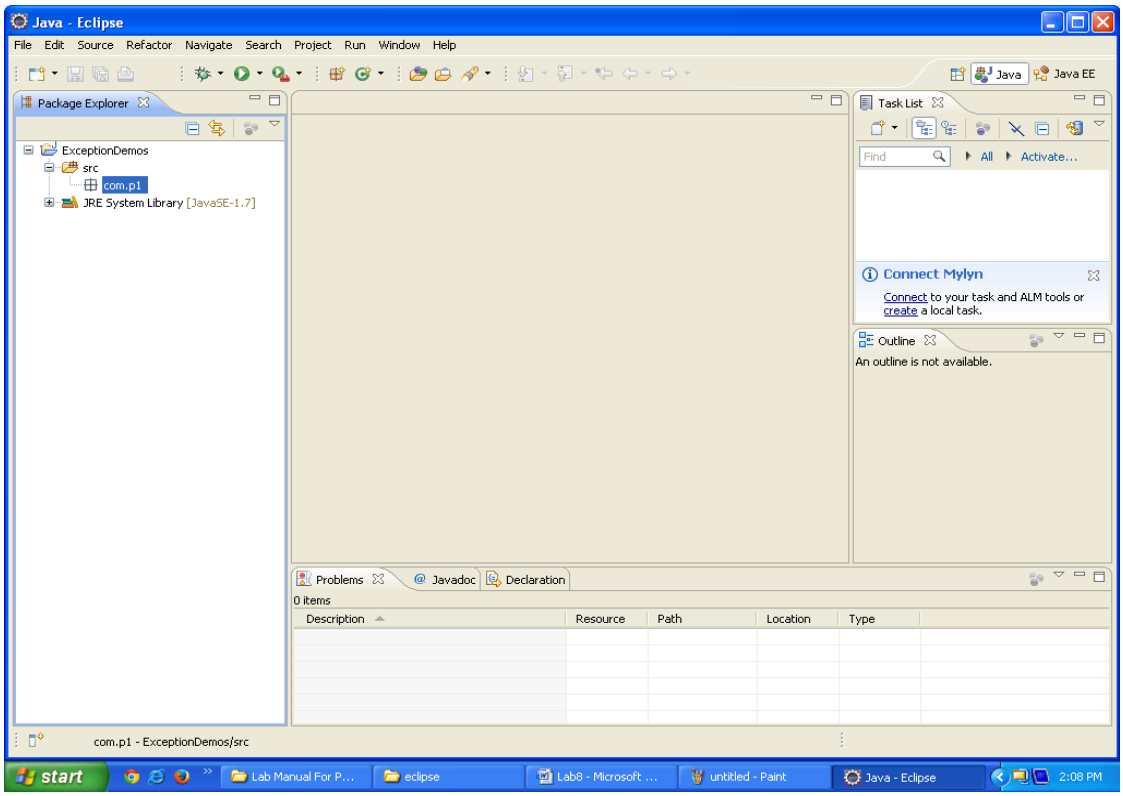
Your Project Gets Created.



Create a package by clicking on
src -> New ->Package

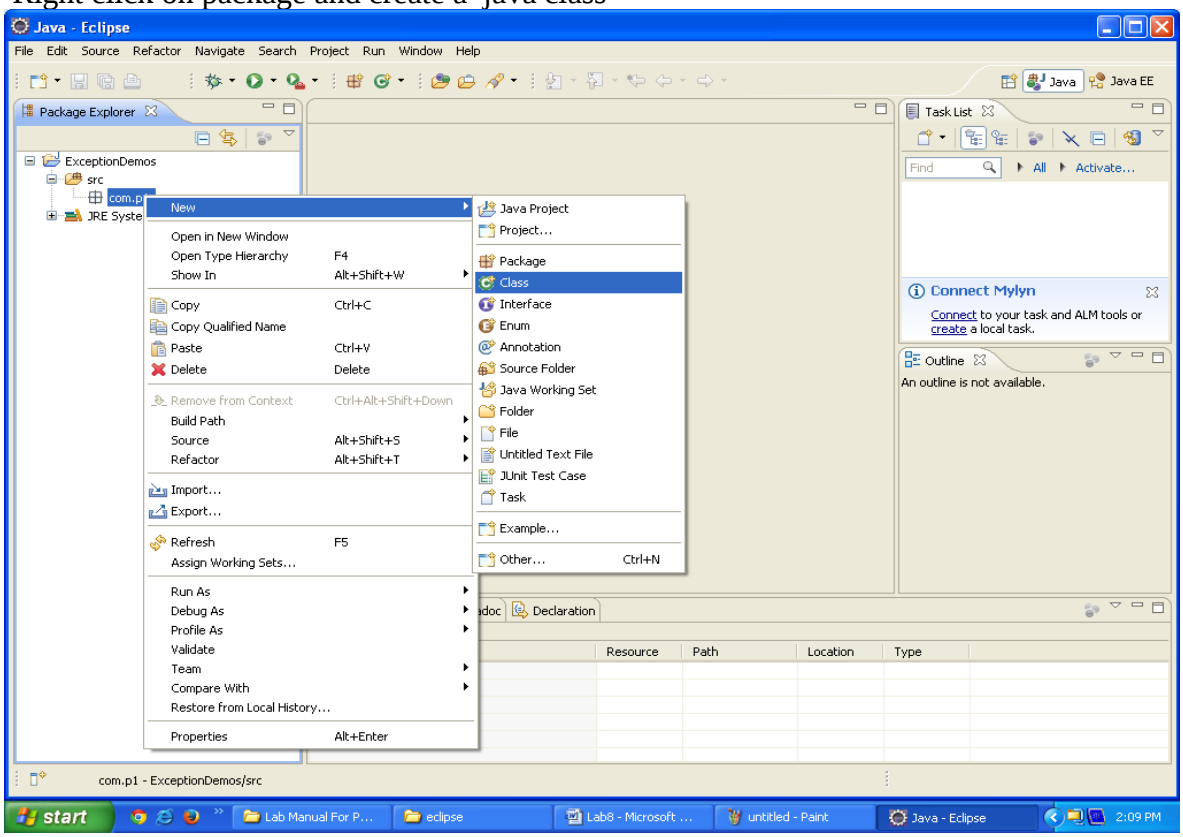


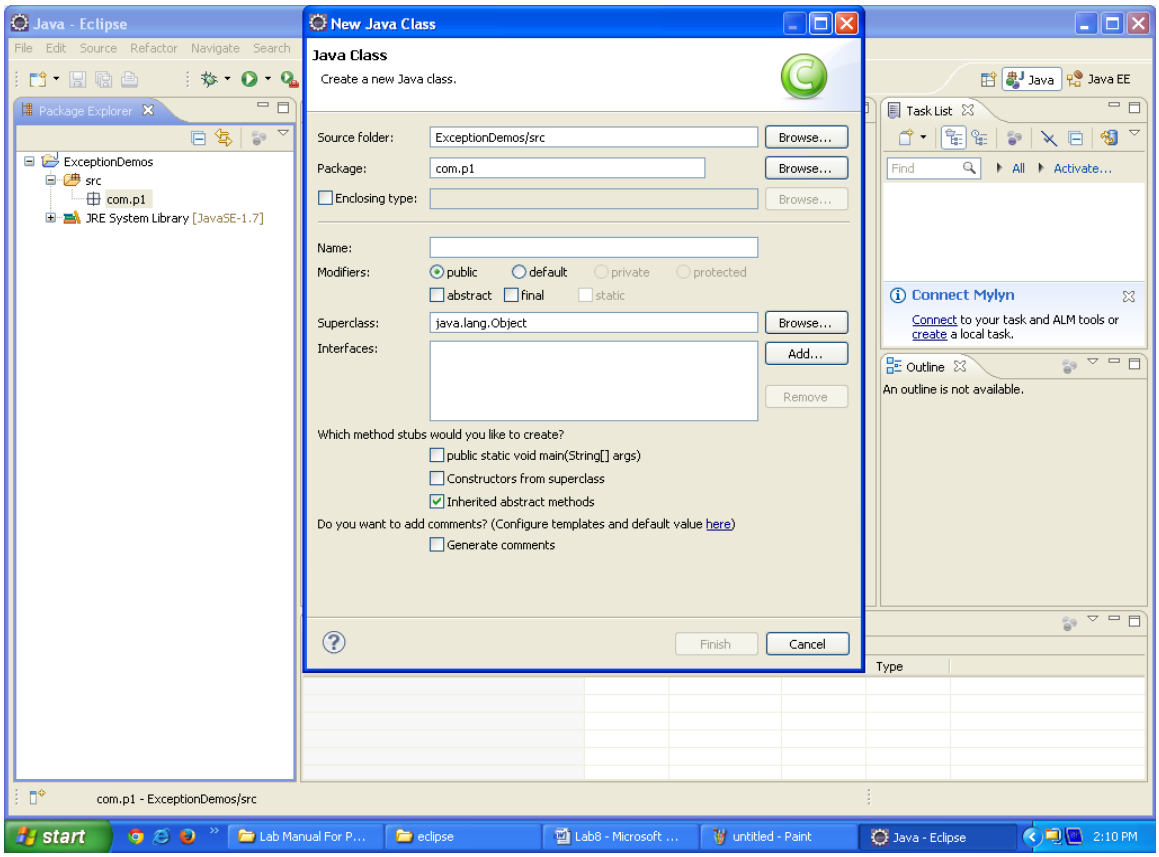
Give name like com.p1 as package name and click on Finish.

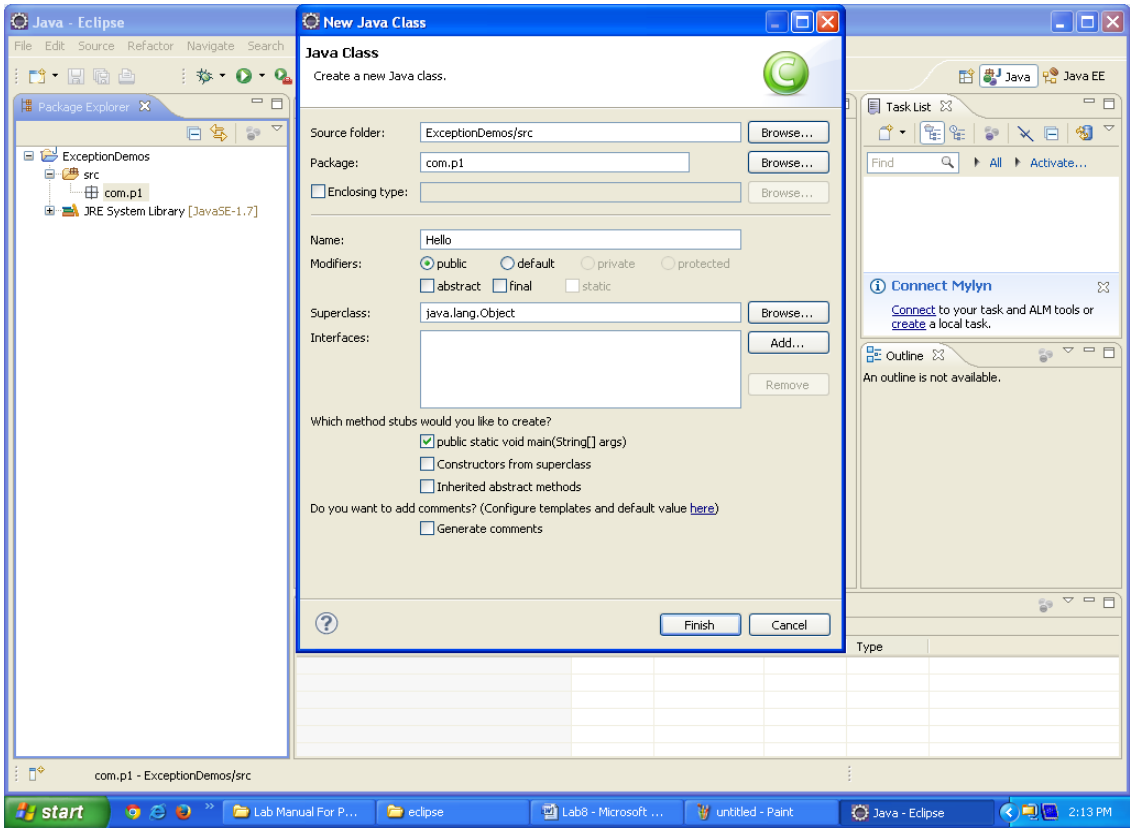


Package got created

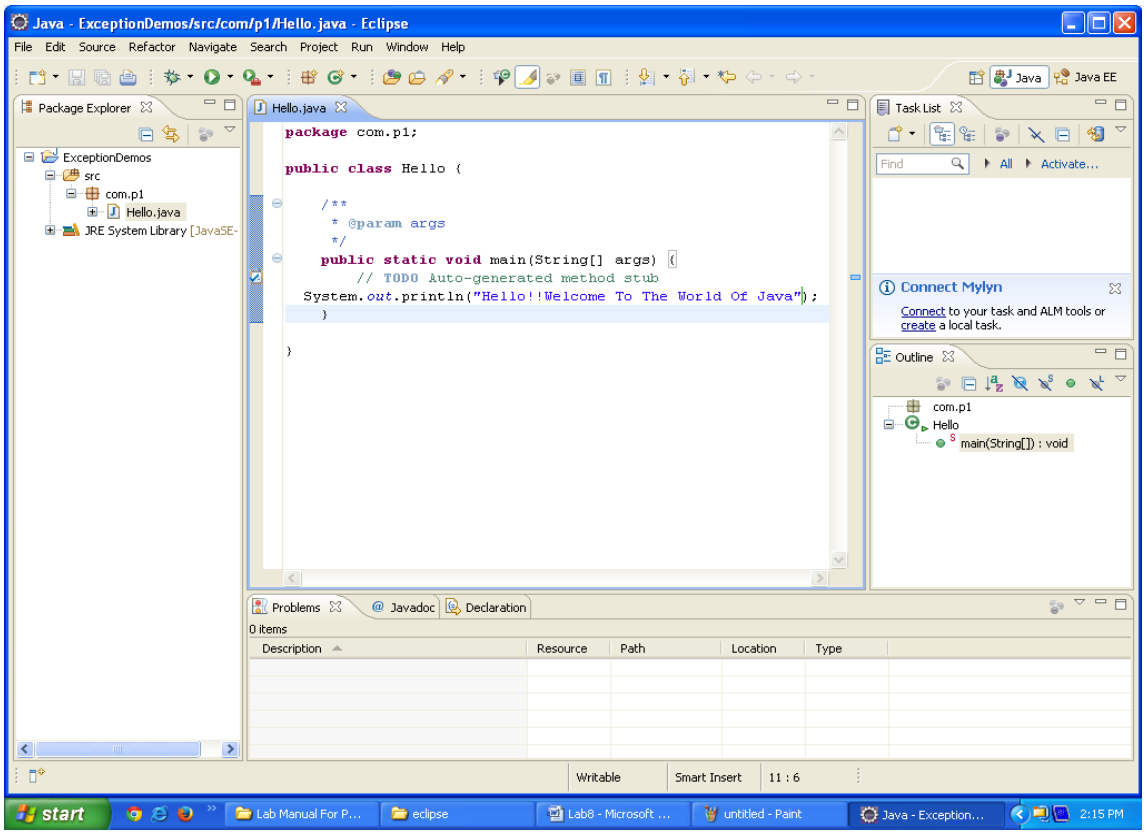
Right click on package and create a java class







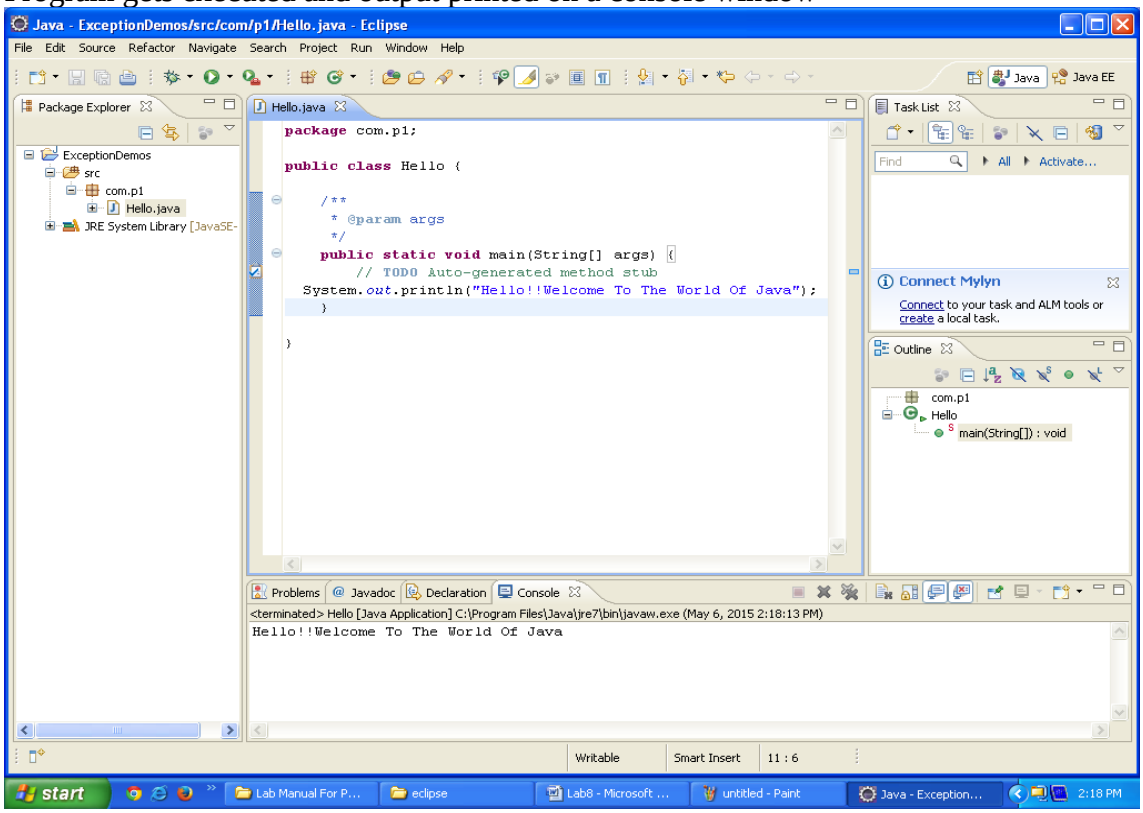
Enter the name of the class as Hello and click on Finish



Execute and run the program

Run -> Run As -> Java Application

Program gets executed and output printed on a console window



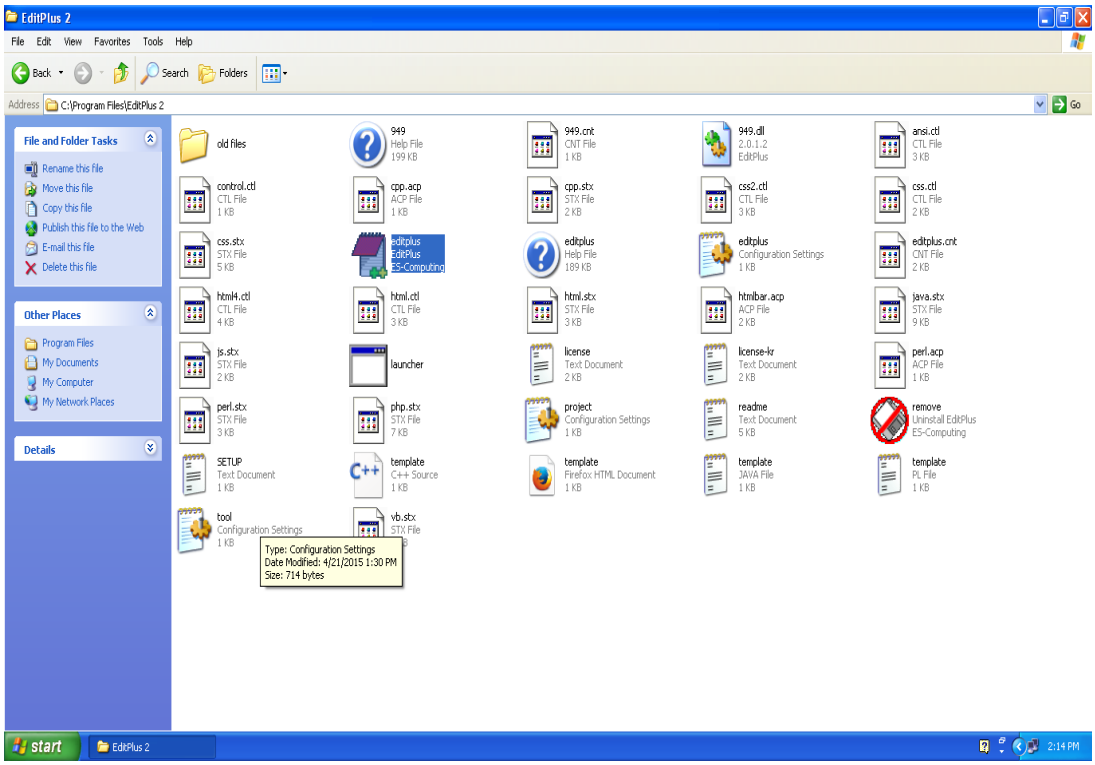
Done with Eclipse….

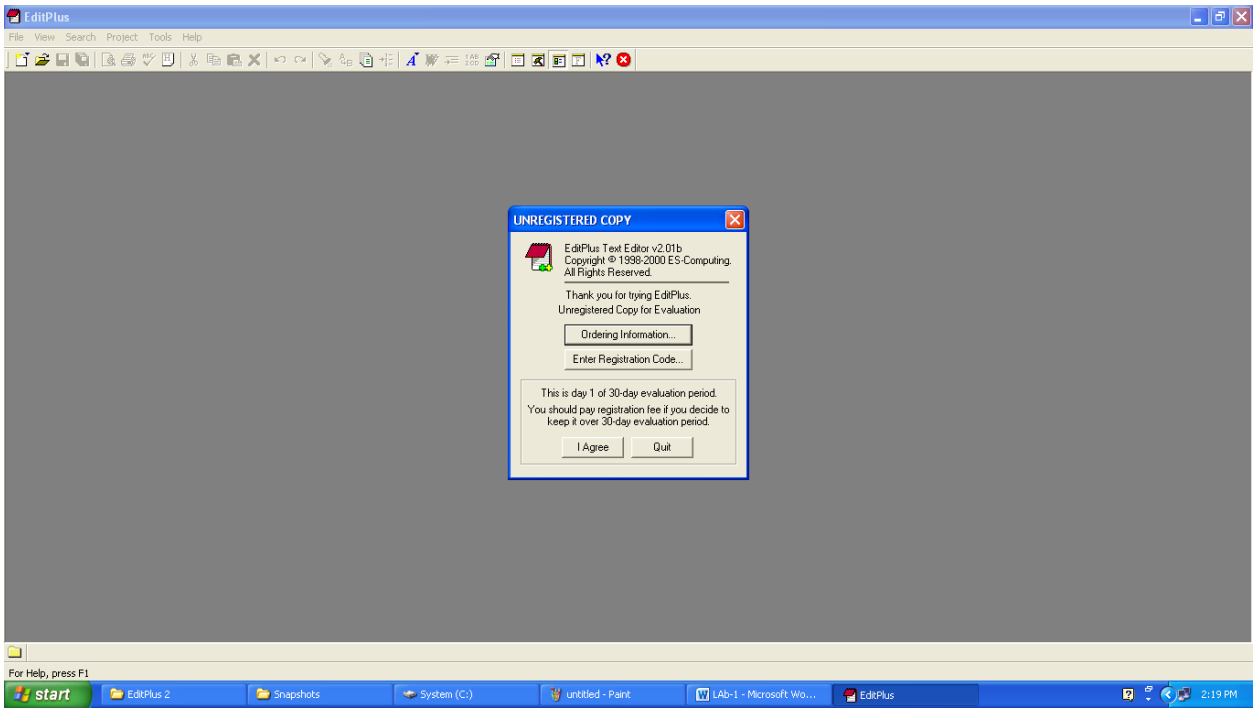
CORE JAVA

Lab-1 Assignments

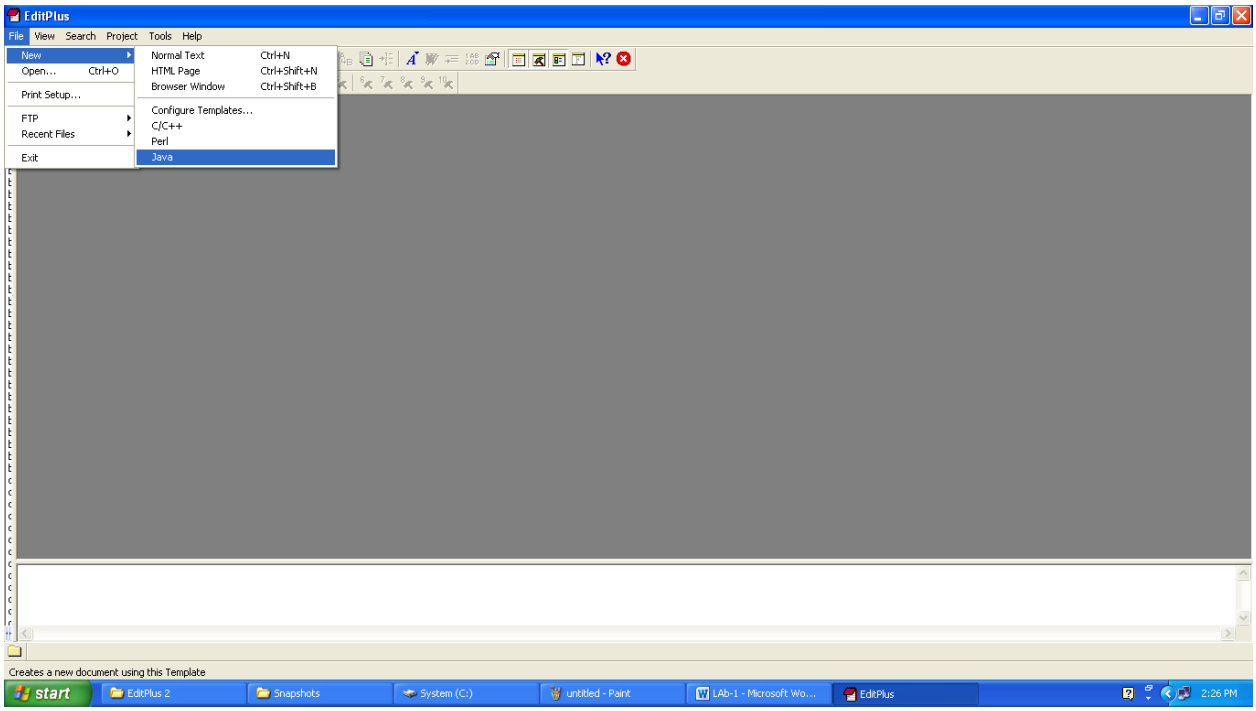
Writing First Java Program

1. Click on start->My Computer-> System (C) -> Program Files -> Edit Plus 2..
2. Double click on editplus icon

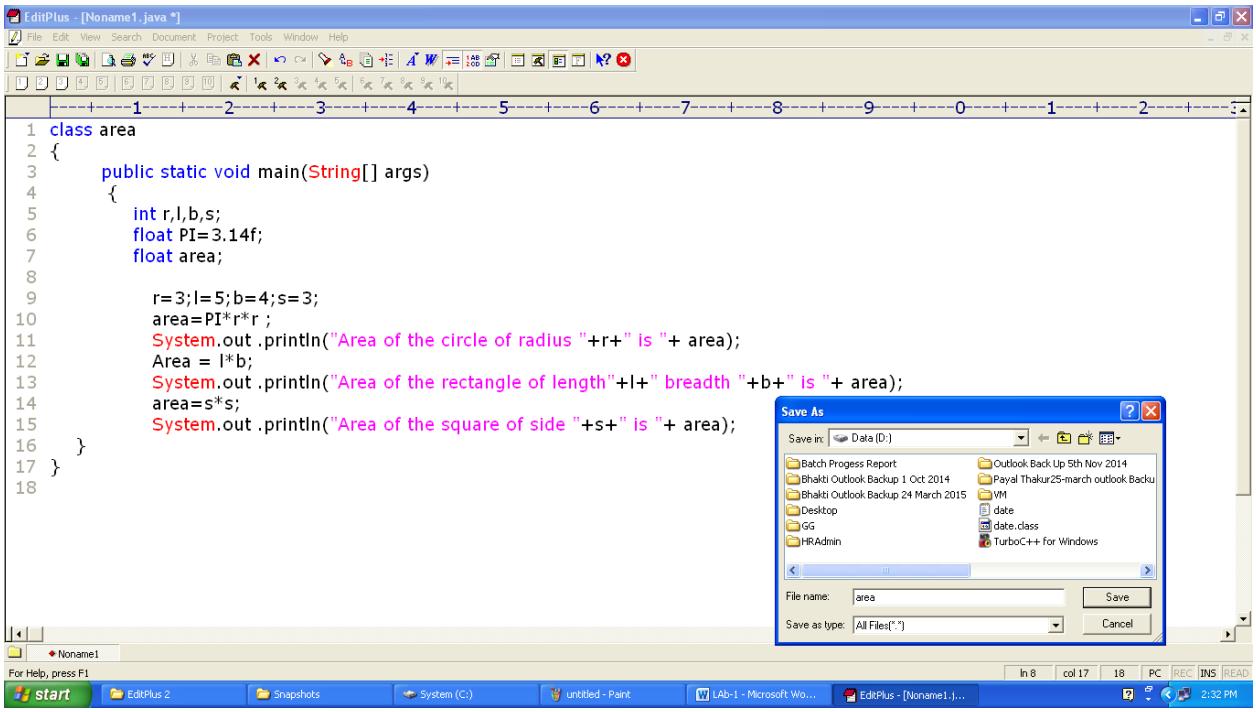




Click on I Agree



Click on
File -> New- > Java
Write your java program in the editor

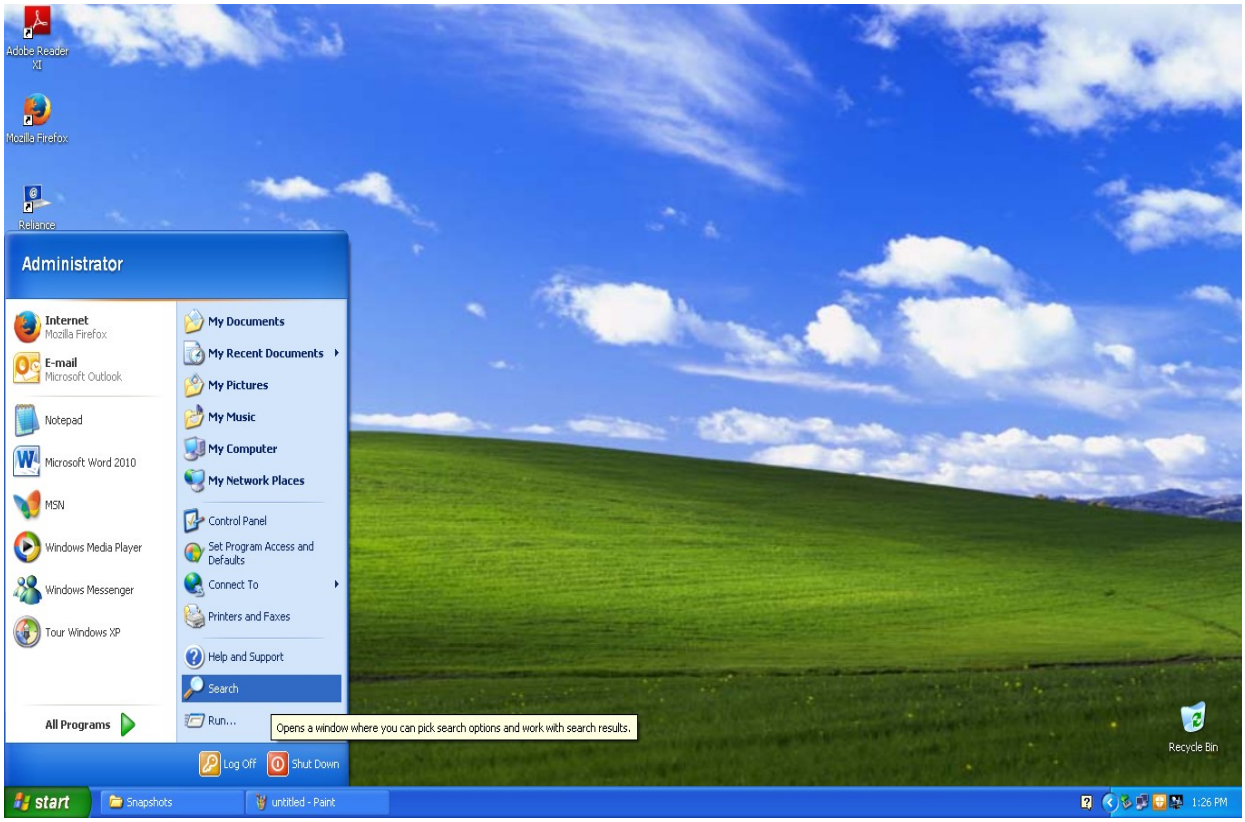


Save the file as area.java (saved in D drive)

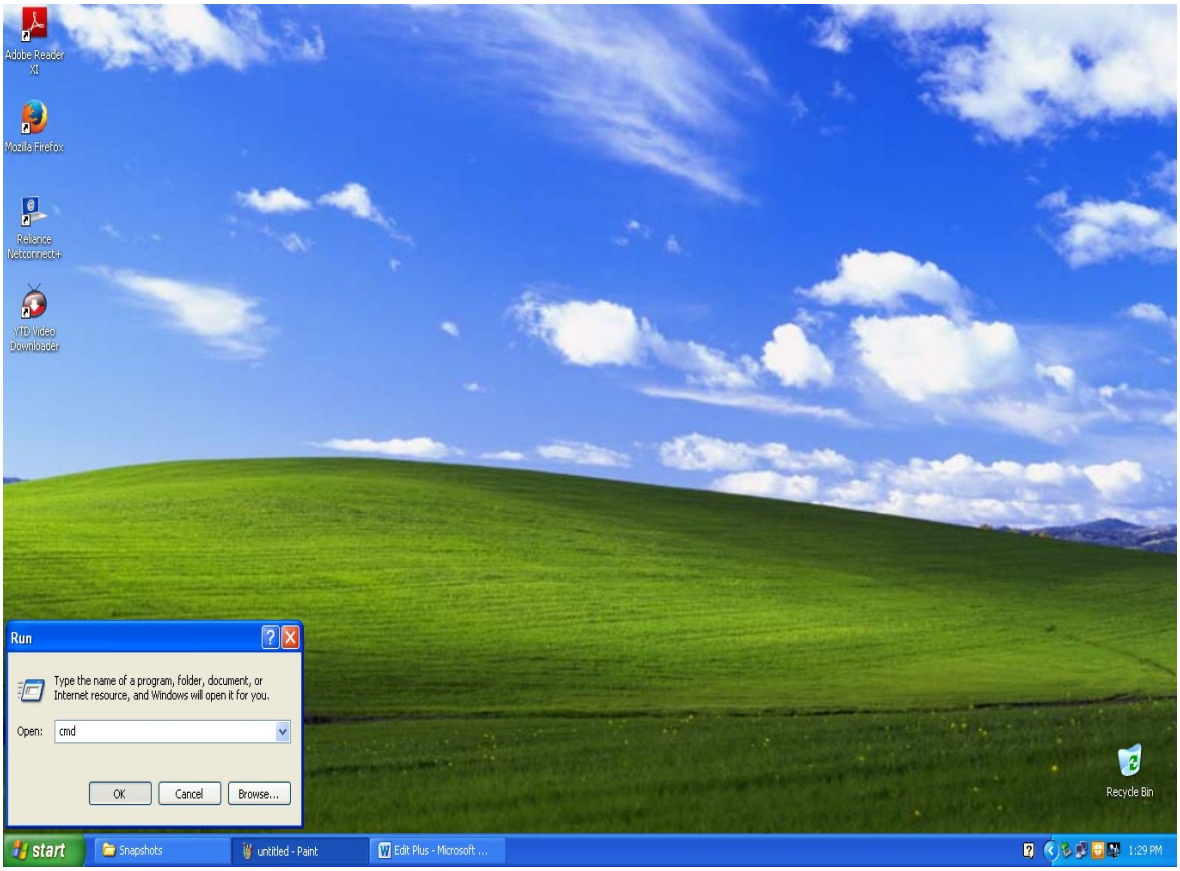
Compile and Run Java Program on Command Prompt

Steps:

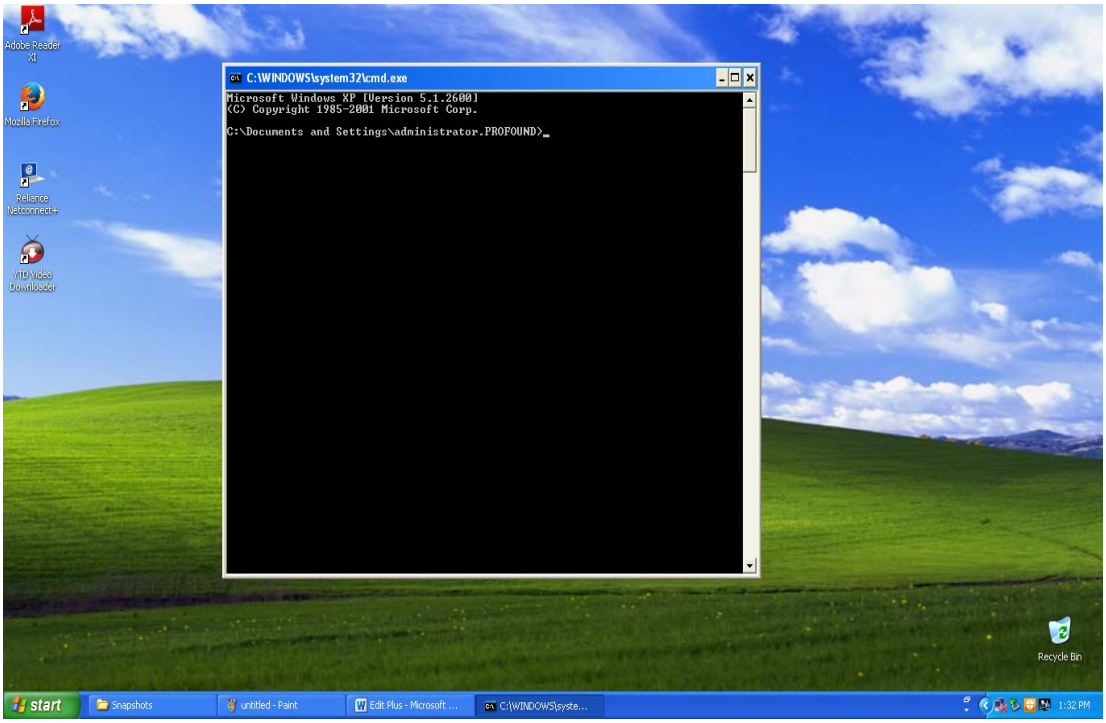
1. **Click on Start**



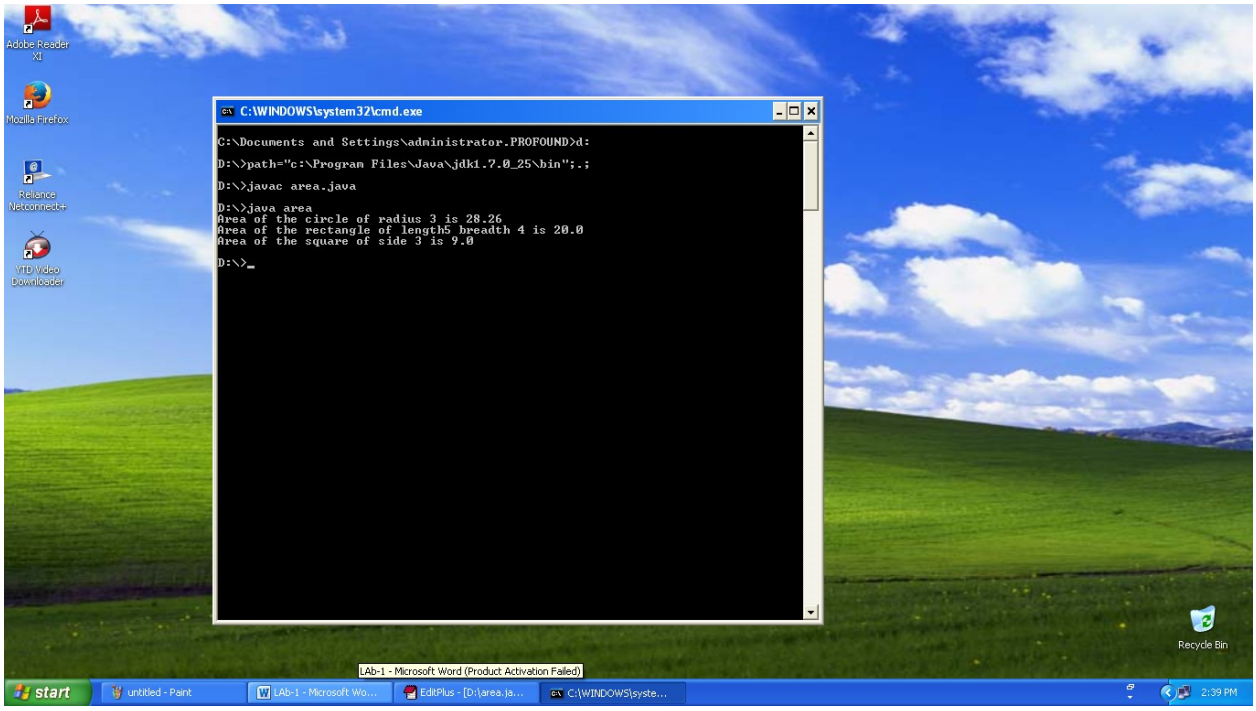
2. **Click on run**



3. **Type cmd and click on OK Button**



4. **From the command prompt move to the drive and folder where you have saved your program**
   like
   c:\Documents and Settings\administrator.PROFOUND> d:(press Enter Key)
   d:\>



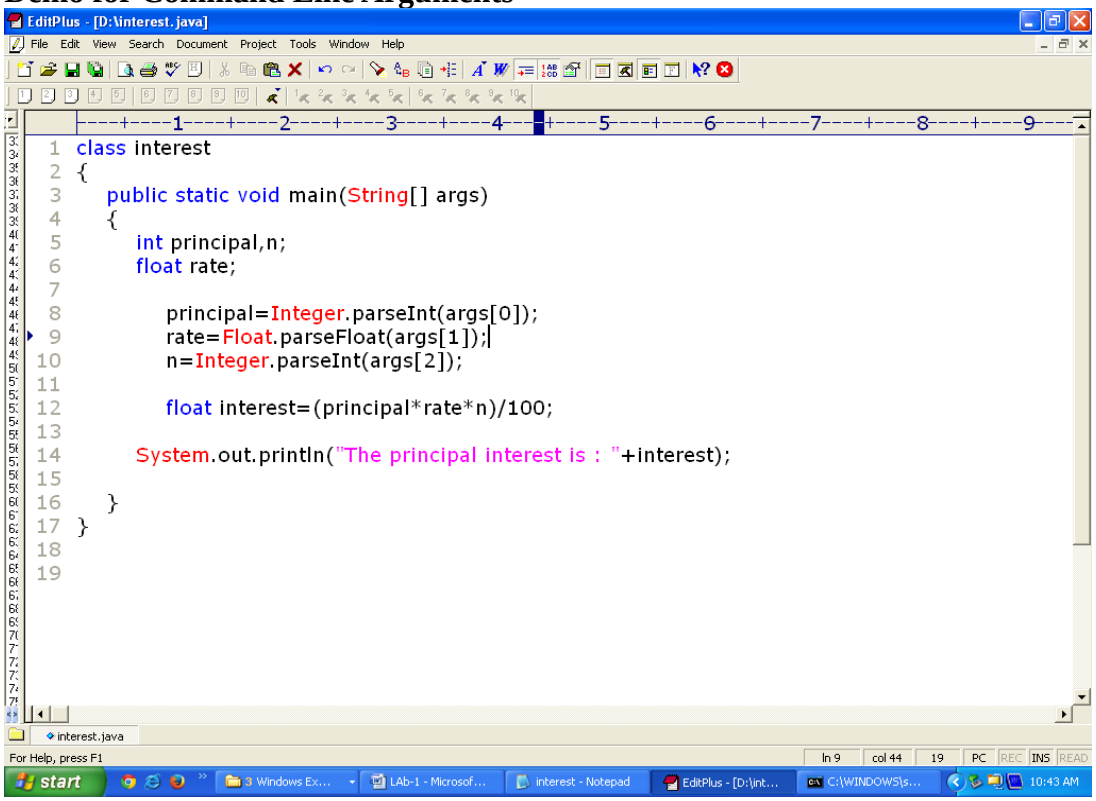
5: set the path

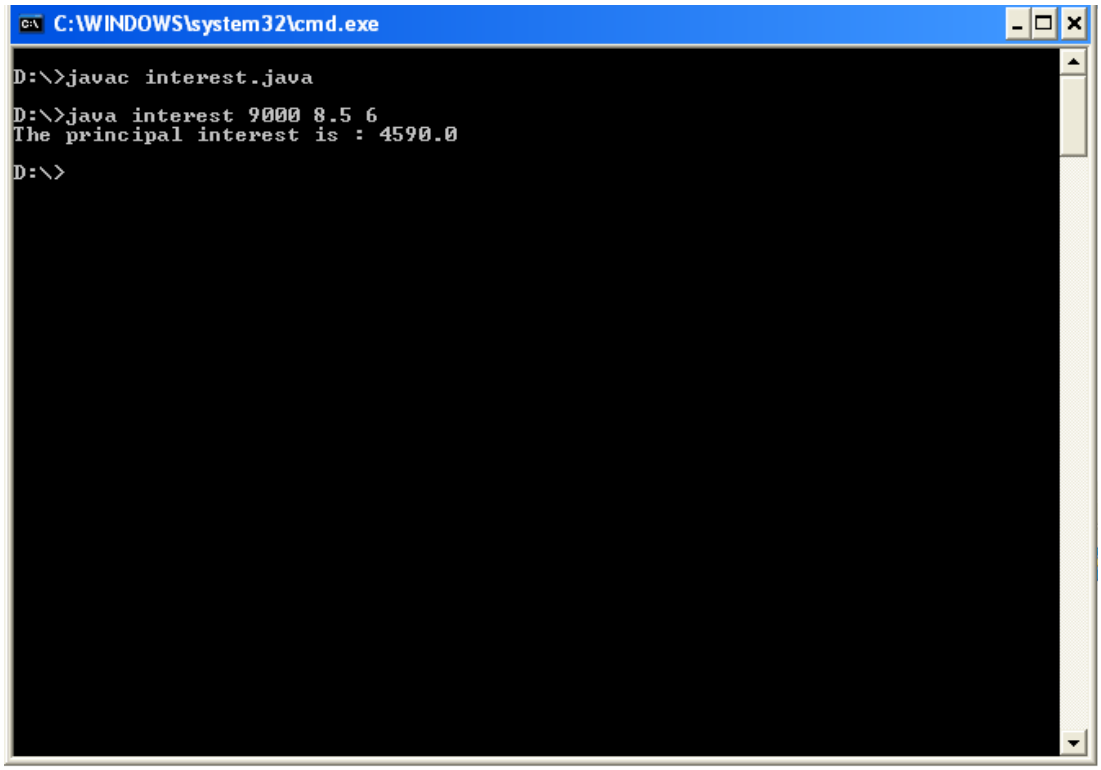
D:\> path=”c:\Program Files\Java\jdk1.7\bin”;.; //(Press enter key)

6. **Compile the source code**
   D:\>javac area.java
   Once compiled generates area.class file

7. **Run the .class file**
   D:\>java area
   Area of the circle of radius 3 is 28.26
   Area of the rectangle of length5 breadth 4 is 20.0
   Area of the square of side 3 is 9.0

Demo for Command Line Arguments





Assignments To Solve

*CLA-----> command Line Argument

1. accept two integer values via CLA* and perform all arithmetic operation
2. Modify the above program ,try it with different data types.
3. Accept basic salary via CLA* and calculate the following

DA=20% of basic
HRA=30% of basic
Calculate gross salary GROSS=BASIC+DA+HRA
Calculate the Income tax (IT) based on the following condition

SAL RANGE	IT
Basic >0 and basic < 4000	4%
Basic >=4000 and basic < 10000	6%
Basic >=10000 and basic < 17000	8%
Basic >=17000 and basic < 27000	10%
Basic >=27000 onwards	15%

5. Accept five different values via CLA* by using for loop and display sum of that values
6. Accept three digit number via CLA* and calculate the sum of digit
7. Accept values via CLA* for 1-D array of integer type and display it on screen
8. modify the above program and calculate the sum of number in array

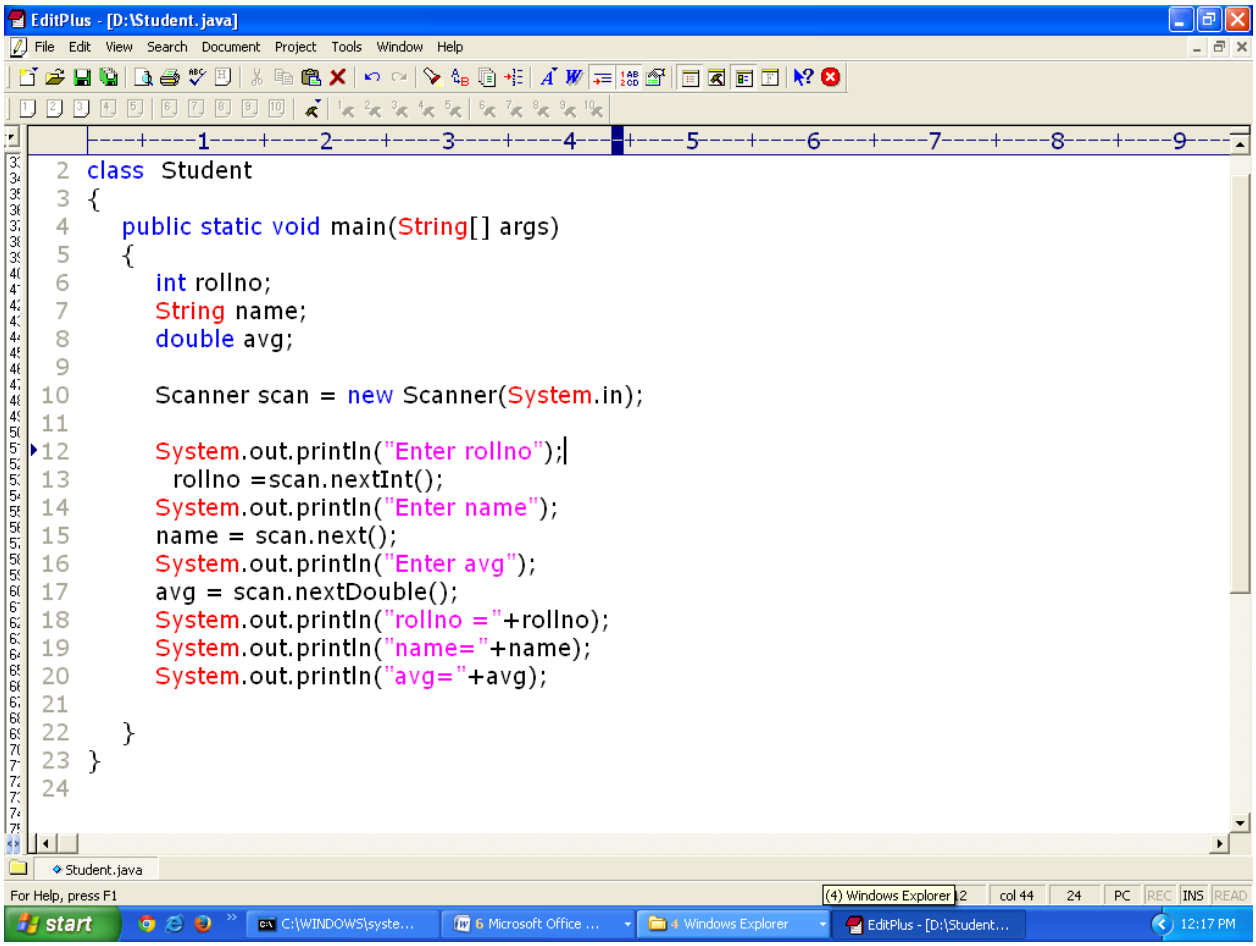
Lab-2 Assignments

1.Classess and Objects

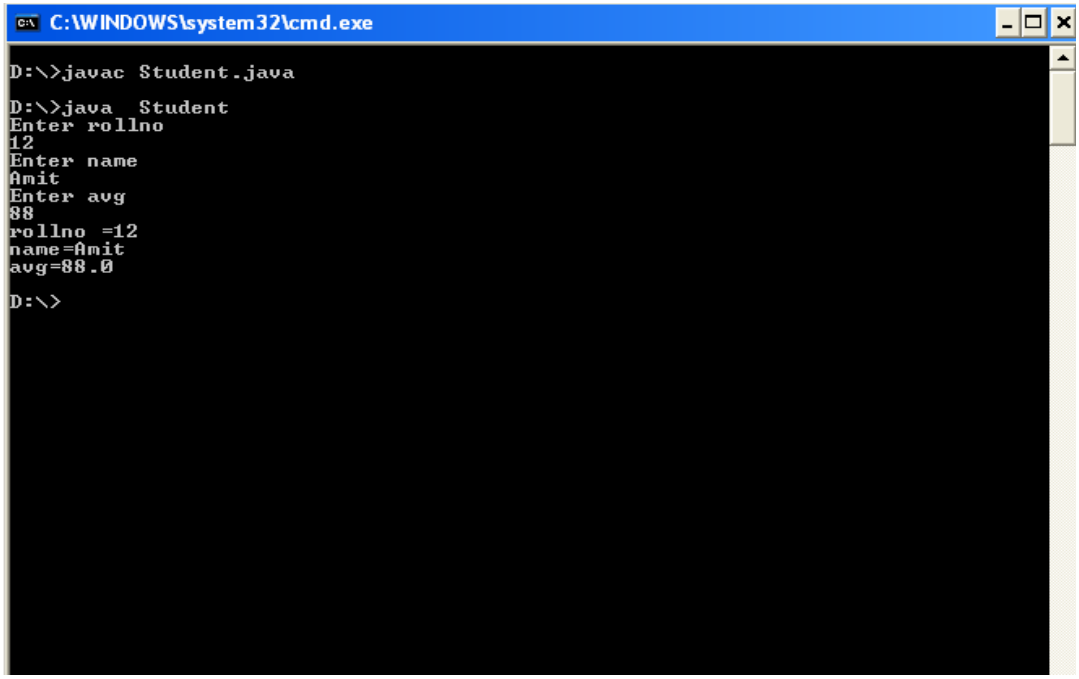
2.Using Scanner class

Introducing Scanner Class To Accept The Values From The User

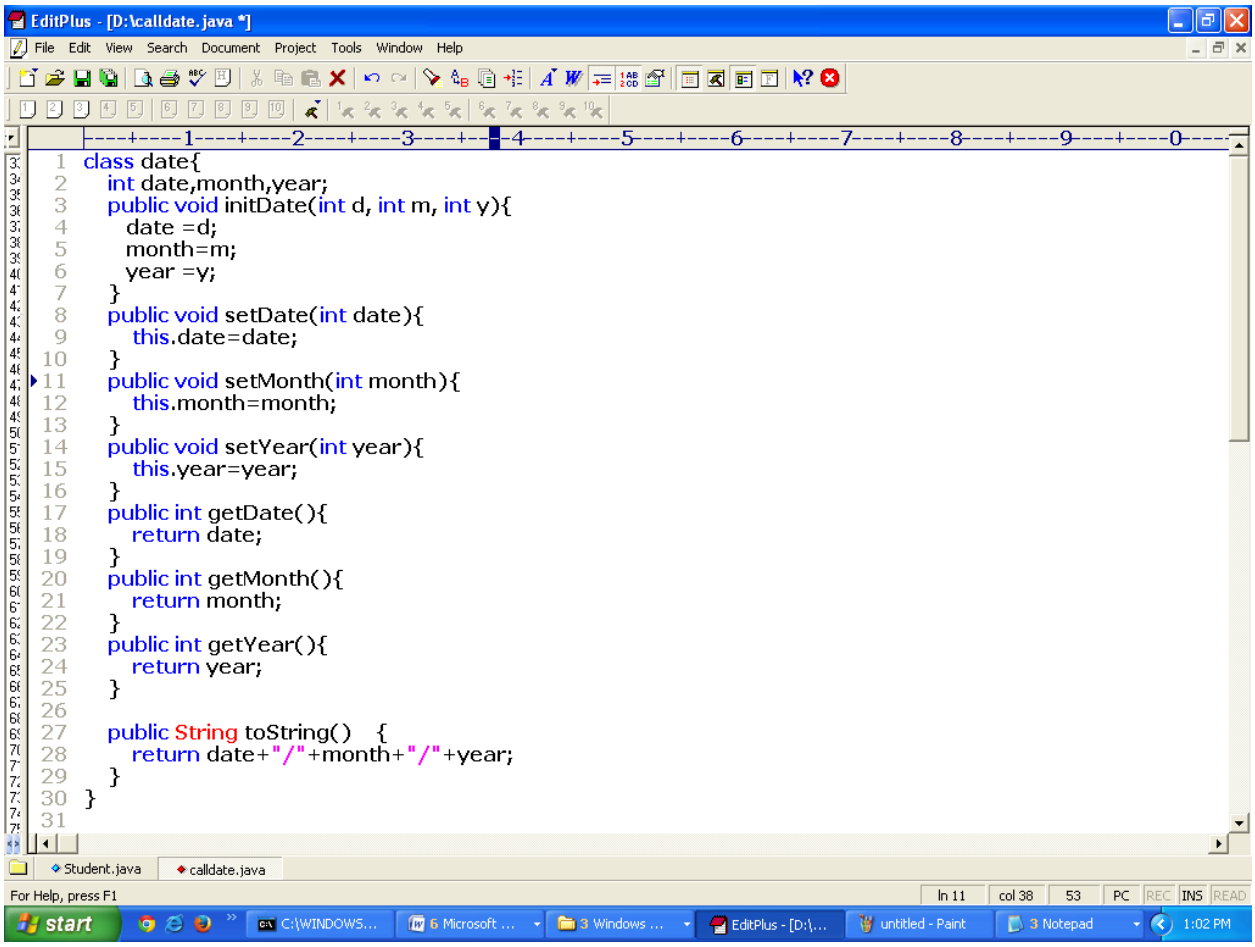
Demo for Scanner Class

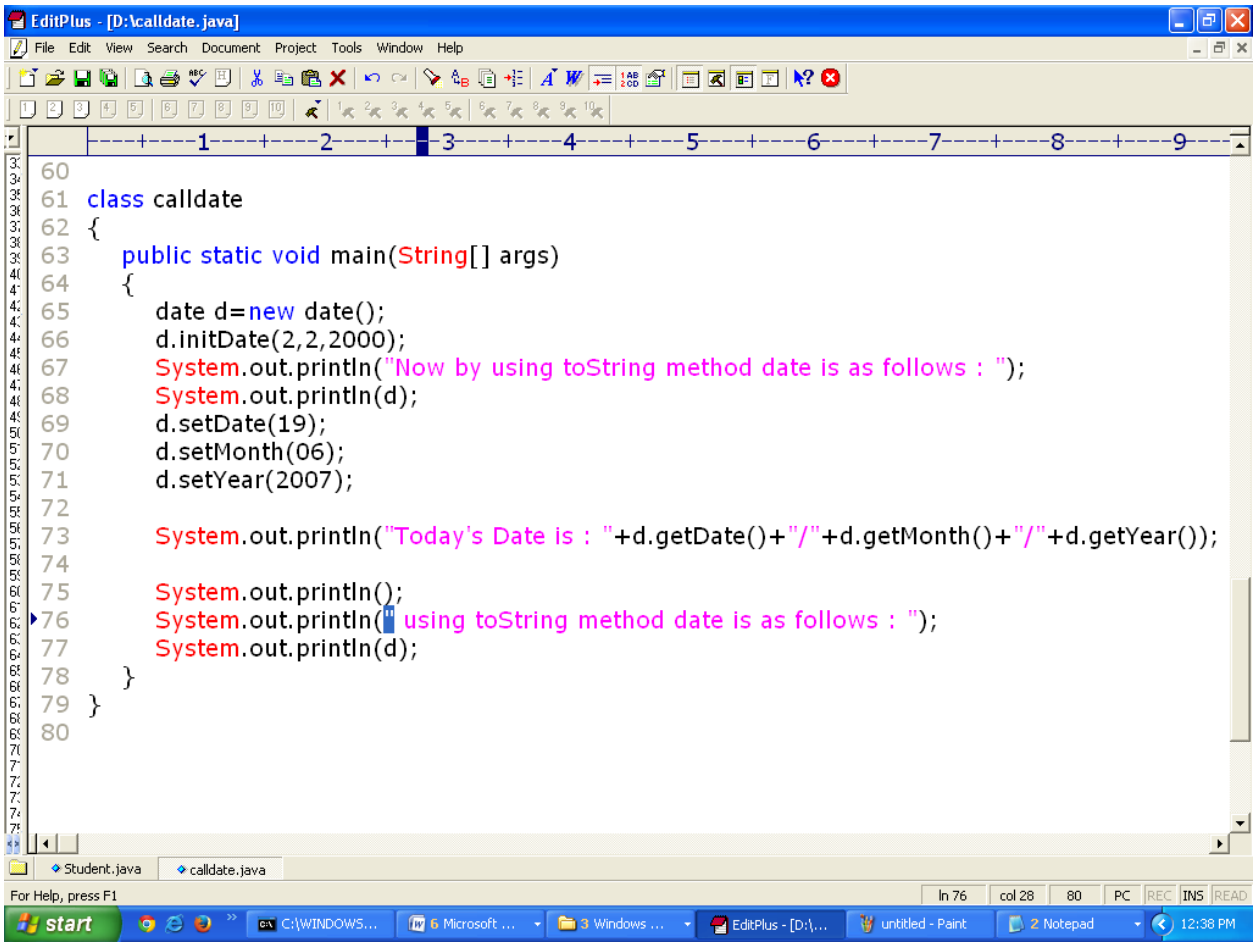


Save as Student.java(in d:\)
Compile and run from Command Prompt
Note: Always set the path if new command prompt is opened(as mentioned in the previous lab demo)



Demo for Getter Setter Methods

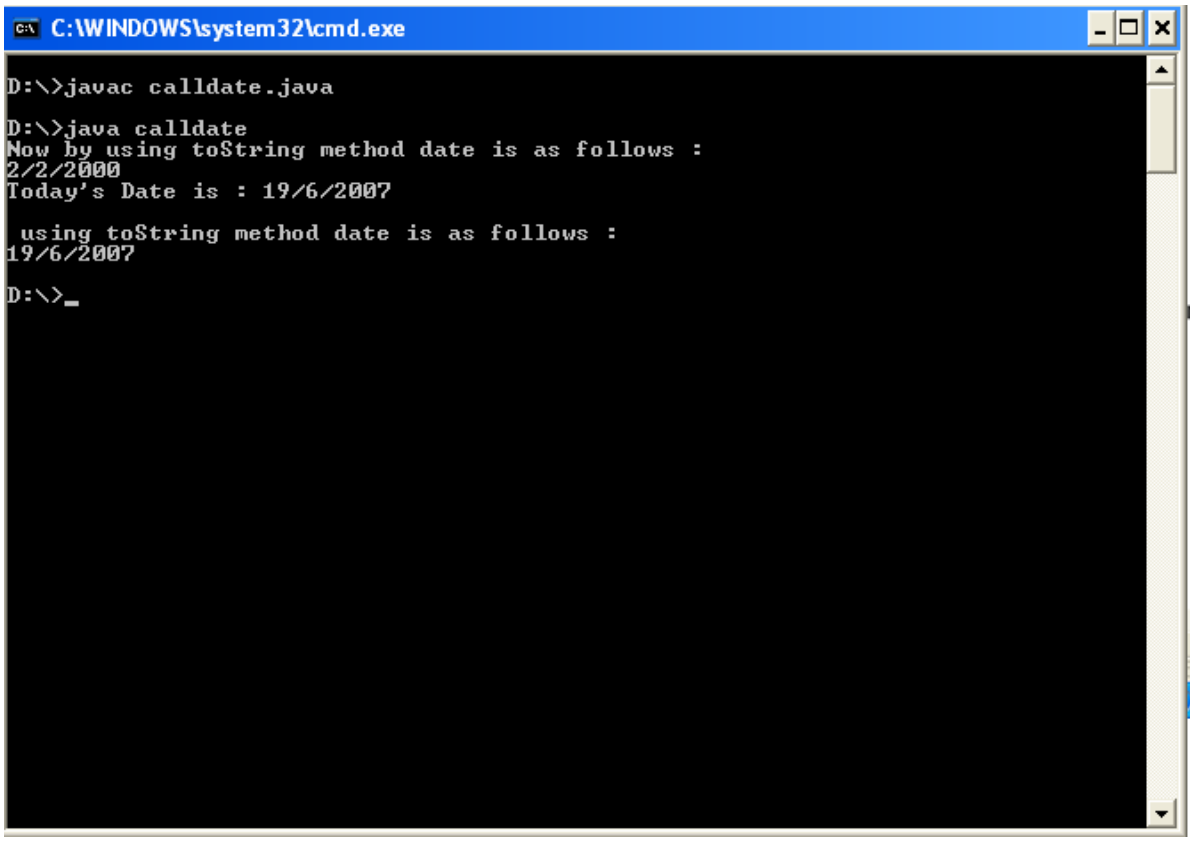




Save as calldate.java

Compile and run calldate.java

Note: When calldate.java is compiled , date class also gets compiled and date.class and calldate.class files get generated



Assignments To Solve :

1. WAP to check whether a person is eligible for voting
2. WAP to check whether a given year is a LEAP year
3. WAP to find the factorial of a number?
4. WAP to find whether a given number is an Armstrong number
5. WAP to reverse the given number
6. Generate the Fib… series 1 1 2 3 5 8 13
7. Create a class Book which describes its Book_title and Book_price. Use getter and setter Methods to get & set the Books description. Implement createBook and showBook methods to create and display Book Information. Write a separate class BookInfo to access the Book class

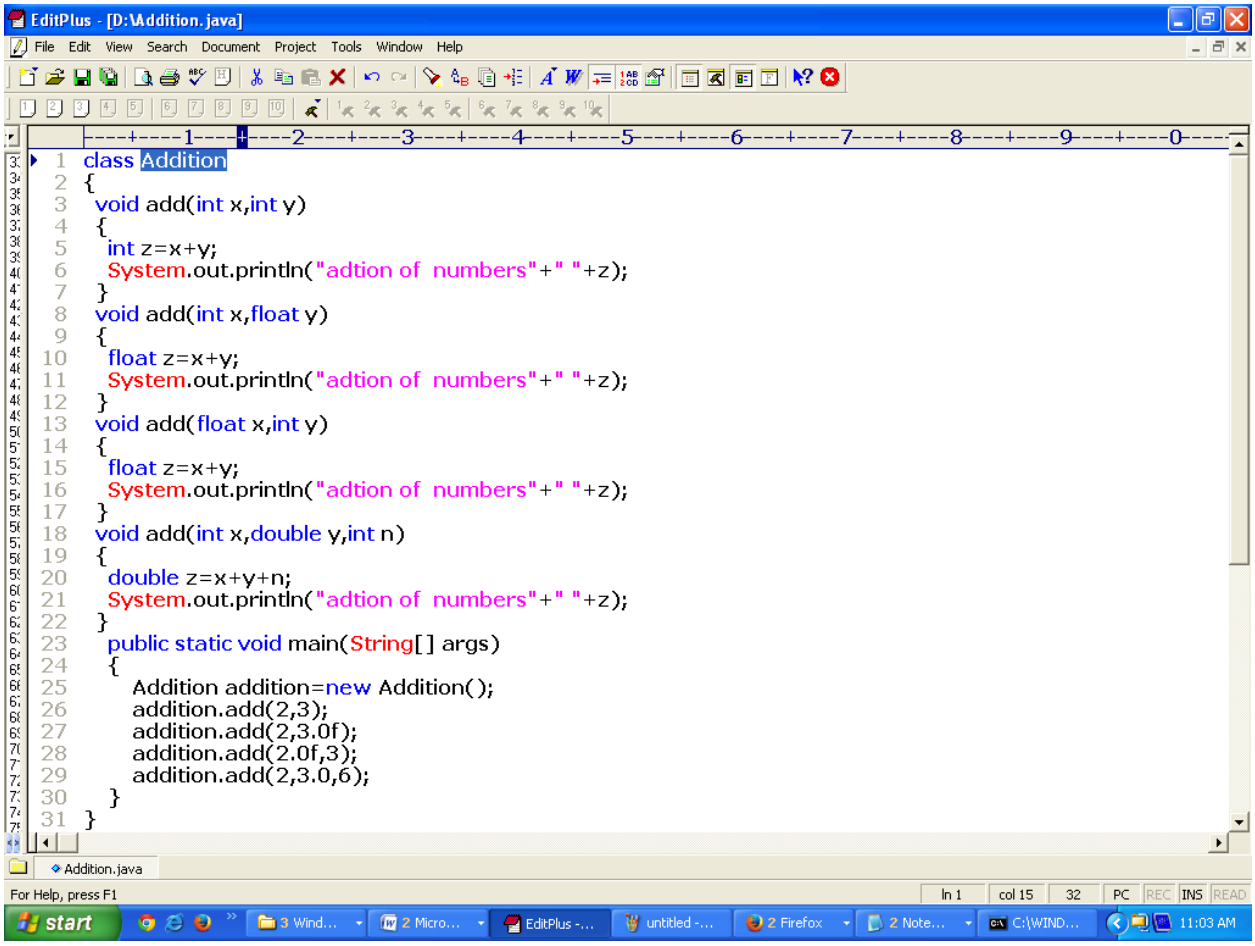
Lab-3 Assignments

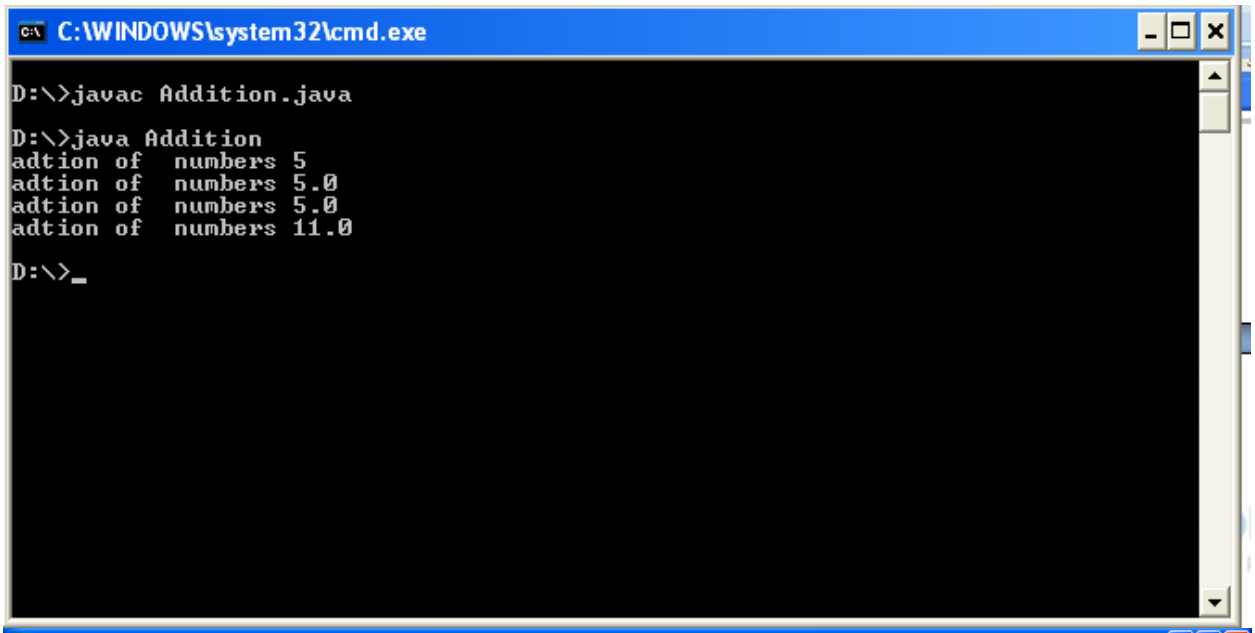
1.Introducing Constructor

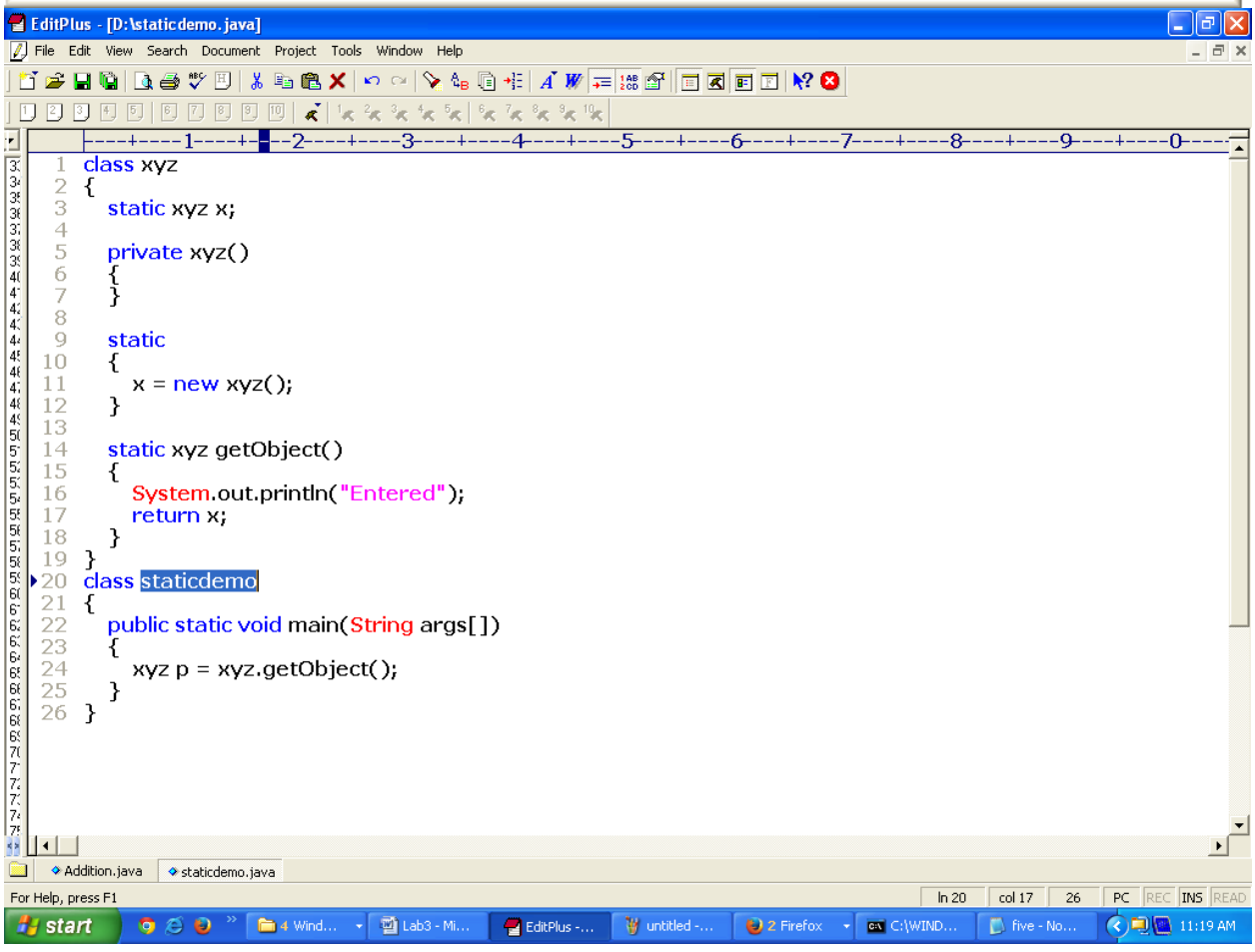
2.OverLoading Constructors

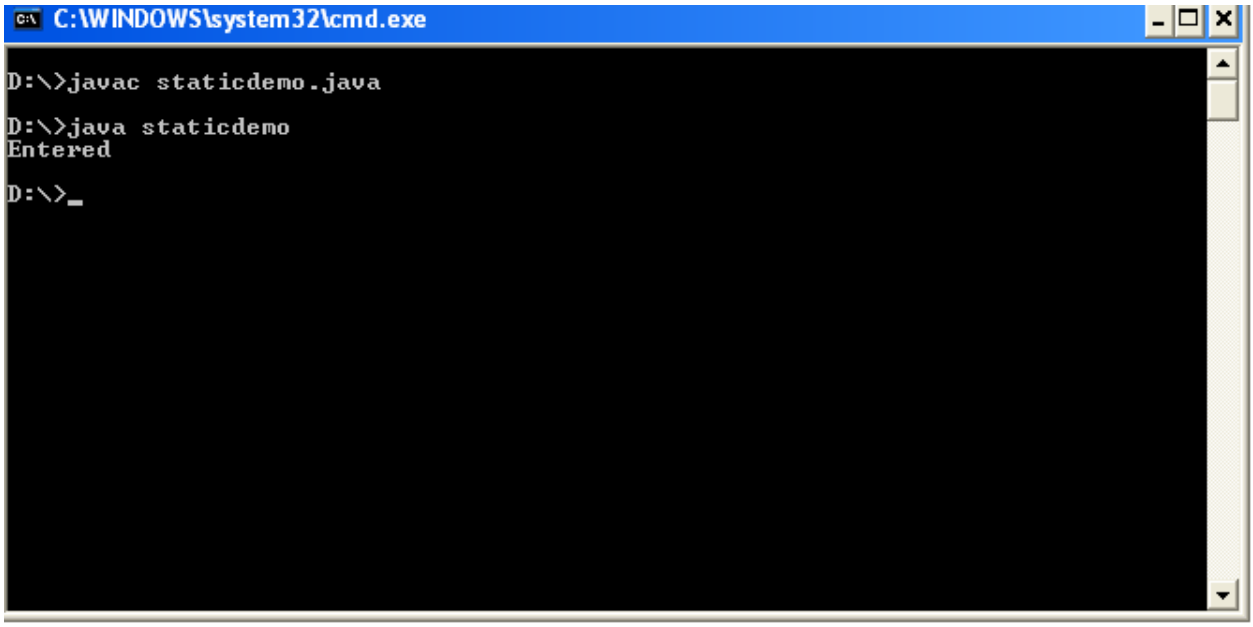
3.Overloading Methods/varargs/foreach

4.singletone object




C:\WINDOWS\system32\cmd.exe
D:\>javac Addition.java
D:\>java Addition
adtion of numbers 5
adtion of numbers 5.0
adtion of numbers 5.0
adtion of numbers 11.0
D:\>


EditPlus - [D:\staticdemo.java]
File Edit View Search Document Project Tools Window Help
class xyz
{
static xyz x;
private xyz()
{
}
static
{
x = new xyz();
}
static xyz getObject()
{
System.out.println("Entered");
return x;
}
}
class staticdemo
{
public static void main(String args[])
{
xyz p = xyz.getObject();
}
}
Addition.java
staticdemo.java
For Help, press F1
start



Assignments To Solve :

1. Write an Account class with default constructor, parameterised constructor and methods toString(), deposit(int amt), withdraw(int amt). withdraw method should take care of insufficient balance. Accept the values from the user..Display the details of various Accounts

2. Write a class Addition2 with add method (overloading) using varargs and enhanced for loop

3. Declare three array of integer types ,Accept values for two array.Perform arithmetic operation on both array values and store corresponding values in third array. e.g: c[0]=a[0]+b[0]

4. Write a class stack. Use methods push and pop to store and retrieve elements from stack(hint: create an array of int to store the elements)

5. Write a singleton class(Singleton class is is a class which has only one object)

6. Consider you are working for Xyz Company and developing a payroll software

   Construct a class employee with following member using private access specifier:

   1. employeeId integer
   2. employeName string
   3. basicSalary double

4. hra double
5. medical double
6. pf double
7. pt double
8. netSalary double
9. grossSalary double

Please use following expressions for calculations.

a. hra=50% of basic salary
b. pf=12% of basic salary
c. pt=Rs.200

1.Write no arguments constructor and parameterized constructor to initialize objects.

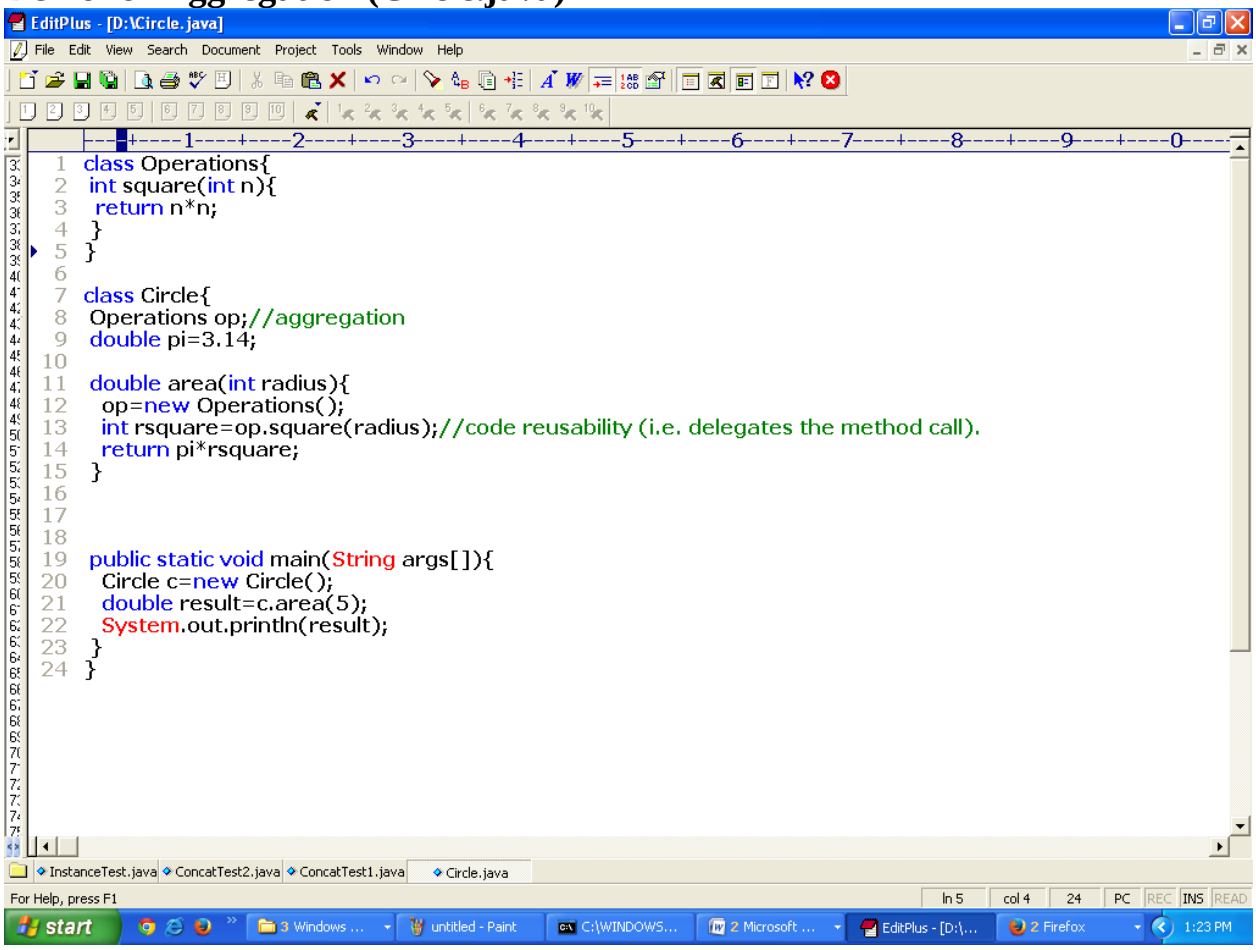
2.Write properties for employeId, employeeName, basic salary, netSalary and grossSalary

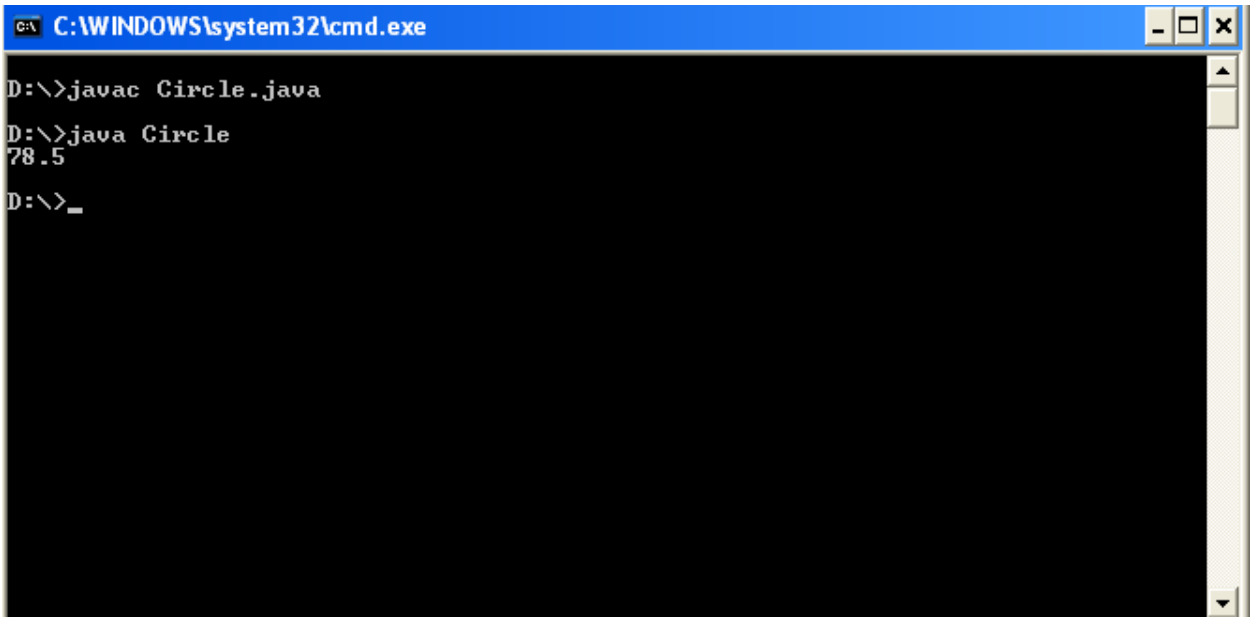
3.Write methods to display the details of an employee and calculate the gross and the net salary. gross salary=basic salary+hra+medical
net salary=gross salary-(pt+pf)

Lab-4 Assignments

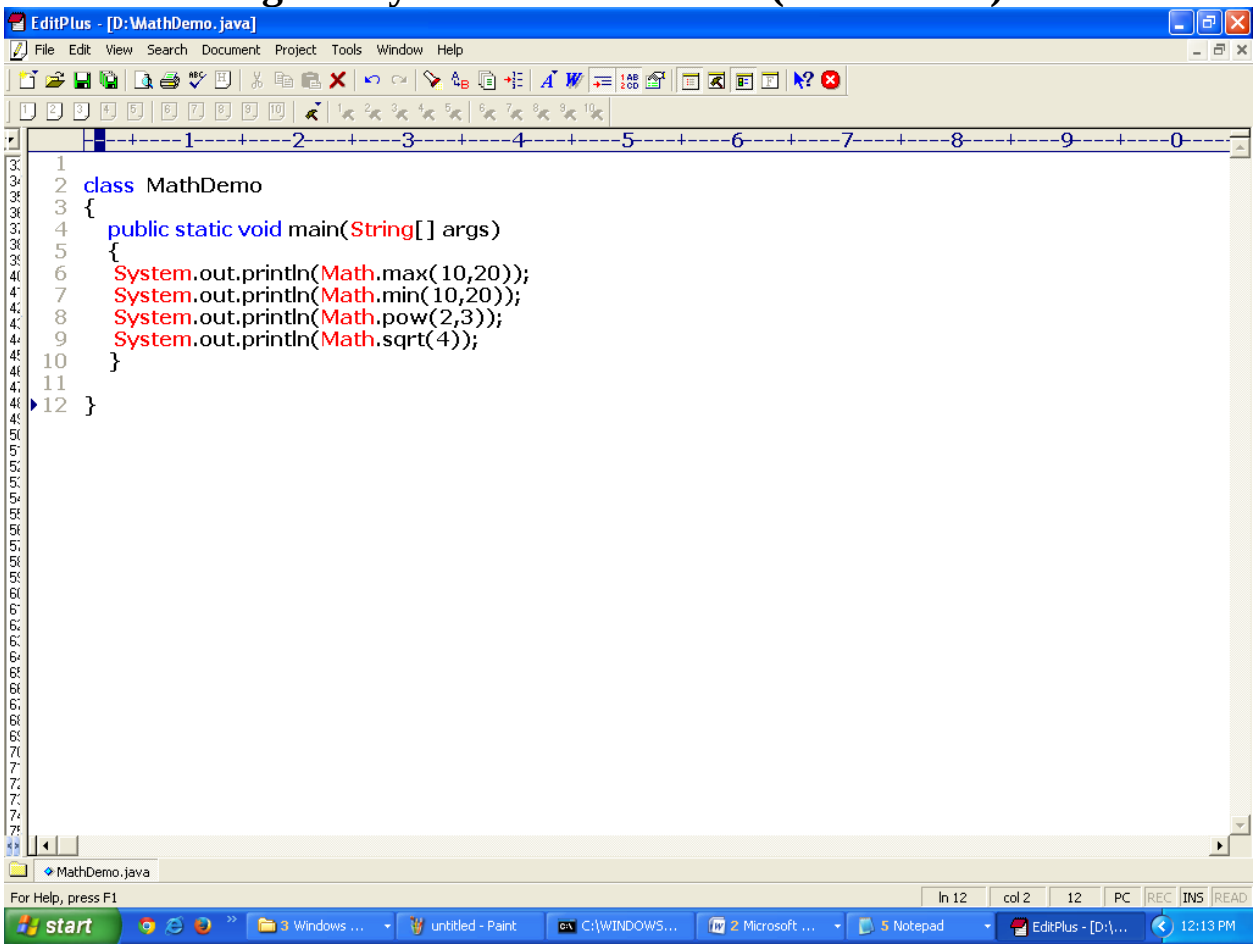
1. Aggregation/Containment
2. String ,StringBuffer,StringBuilder
3. Math and Wrapper classes
4. Enum

Demo for Aggregation (Circle.java)

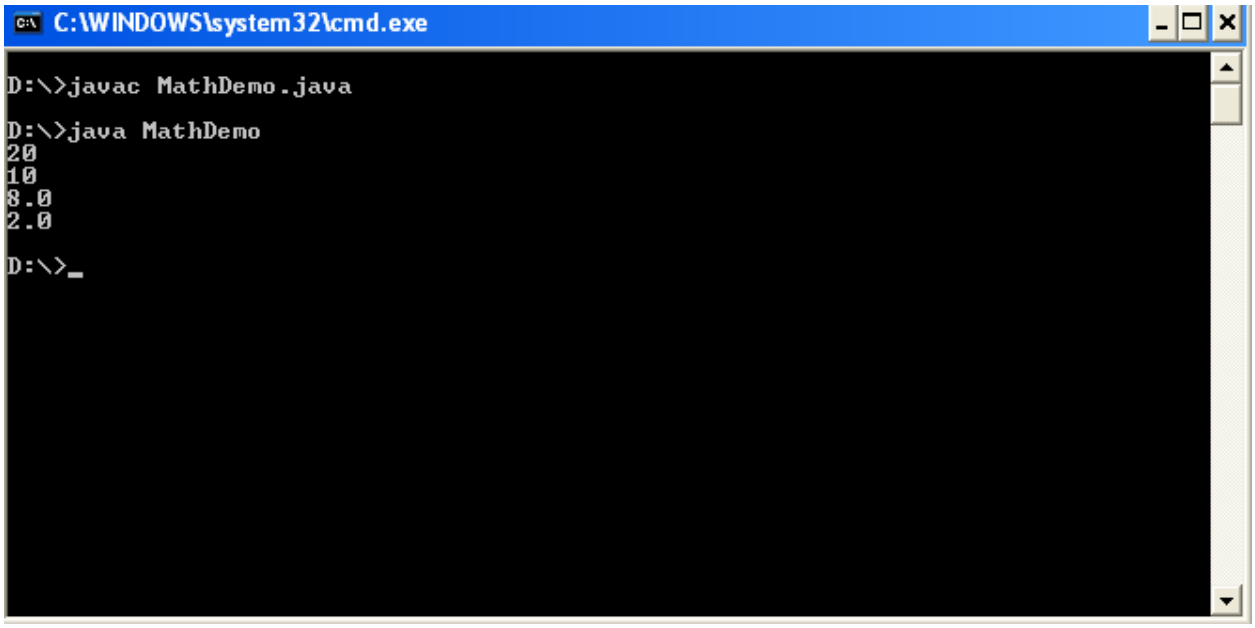




Demo for using Readymade class methods(Math class)



Note: Use some more methods of Math class to do math operations

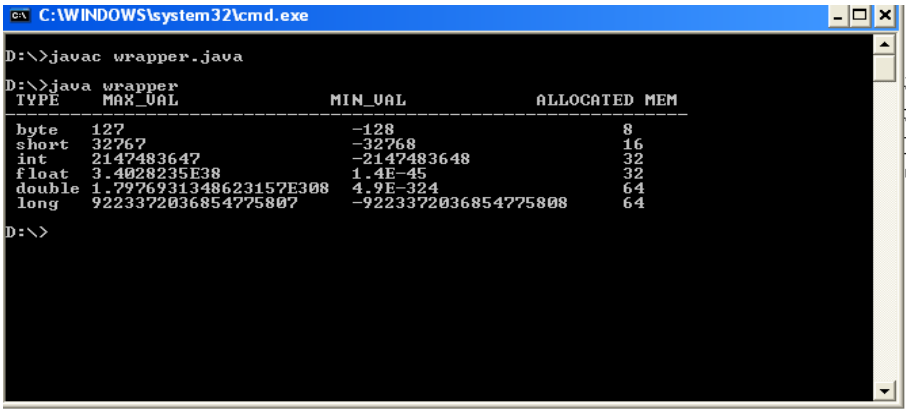


Assignments To Solve :

1. Write an Employee class with id, name and dob(Contained Object.. Containment)with Default and parameterised constructor & toString()..
2. Write program to create array of employee object, and print all employee information

3. Use the String Class Methods like length, hashCode,equals, replace, trim, subString, concatinate , compare , charAt, subString etc.. for a given String(s)
4. practice StringBuffer methods such as length(), capacity(),append(),insert(),delete() etc
5. Write a java program to split string by “blank space” and “-” patterns
6. Create string by using below ways and compare using “==” and equals() method:
    a. String s1=”abc”;//literal way
    b. String s2=”abc”;
    c. String s3=new String(“abc”); //using new keyword
    d. String s4=new String(“abc”);
7. Write a Java program to return the sum of the digits present in the given string. If there is no digits the sum return is 0.
    a. String s=”14abc9kj”;
    b. O/p: 1+4+9=14

8. Use Wrapper class methods to print MaxValue, MinValue and Size for various primitive types



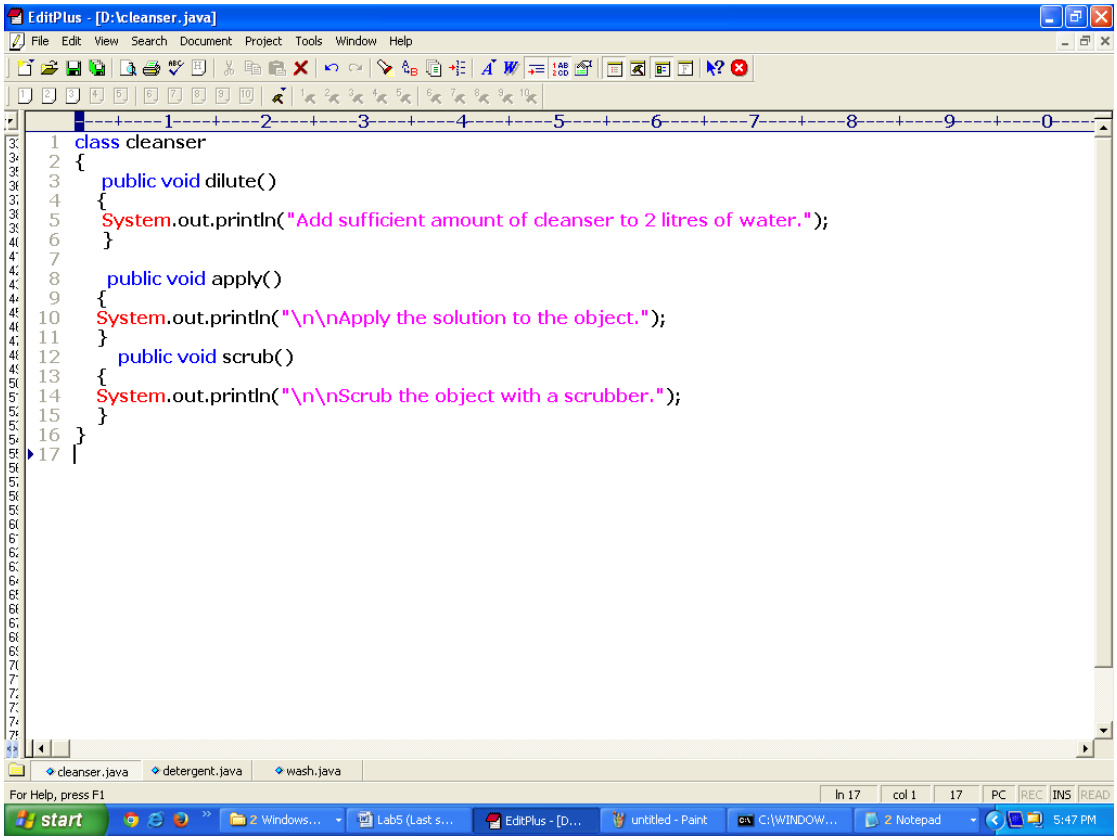
9. Accept Integer values from user and display it's equivalent Binary, Hexadecimal, Octal values.

10. Declare enum of weekday: Print number with day
    Hint: public enum day{SUNDAY (0), MONDAY (1)}
    Print: 0 SUNDAY 1 MONDAY

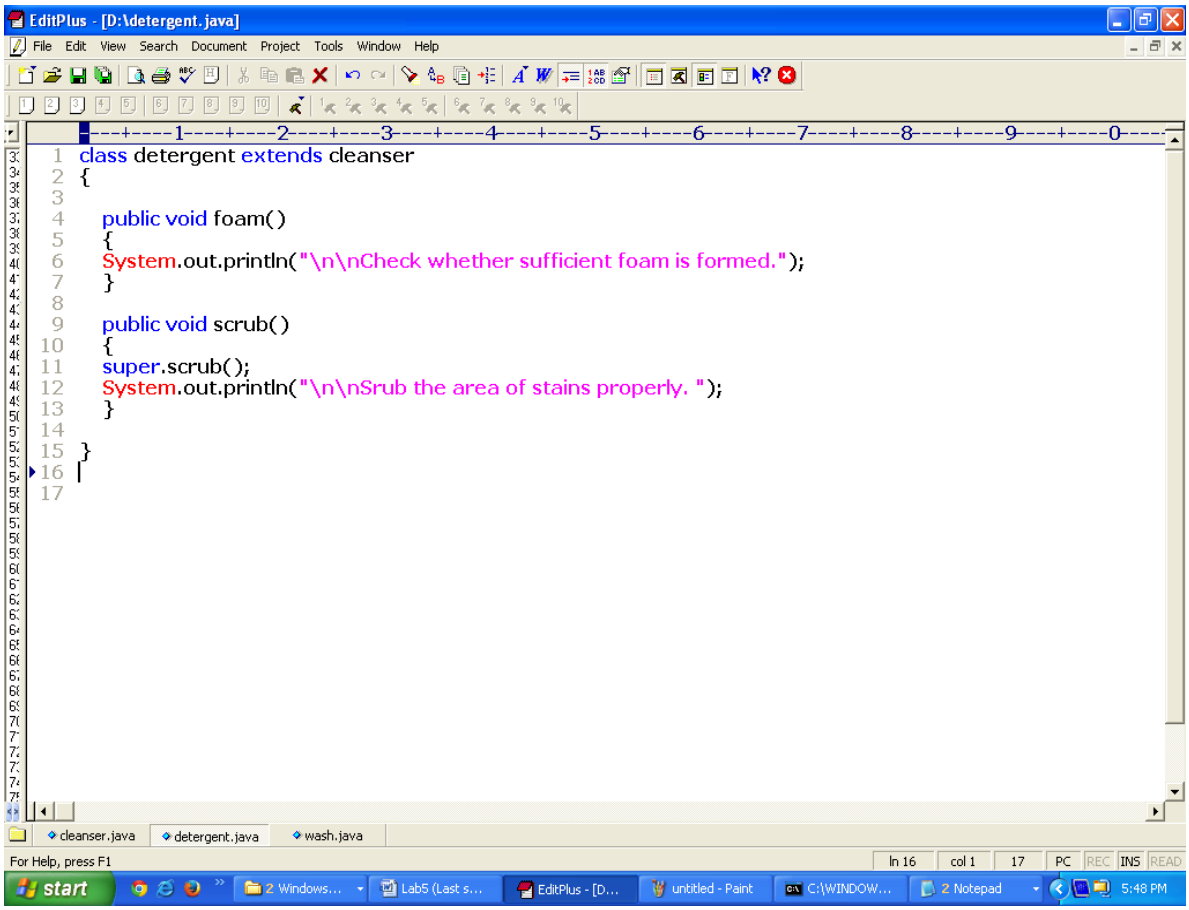
Lab-5 Assignments

Introducing Inheritance (is a a type of relationship)

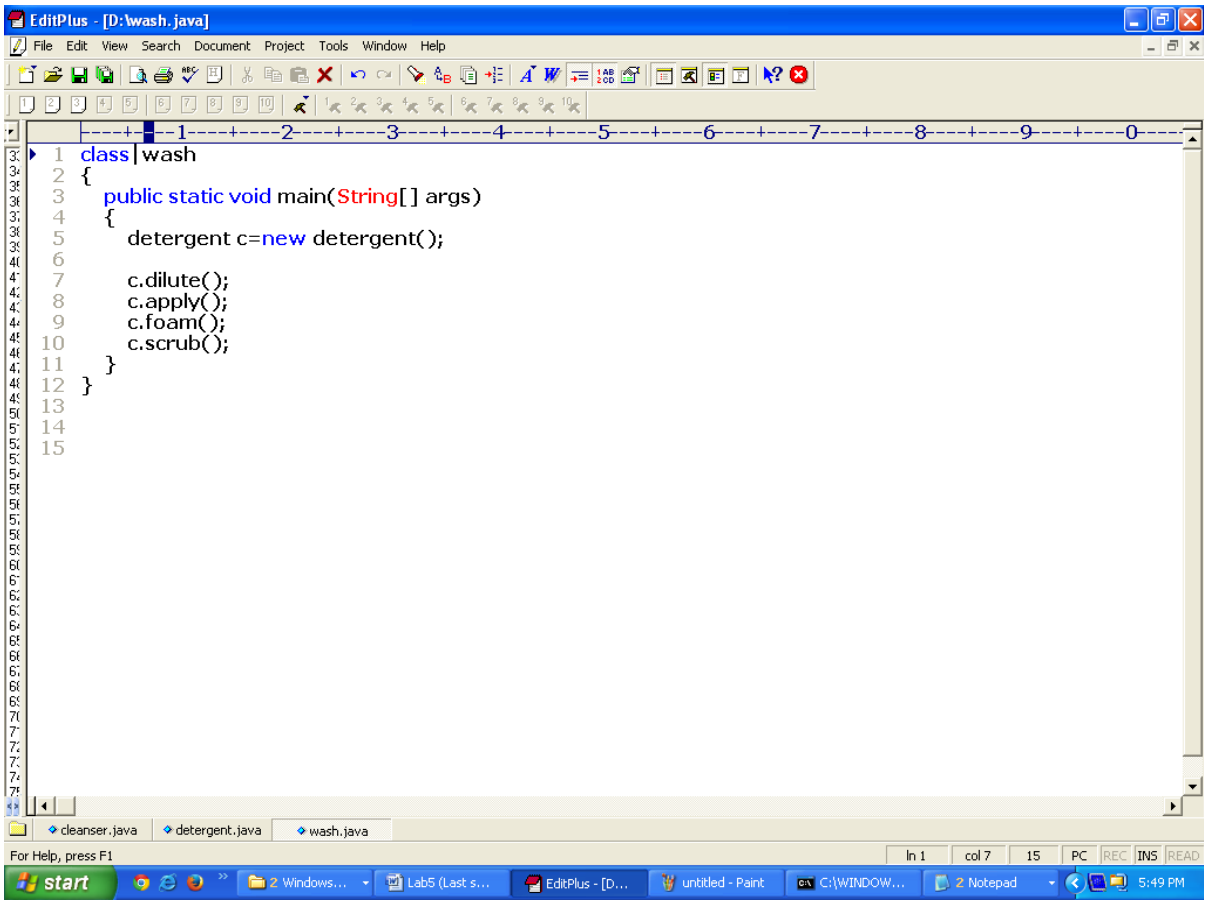
Demo for Inheritance



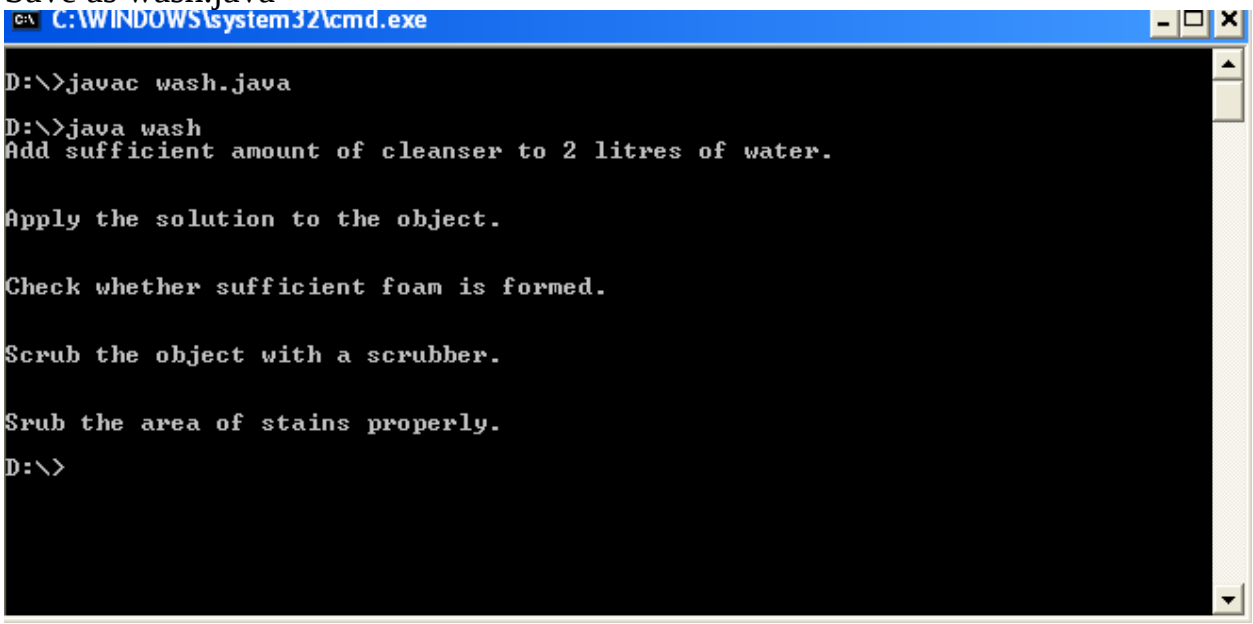
Save as cleanser.java



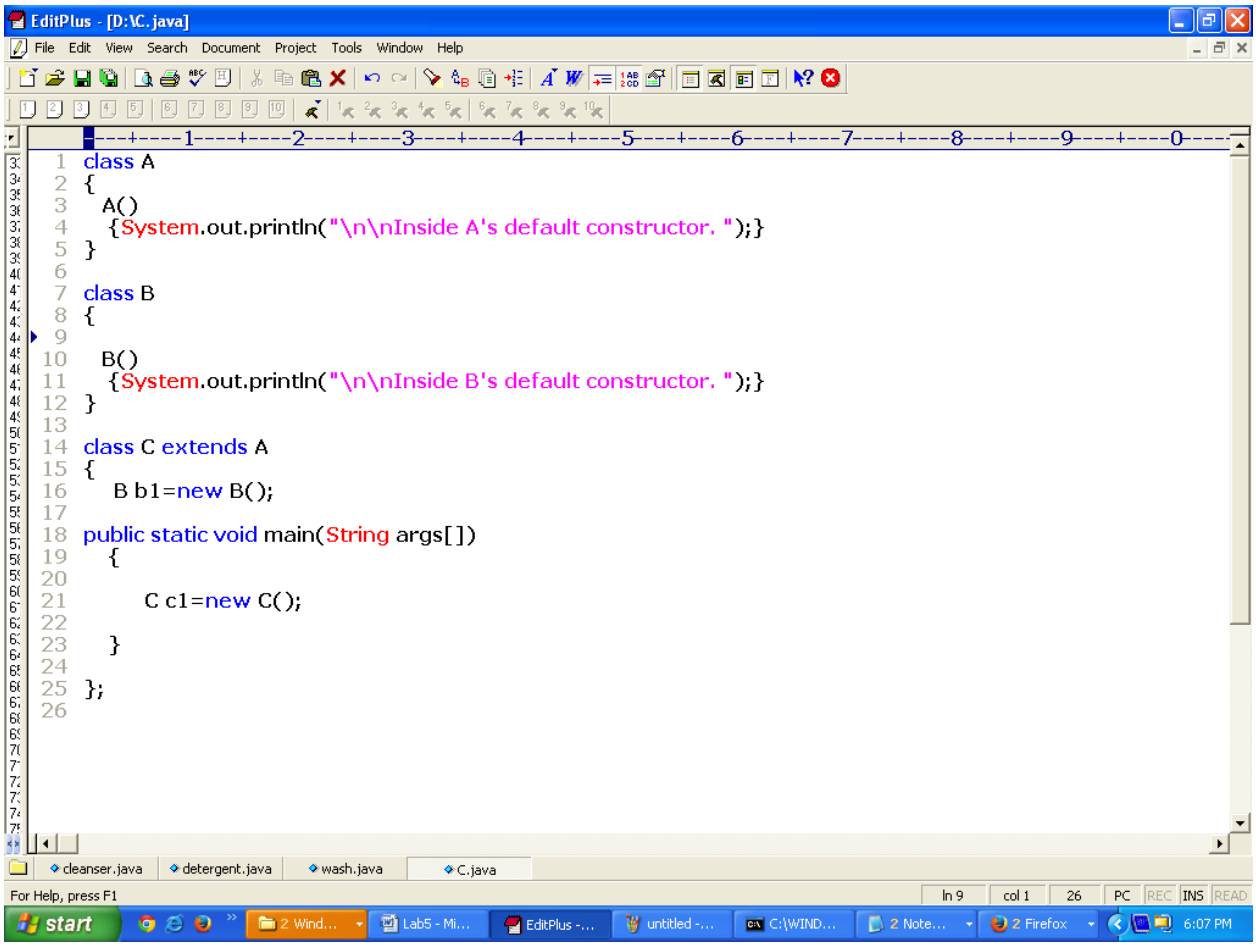
Save as detergent.java



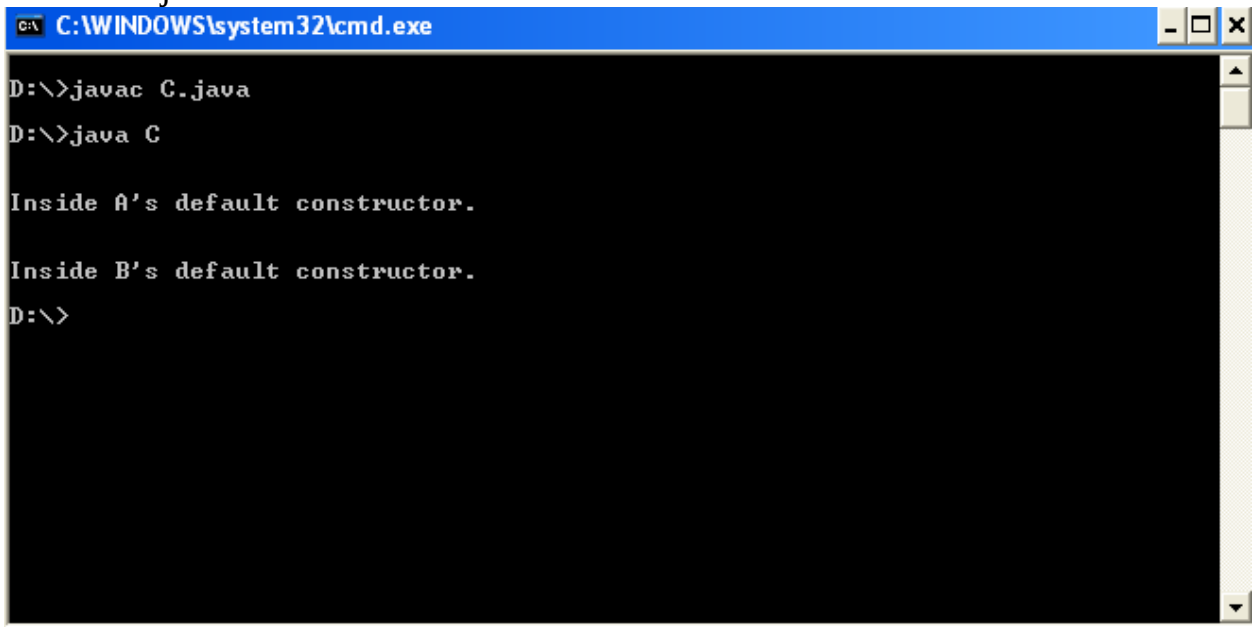
Save as wash.java



Demo2



Save as C.java



Assignments To Solve :

1. Write class game{ }

   class boardgame extends game{ }
   class chess extends boardgame{ }
   class gamedemo{
   public static void main(String[] args) {
   chess ch=new chess(); }
   }
   (Note: Write only default constructors in each class with specific information . See the order of constructors invoked when chess object is created)

2. Create class **WageEmployee** extending **Employee** class with attributes as hrs (int)and rate(int) and method computeSalary() to calculate the salary.Print the salary and details of WageEmployee.
   (Note: Use the previous date and Employee classes. Accept the values from the user..Default, Parameterised Constructor and toString() to be written in all the classes)

3. Create **SalesPerson** class extending **WageEmployee** with attributes as sales(int) and commission (int). Overide the ComputeSalary()
   in Salesperson class and print the salary and details of SalesPerson

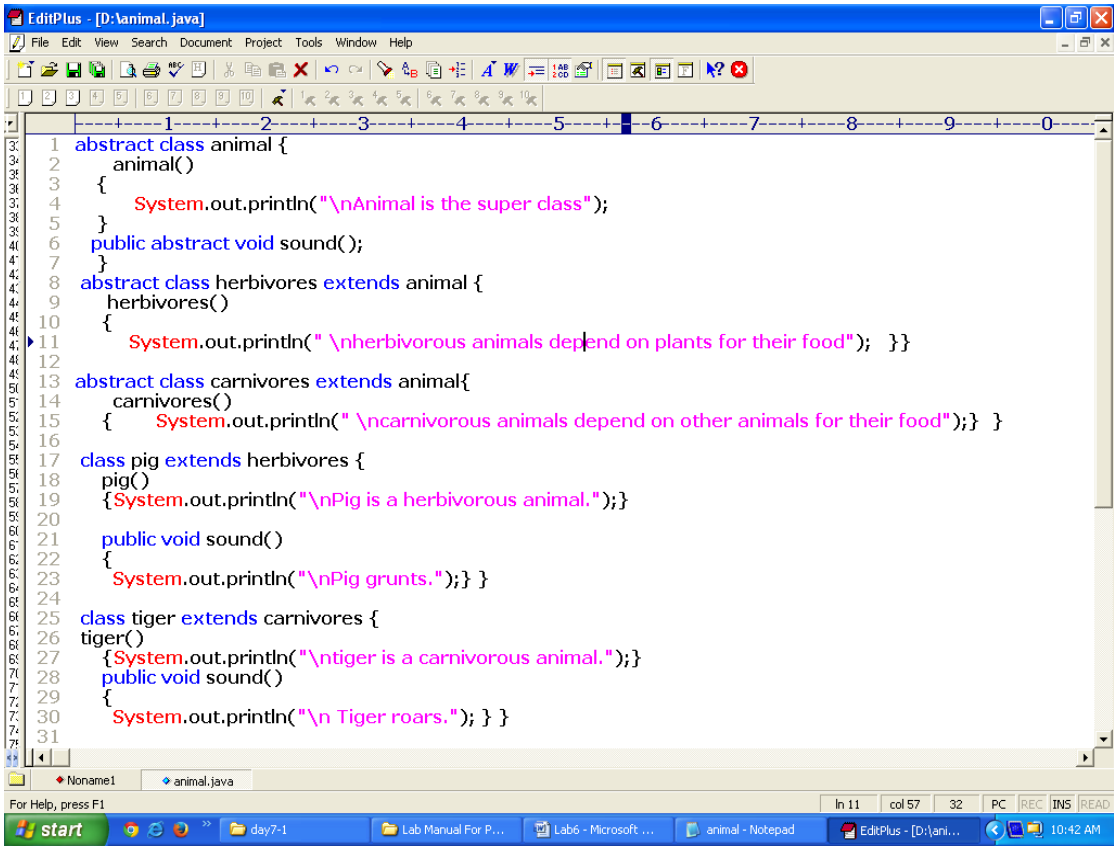
4. Create **Manager** class extending **Employee** class with attributes as fixedsalary(int) and incentives(int) and method computeSalary() to calculate the salary of Manager .Print the salary and details of Manager

5. Write a **TestEmployee** class to print the details of all types of employees (use array[] of Employee Objects)

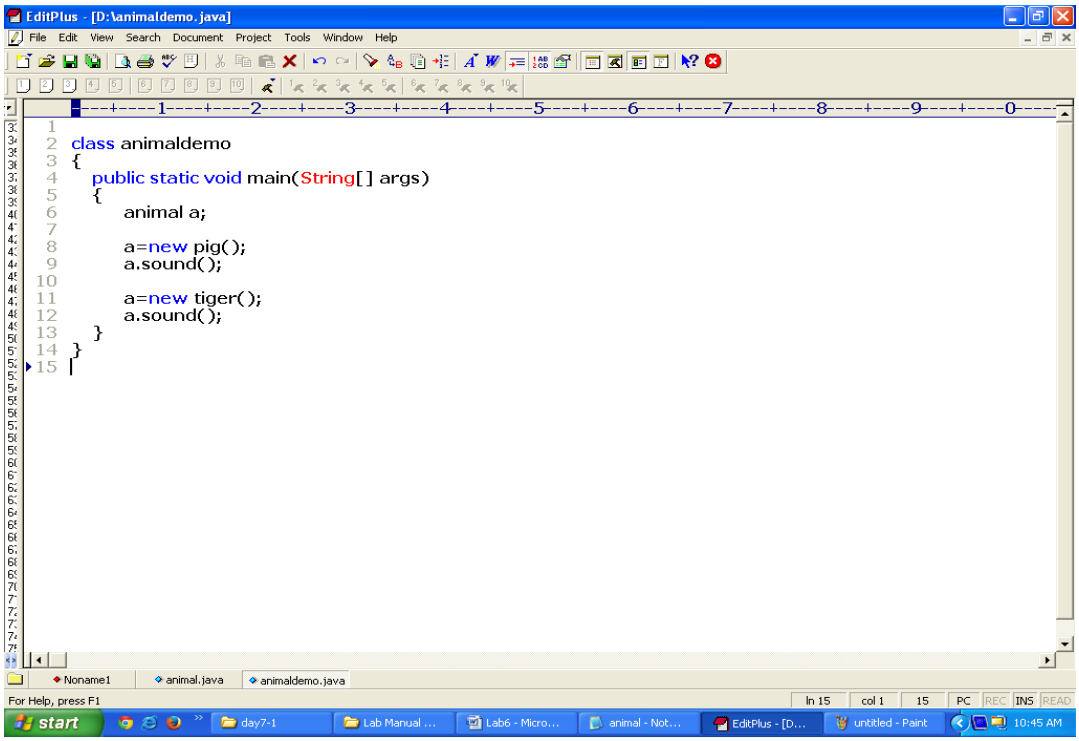
Lab-6 Assignments

Introducing Abstract class

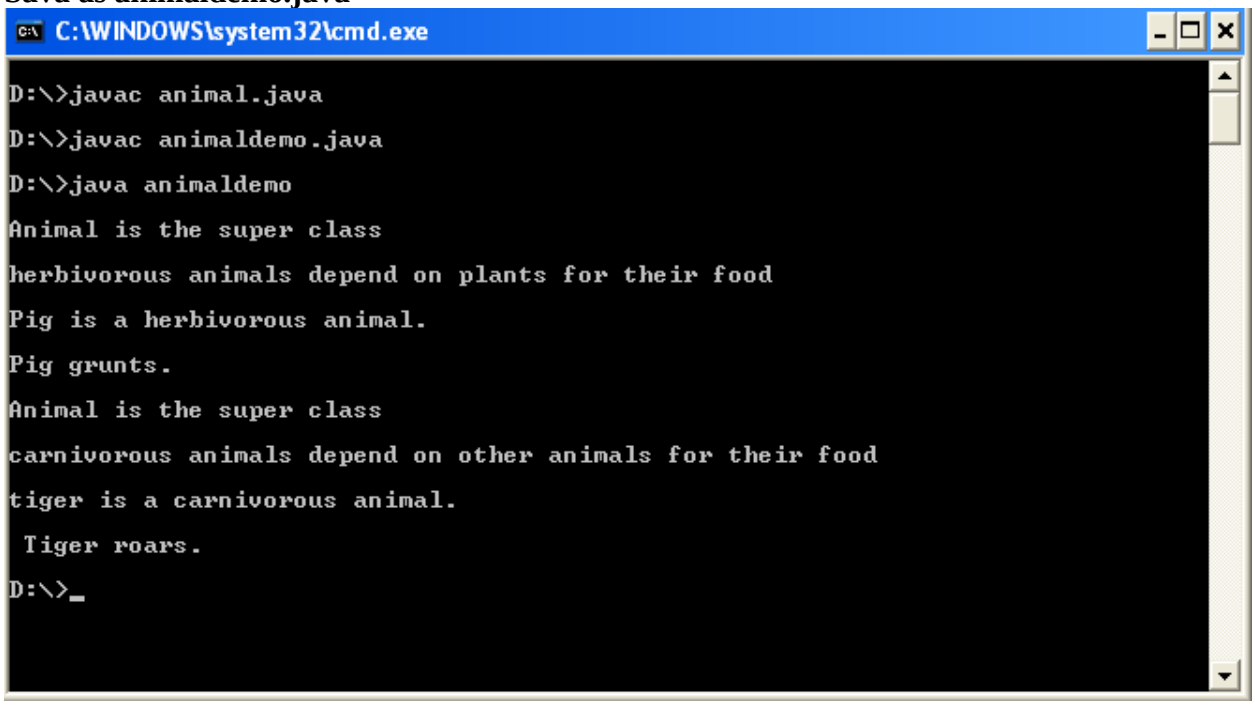
Demo for Abstract class



Save as animal.java



Sava as animaldemo.java

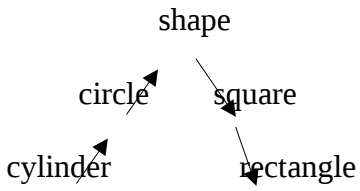


Assignments To Solve

1. Create an abstract class Instrument which is having the abstract function play. Create three more sub classes from Instrument which is Piano, Flute, Guitar. Override the play method inside all three classes printing a message .
   "Piano is playing tan tan tan tan " for Piano class
   "Flute is playing toot toot toot toot" for Flute class
   "Guitar is playing tin tin tin " for Guitar class
   You must not allow the user to declare an object of Instrument class.
   Create an array of 10 Instruments.
   Assign different type of instrument to Instrument reference.
   Check for the polymorphic behavior of play method.
   Use the instanceof operator to print that which object stored at which index of instrument array.

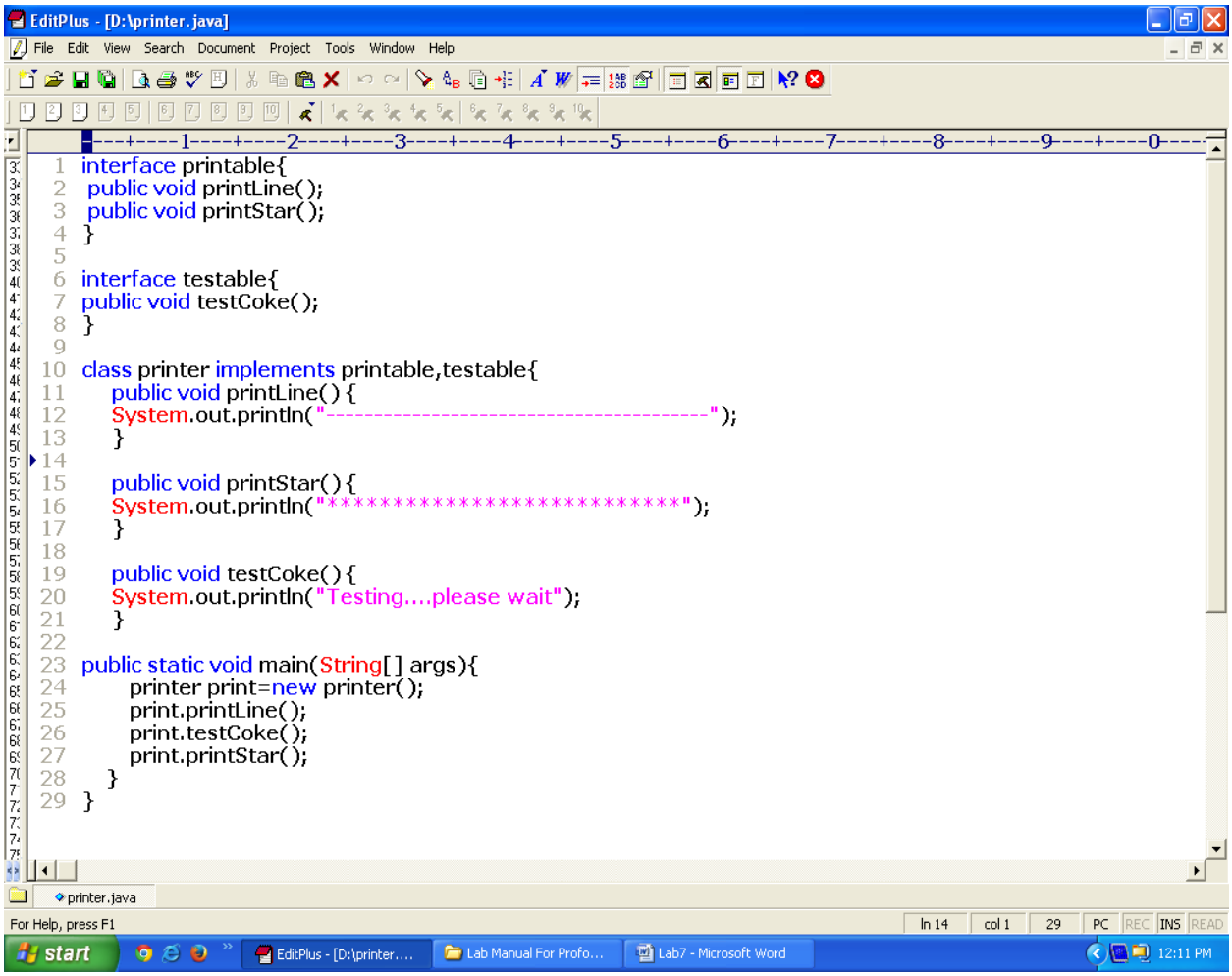
2. Create a class Medicine to represent a drug manufactured by a pharmaceutical company. Provide a function displayLabel() in this class to print Name and address of the company.
   Derive Tablet, Syrup and Ointment classes from the Medicine class. Override the displayLabel() function in each of these classes to print additional information suitable to the type of medicine. For example, in case of tablets, it could be "store in a cool dry place", in case of ointments it could be "for external use only" etc.
   Create a class TestMedicine. Write main function to do the following:
   Declare an array of Medicine references of size 10
   Create a medicine object of the type as decided by a randomly generated integer in the range 1 to 3.
   Refer Java API Documentation to find out random generation feature.
   Check the polymorphic behavior of the displayLabel() method.

3. Create an abstract class shape with abstract method
   void area();
   create 4 more classes circle, cylinder, square and rectangle



   Override the area() in all the classes
   Create an array of references of type shape in TestShape class and print the area of different types of shapes.

Lab-7 Assignments
Introducing Interfaces and Packages



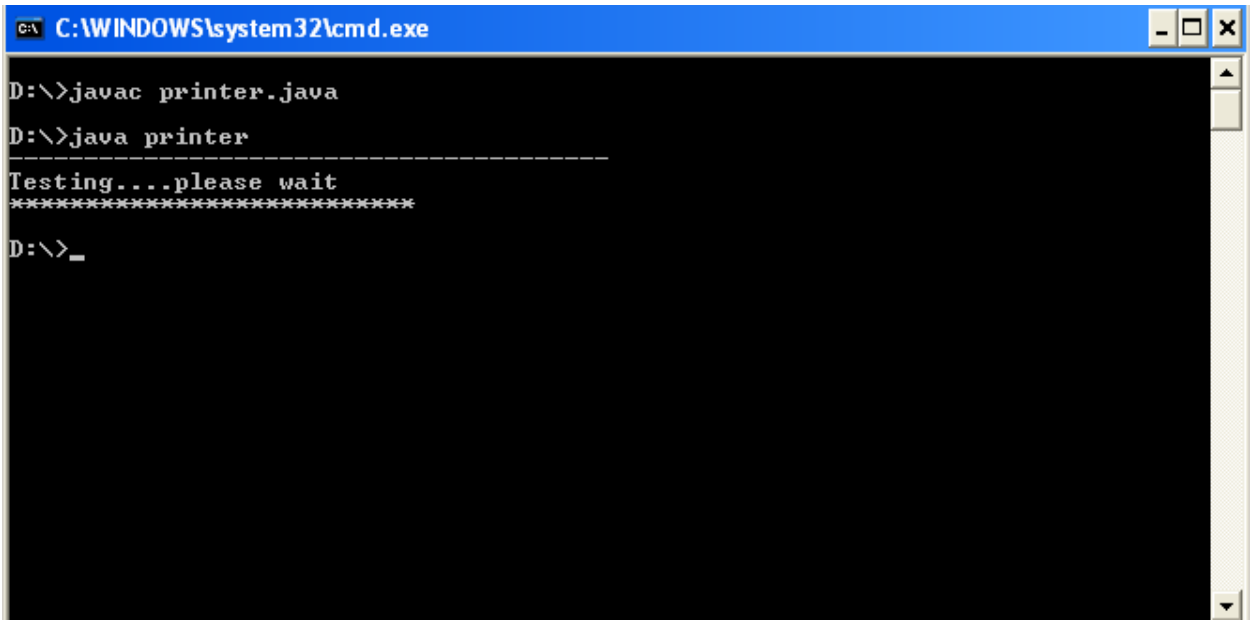
Save as printer.java

Note: three .class files get created

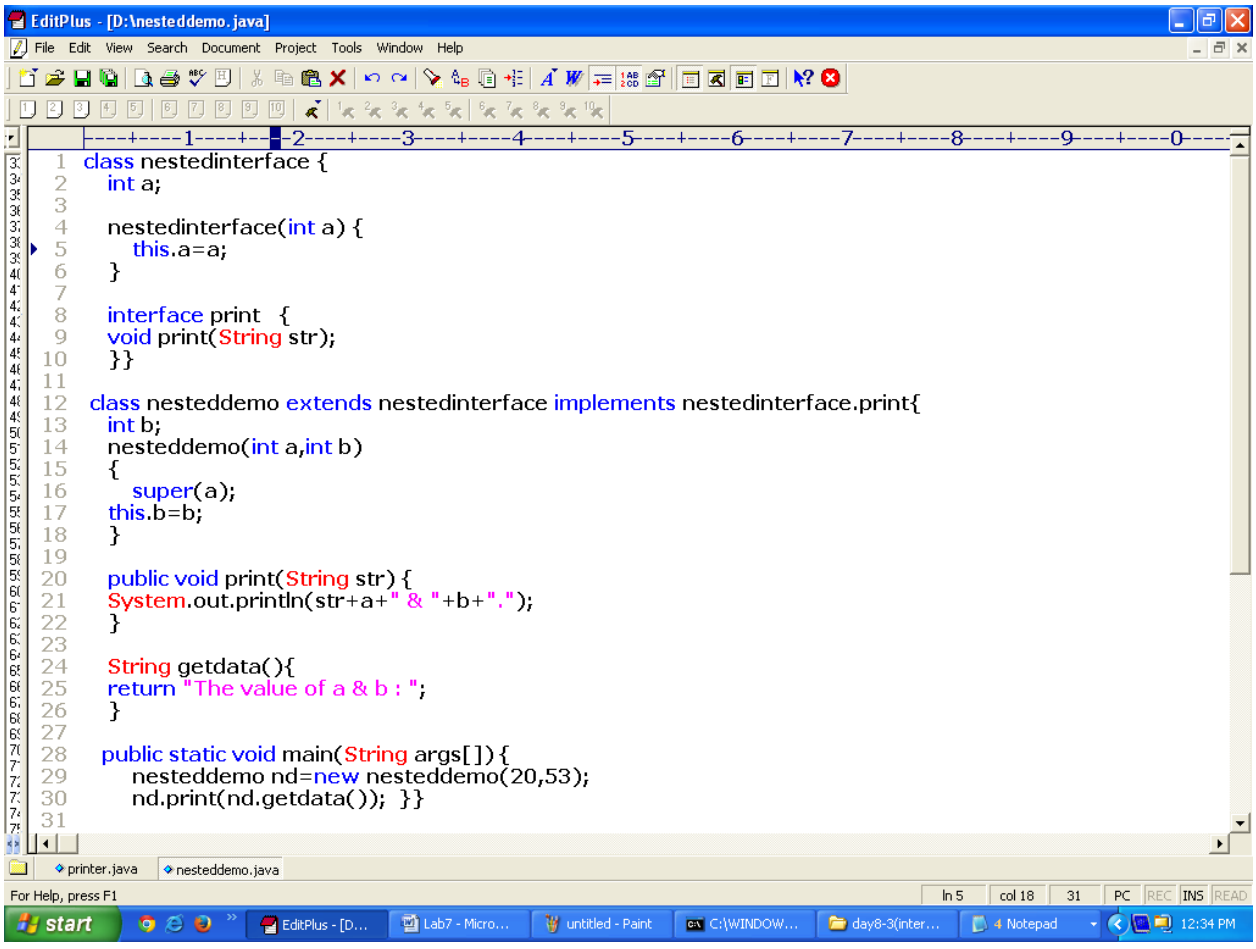
printable.class

testable.class

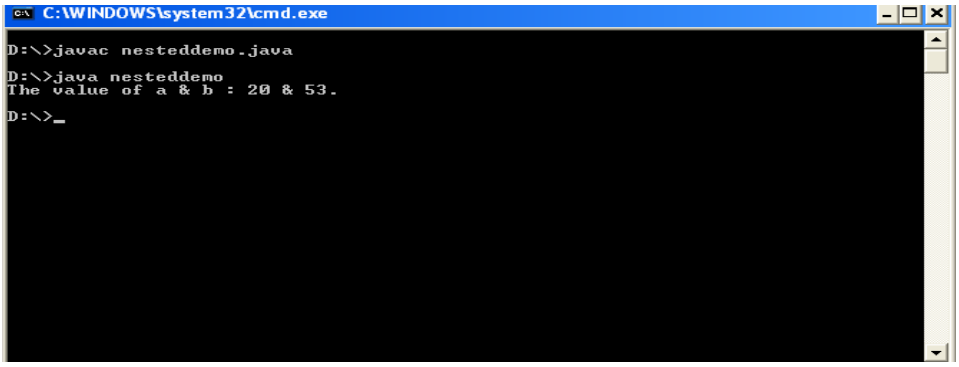
printer.class



Demo2



Save as nesteddemo.java



Assignments To Solve

1. Create an interface relational

```
interface relational
{
void greaterThan();
void lessThan();
void greaterThaneq();
void lessThaneq();
}
```

Write an implementing class implRel which implements realational

```
class  implRel implements relational
{
     int a, b;
          implRel(int a,int b)
          {
          this.a=a;
          this.b=b;
          }
              -   - -
 }
```

Create a class relationdemo .In main create an object of implRel (implementing class) and invoke all the methods …

2. Create an interface

```
interface MyMath
{
     double sqr(double a);
     double cube(double a);
     double cosine(double a);
     double sine(double a);
}
```

Write a class implMath which implements MyMath interface.
Write another class MathDemo ..In main create a reference of MyMath pointing to implMath class and invoke the methods..Accept the values from the user..

3. Create 2 classes **Student** and **Batch**, Student class is in pack1n and Batch class is in pack2. Write a **PrintData** class to print Student and Batch information.

4. Use the Student and Batch classes created earlier. it should contain public, protected, private and default attributes and using Batch class,
   Check accessibility of their attributes.

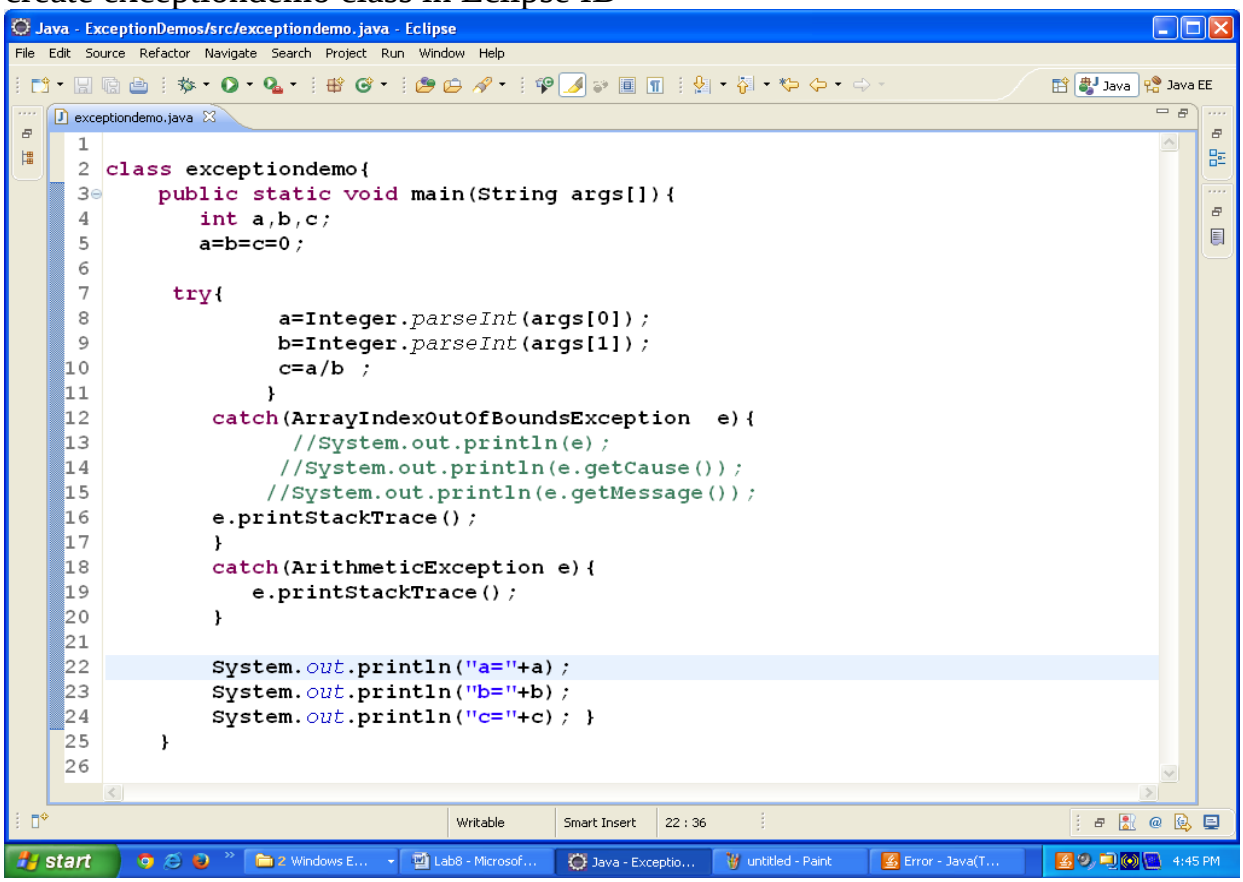
5. Create a package com.user .
   Now create a Greeter class in this package having the following features:
   Attributes:
   name string //indicates name of the person to be greeted
   Member functions:
   Greeter(aName) //constructor to initialize the name of the
   //person to be greeted by this greeter.
   sayHello() //returns a hello message with the name of the
   //person initialized earlier.
   sayGoodBye() //bids goodbye to the person named earlier.
   Create another class in the same package called Advisor
   Attributes:
   message string[5] //contains five advice messages
   Member functions:
   Advisor() //default constructor to initialize an array of
   //strings with atleast five advice messages
   getAdvice() //randomly selects an advice from the available
   //list of messages and returns it to the caller of this method
   Outside the package, from your working directory, create a class GreeterTest that constructs Greeter objects for all command-line arguments and prints out the results of calling sayHello().
   The program should then display an advice and finally bid goodbye to each of the persons/entities in reverse order of the names entered at the command line.
   java GreeterTest Mars Venus
   then the program should print
   Hello, Mars!
   Hello, Venus!
   Advice: Never say No
   Goodbye Venus!
   Goodbye Mars!

Lab-8 Assignments

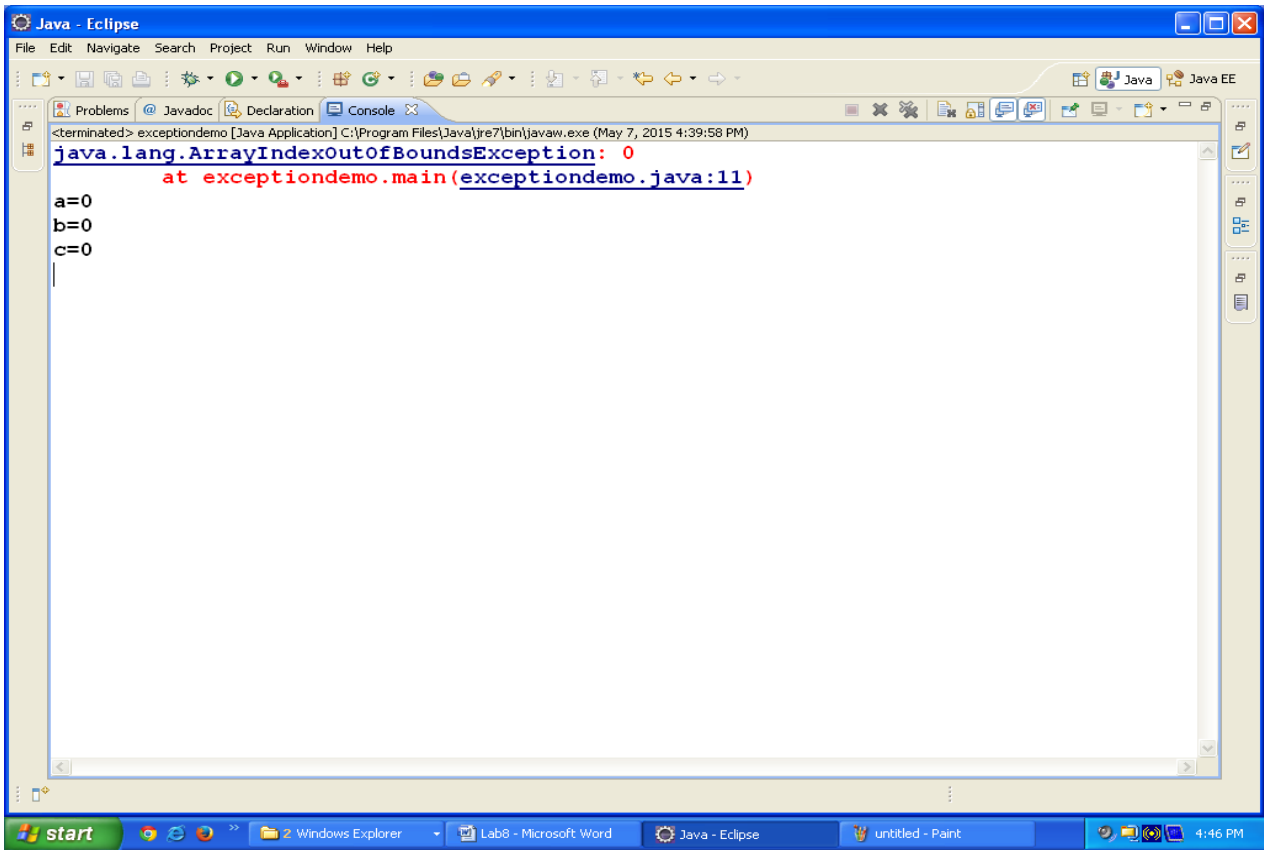
Exception Handling

Demo1

create exceptiondemo class in Eclipse ID

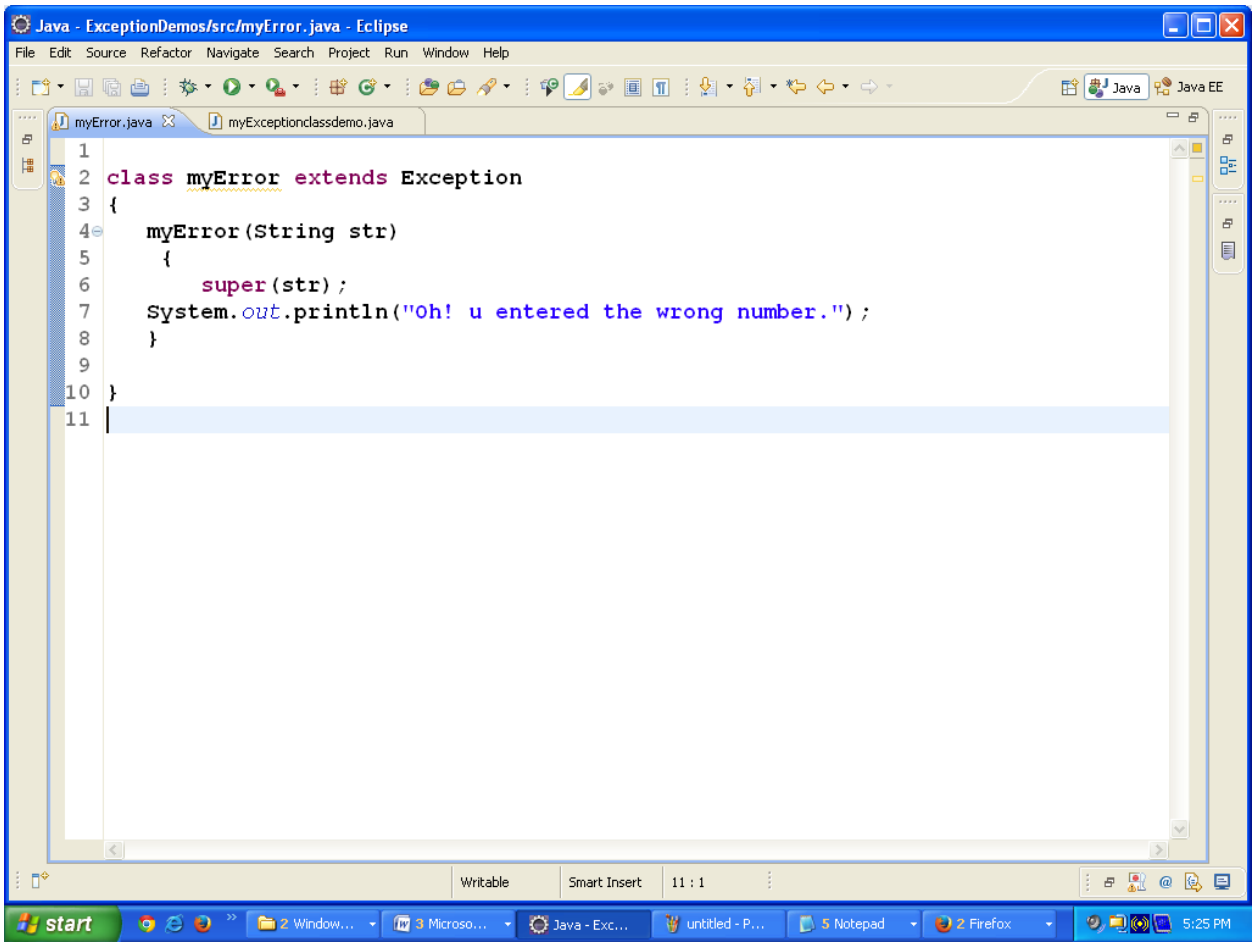


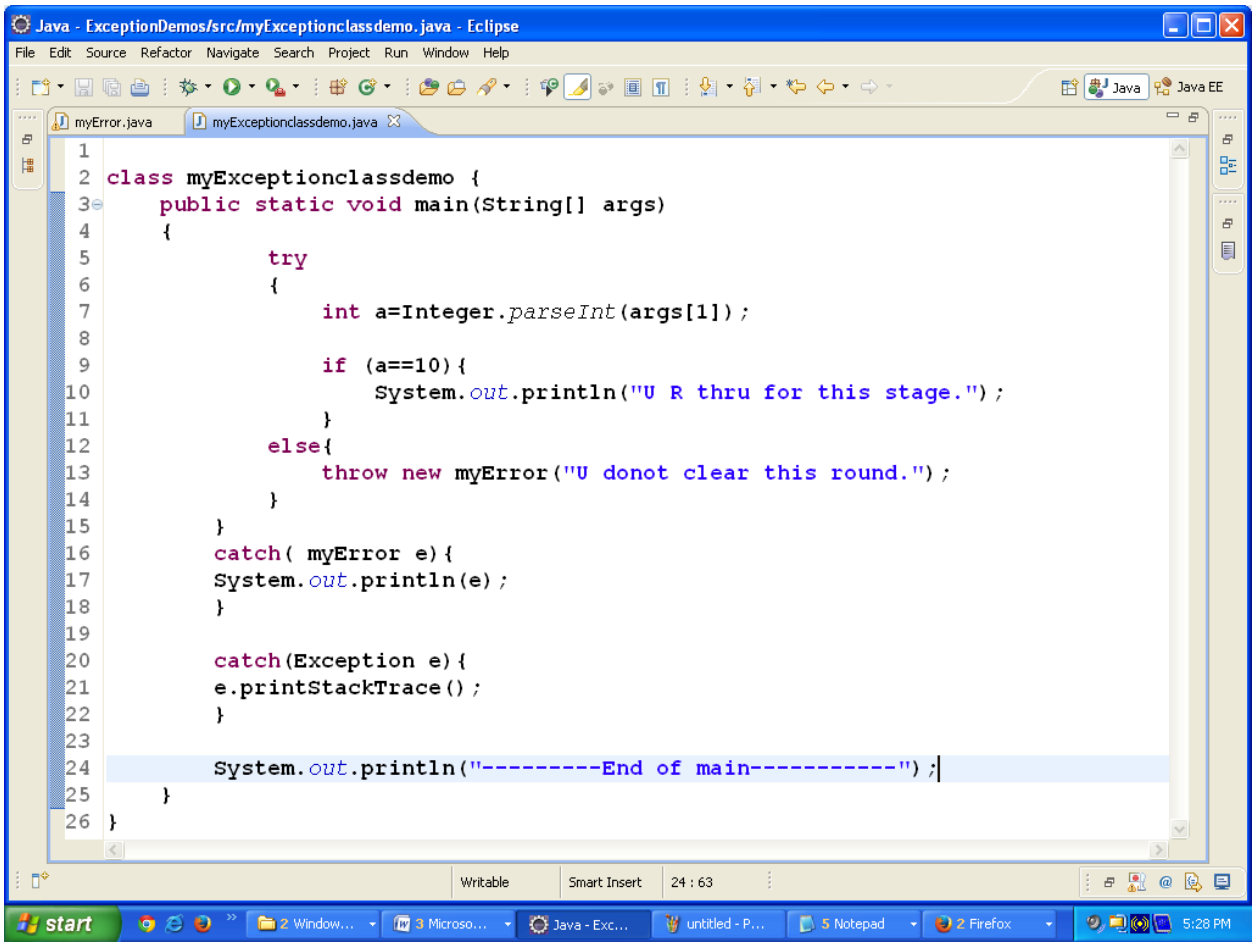
Run as -> Java Application



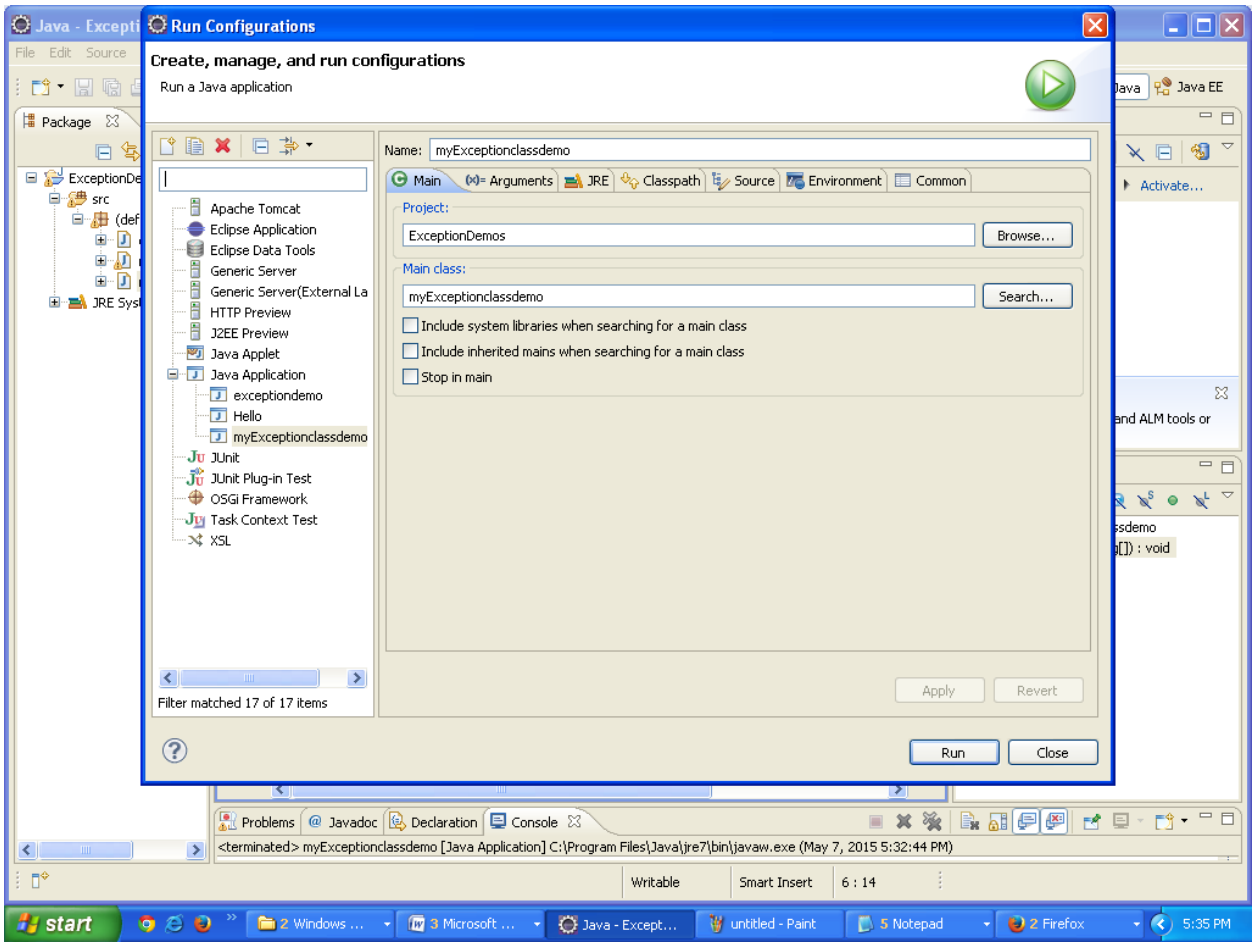
Demo2

create userdefined exception class **myError** and **myExceptiondemoclass**

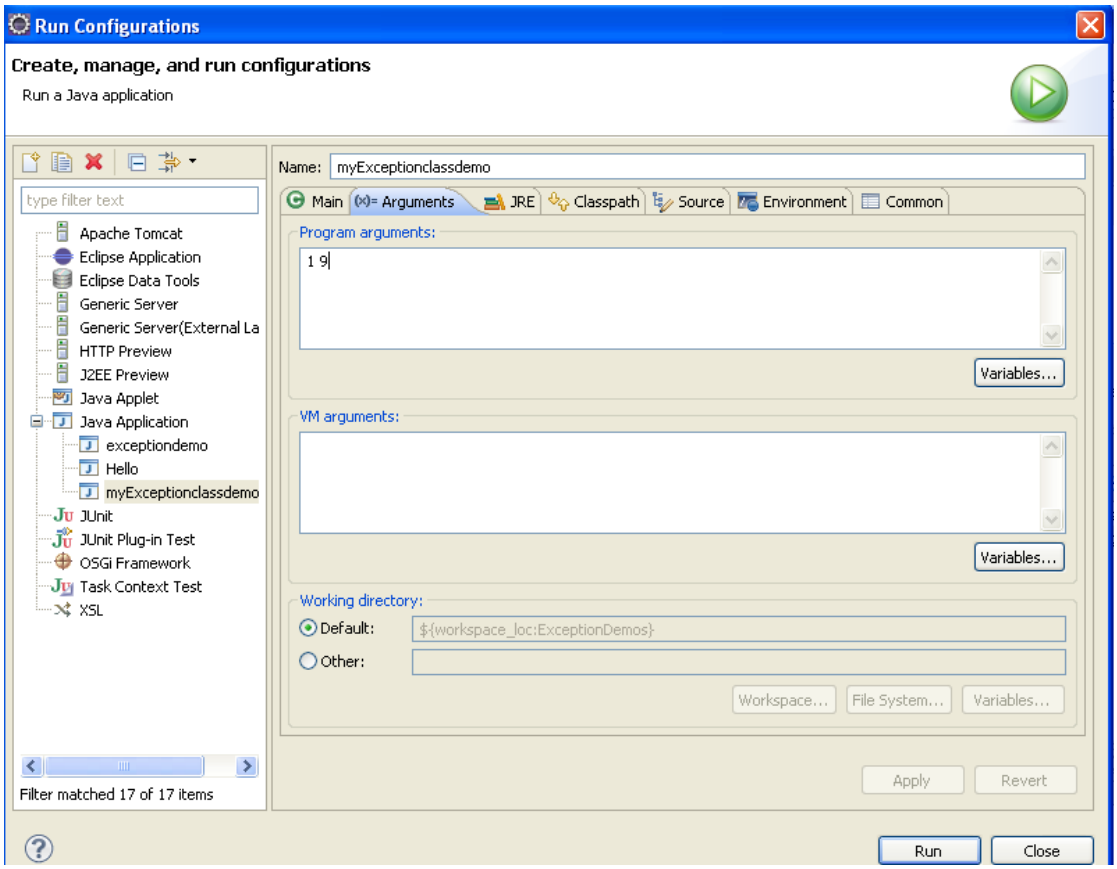

Java - ExceptionDemos/src/myError.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
myError.java
myExceptionclassdemo.java
class myError extends Exception
{
myError(String str)
{
super(str);
System.out.println("Oh! u entered the wrong number.");
}
}
Writable
Smart Insert
11 : 1
start
5:25 PM



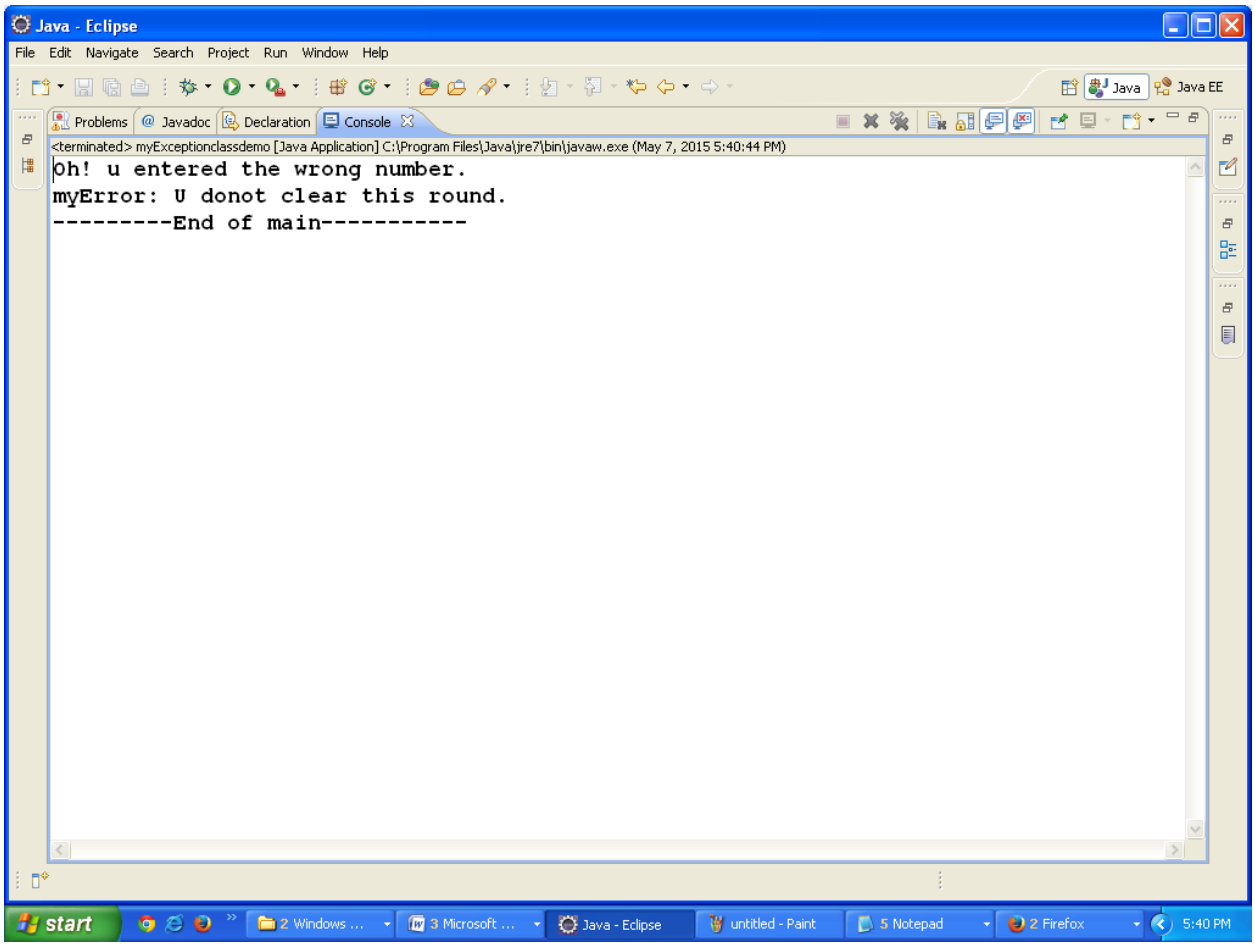
Steps for passing CLA(command Line Arguments in eclipse)
Run as -> Run Configurations…



Click on Arguments



Enter Arguments and click on Run



Assignments To Solve:

1. Write a class **TestException** to test different types of exceptions, such as NumberFormateException,ArrayIndexOutOfBoundsException,ArithMeticException,NullPointerException,ClassNotFoundException etc

2. Create a class called **CalcAverage** that has the following method:(use throw keyword)

public double avgFirstN(int N)

This method receives an integer as a parameter and calculates the average of first N natural numbers. If N is not a natural number, throw an exception IllegalArgumentException with an appropriate message.

3. Create a class **Number** having the following features:(use throw keyword)

Attributes

int first number

int	second number	
result	double	stores result of math operations performed on a & b

Member functions(Methods)

Number(x, y)	constructor to initialize the values of a and b
add()	stores the sum of a and b in result
sub()	stores difference of a and b in result
mul()	stores product in result
div()	stores a divided by b in result

Test to see if b is 0 and throw an appropriate exception since division by zero is undefined.

Display a menu to the user to perform the above four arithmetic operations.

4. Create a user defined Exception class called **InsufficientFundsException**

(Note: Use the existing **Account** Class (created in lab3))

Handle an exception for **Withdraw(int amt)** in Account class

Withdraw(int amt) should throw **InsufficientFundsException** if the amount to be withdrawn is greater than the balance.

Accept the values from user for creating Account object and amt to withdraw

5. Create a userdefined Exception classes **IncorrectAgeException** & **IncorrectNationalityException**

Write a class **Voter** with constructors and methods **toString(), check**(String name, int age)

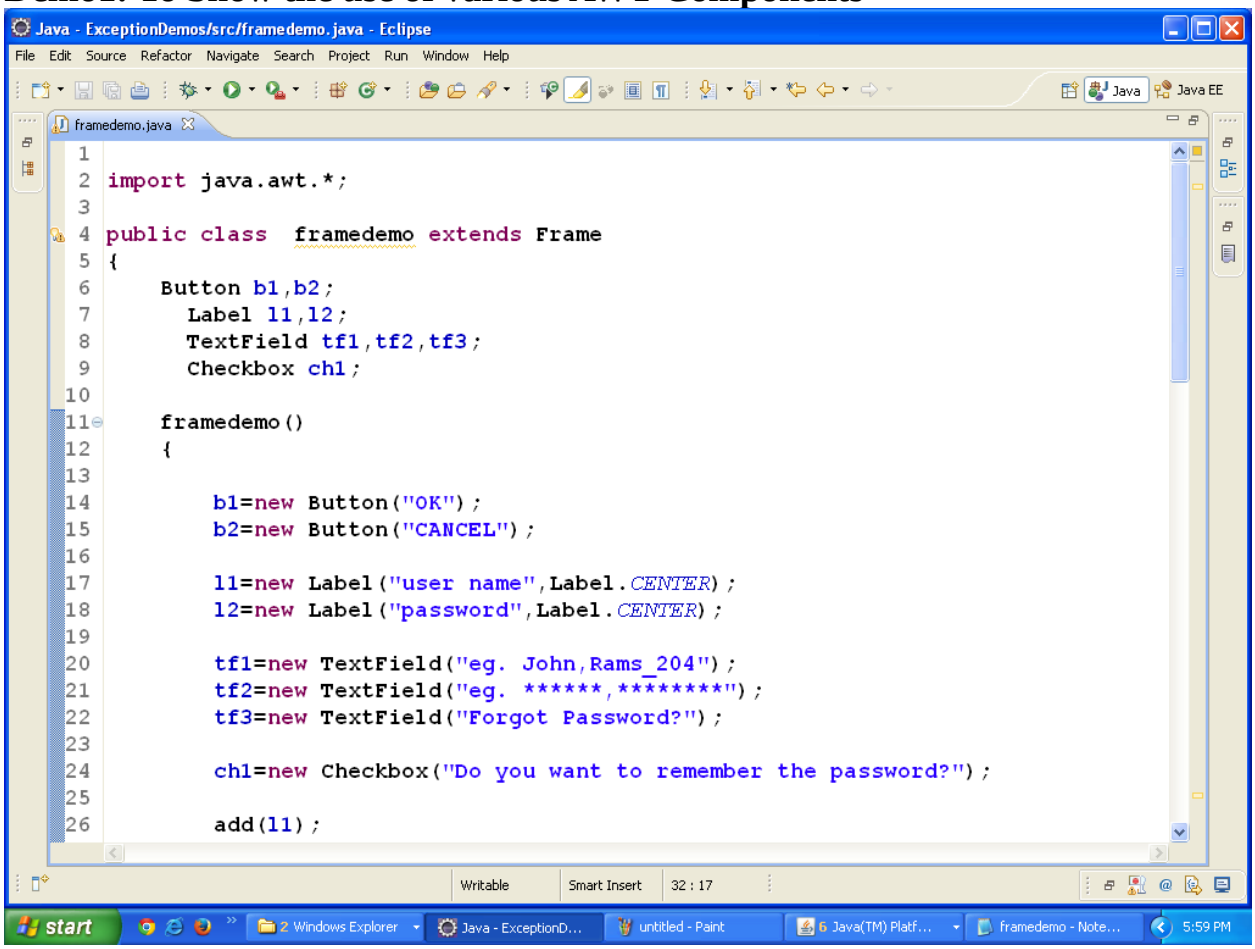
The check (…) should check for Nationality and age for voting and throw appropriate Exception

Accept the Name, Nationality & age from user

Lab-9 Assignments

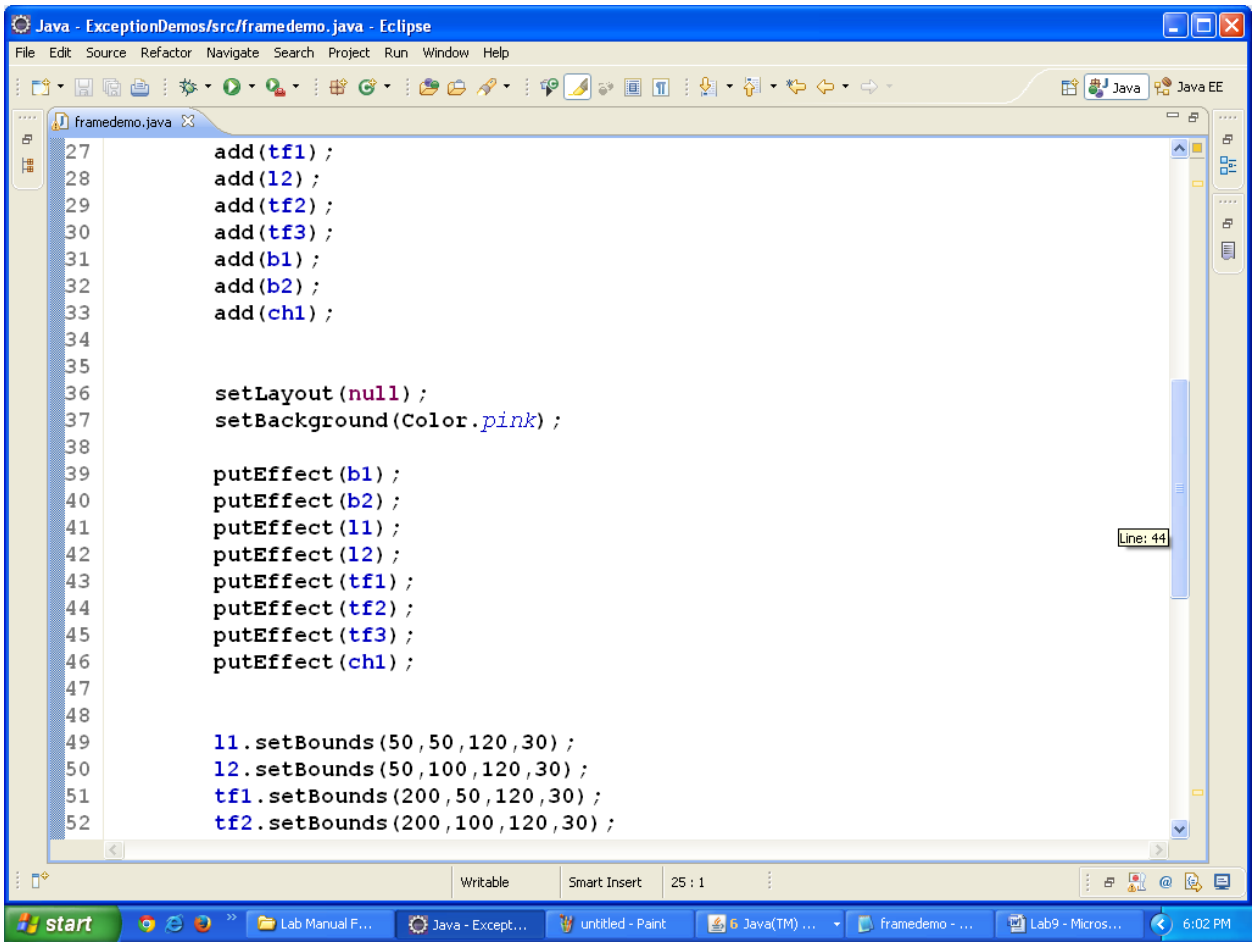
AWT(Abstract Window ToolKit)
1.Design frames using AWT classes

Demo1: To Show the use of various AWT Components

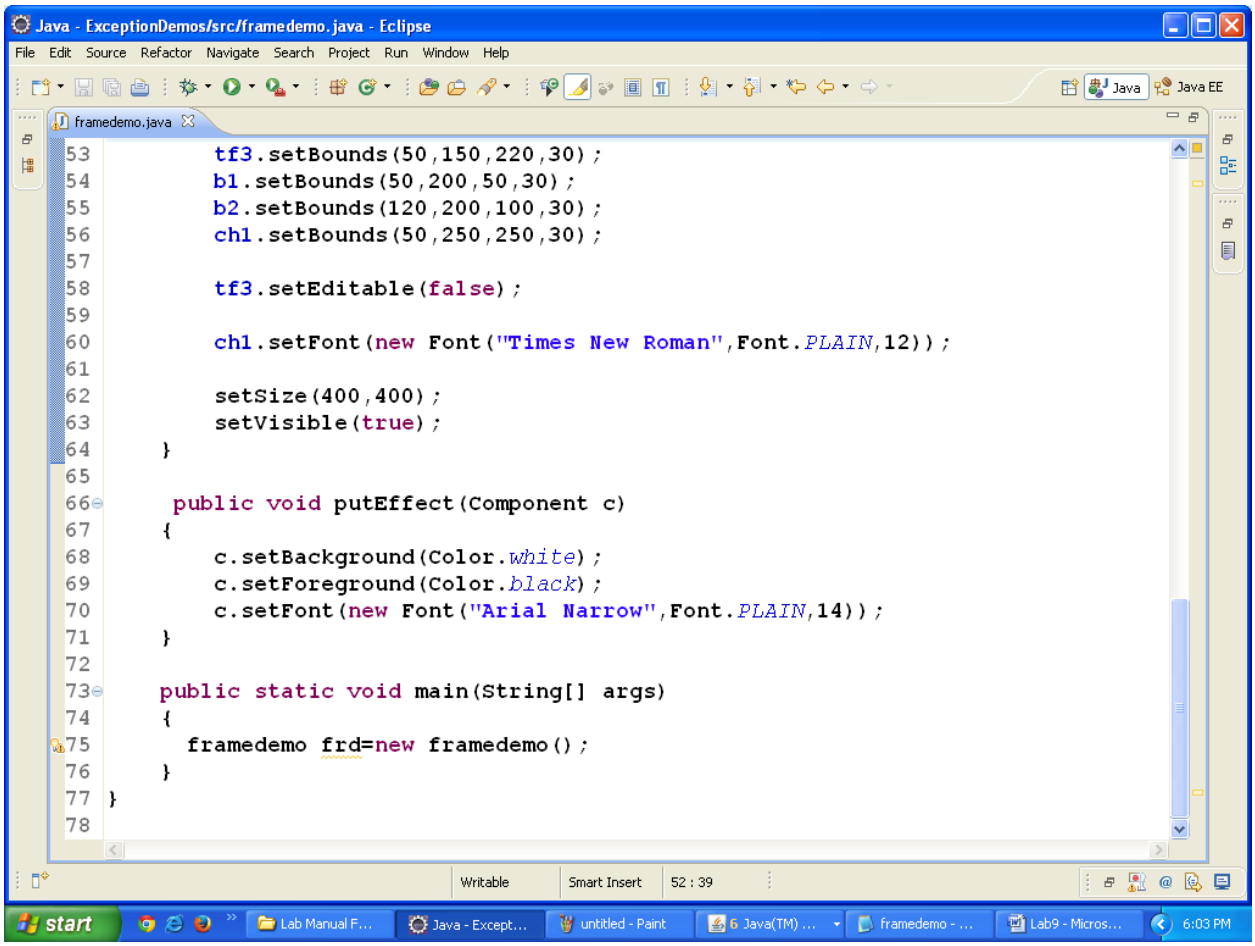



```
import java.awt.*;

public class  framedemo extends Frame
{
    Button b1,b2;
      Label l1,l2;
      TextField tf1,tf2,tf3;
      Checkbox ch1;

    framedemo()
    {

        b1=new Button("OK");
        b2=new Button("CANCEL");

        l1=new Label("user name",Label.CENTER);
        l2=new Label("password",Label.CENTER);

        tf1=new TextField("eg. John,Rams_204");
        tf2=new TextField("eg. ******,********");
        tf3=new TextField("Forgot Password?");

        ch1=new Checkbox("Do you want to remember the password?");

        add(l1);
```

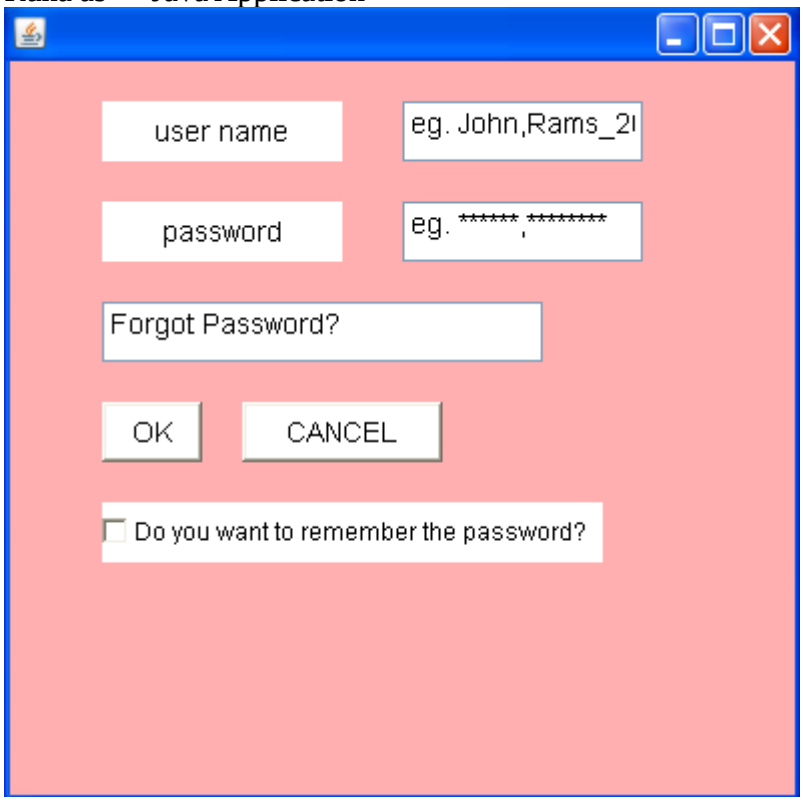
Continued…


Java - ExceptionDemos/src/framedemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
framedemo.java
add(tf1);
add(l2);
add(tf2);
add(tf3);
add(b1);
add(b2);
add(ch1);
setLayout(null);
setBackground(Color.pink);
putEffect(b1);
putEffect(b2);
putEffect(l1);
putEffect(l2);
putEffect(tf1);
putEffect(tf2);
putEffect(tf3);
putEffect(ch1);
l1.setBounds(50,50,120,30);
l2.setBounds(50,100,120,30);
tf1.setBounds(200,50,120,30);
tf2.setBounds(200,100,120,30);
Line: 44
Writable
Smart Insert
25 : 1
start
6:02 PM

Continued..

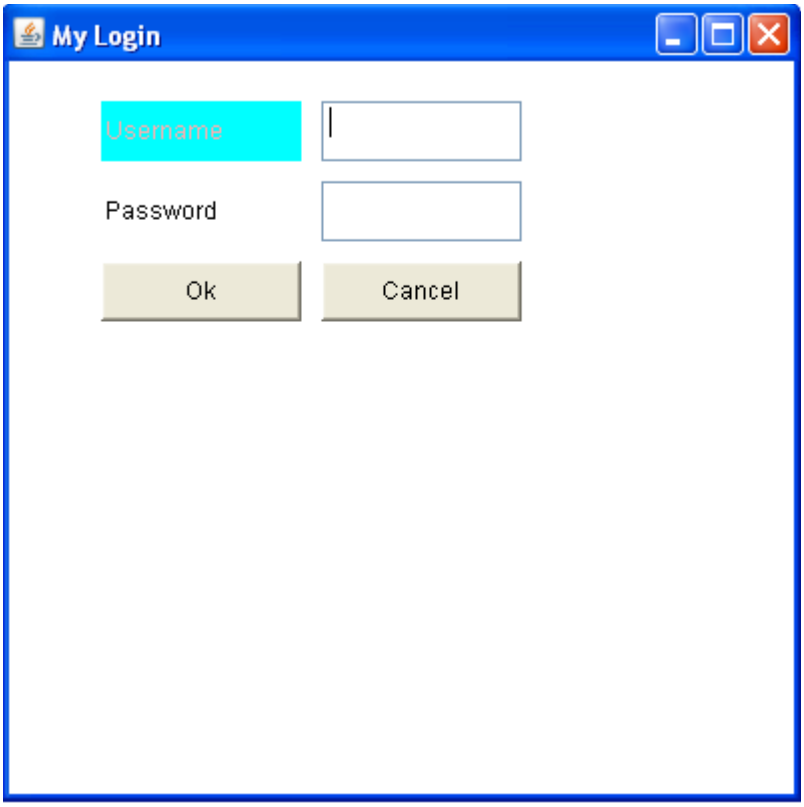


Runa as -> Java Application



Assignments To Solve

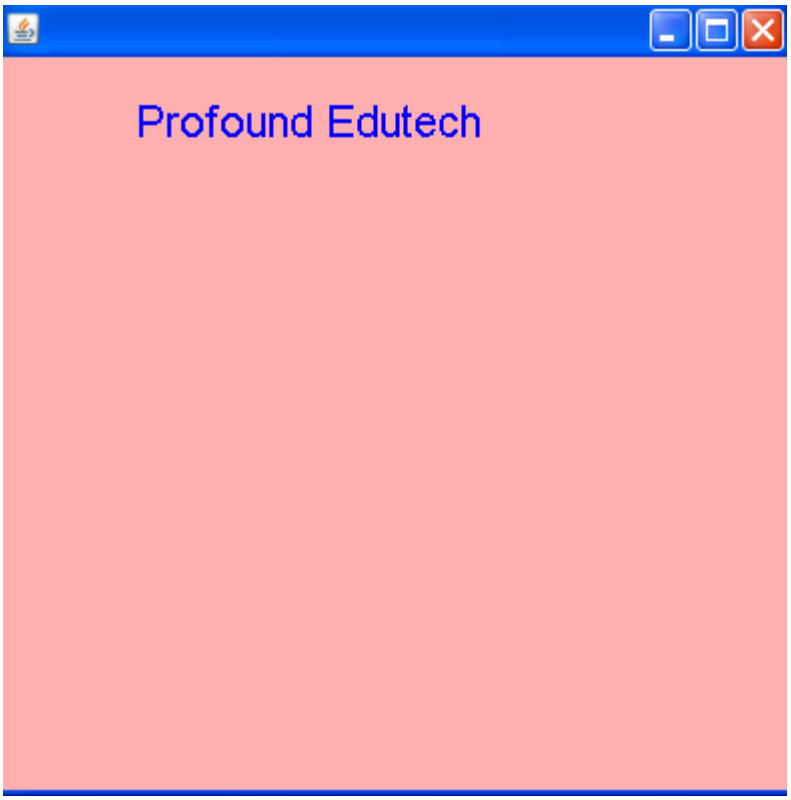
3. Generate the Login Page as shown below. (Event for which should be handled in the next lab..)



4. Generate the below page.. Draw 3 Scrollbars with backgrounds.(Event for which should be handled in the next lab..)

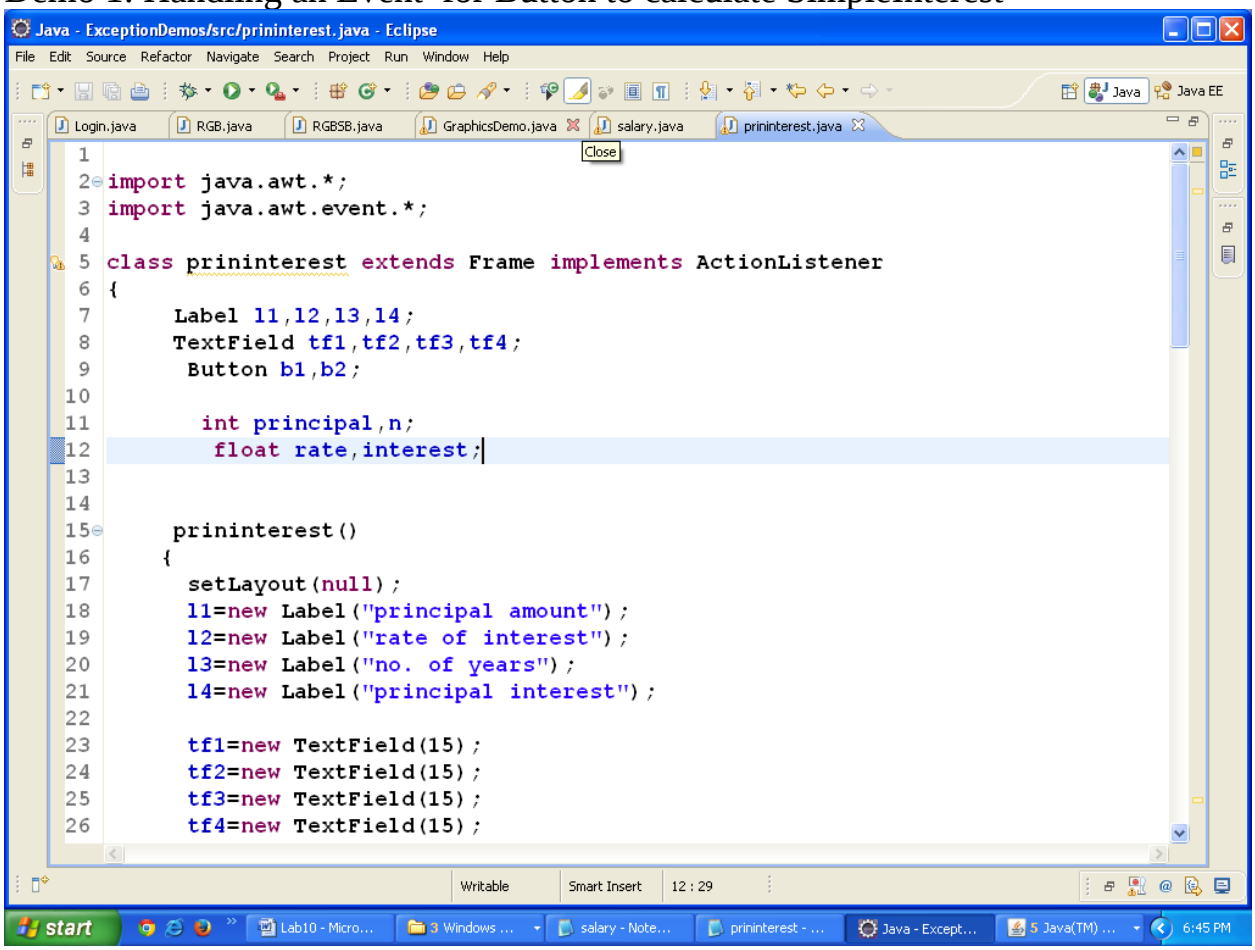
5. WAP to print your name with the particular Font on the Frame

   Like….

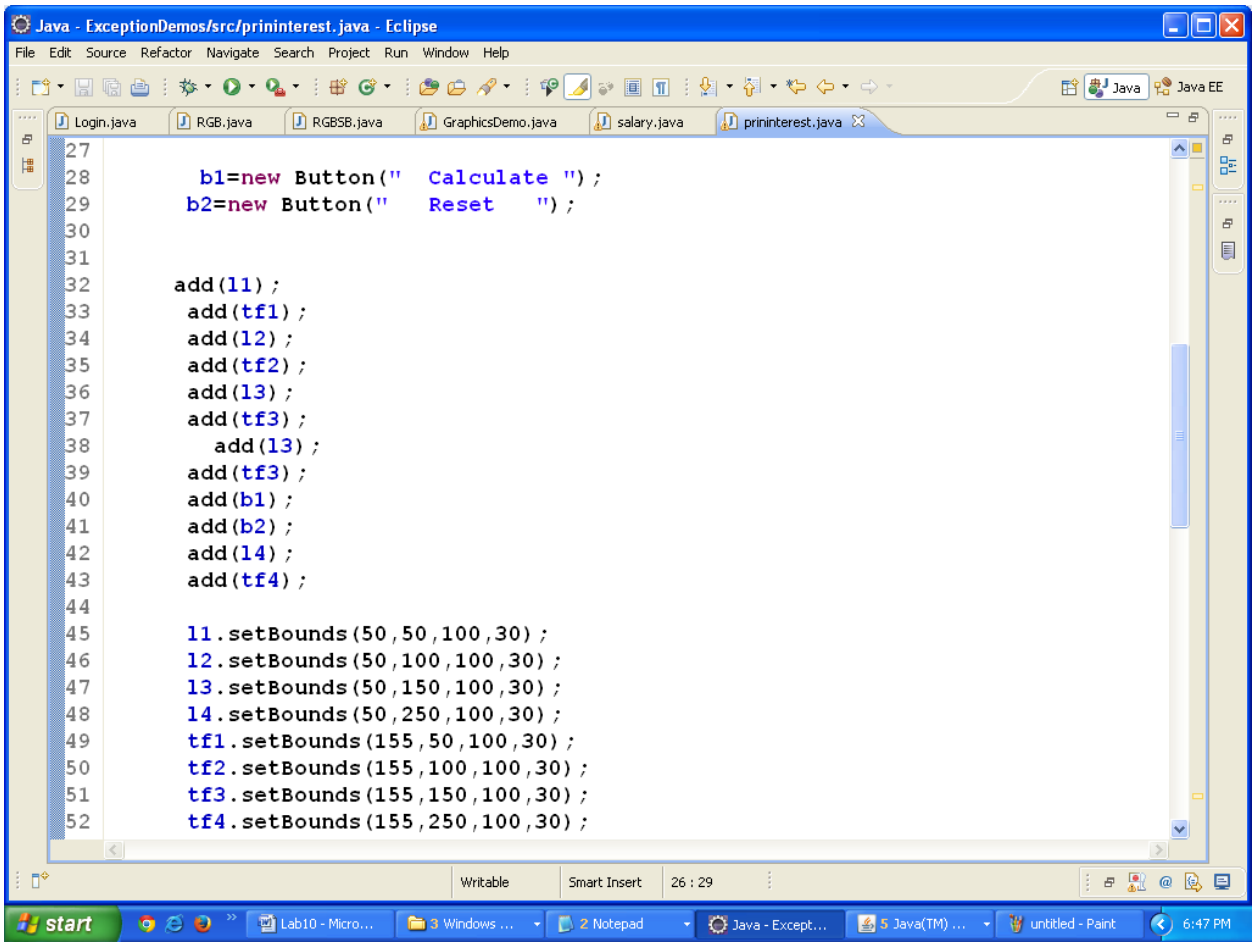


Lab-10 Assignments
Event Handling

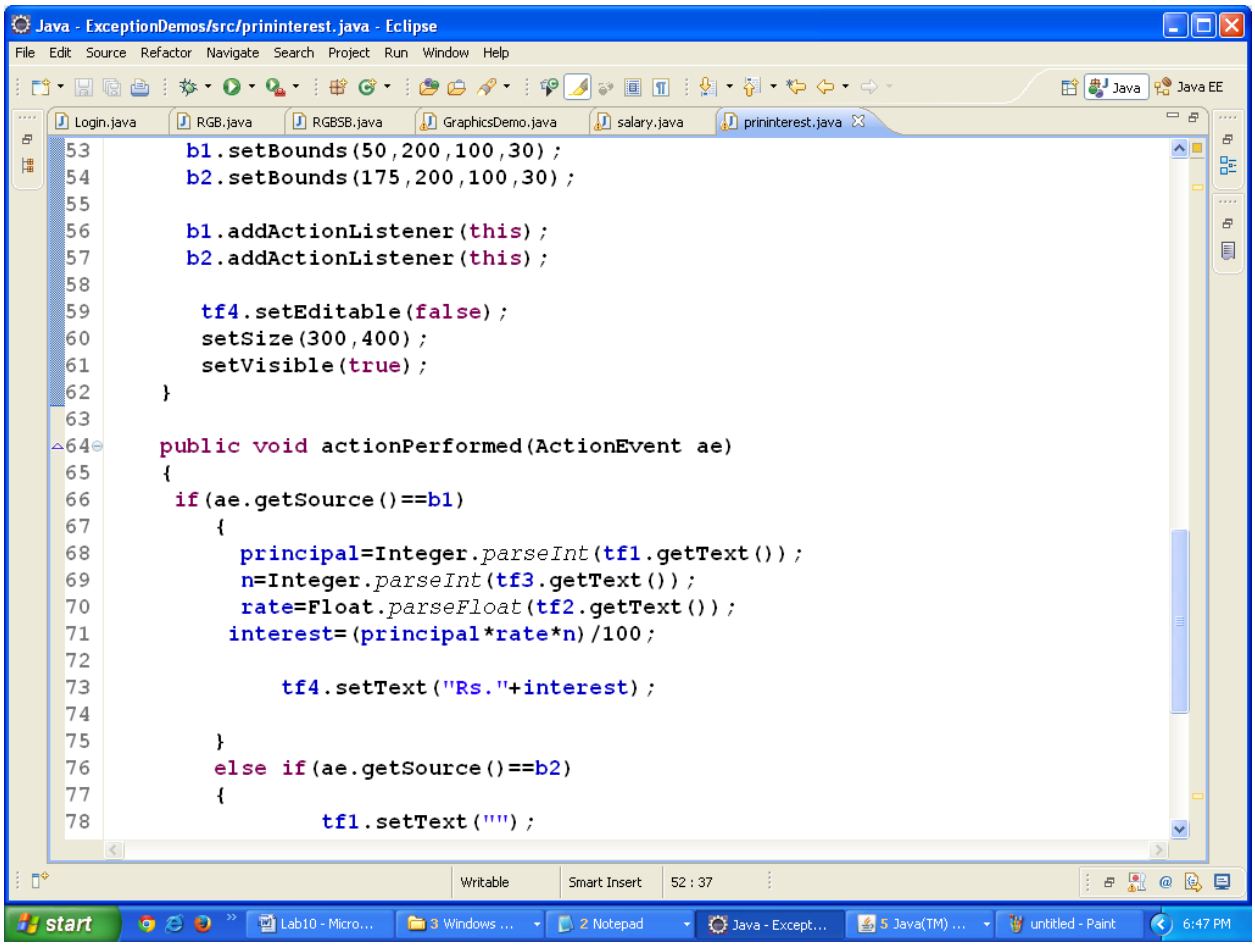
Demo 1: Handling an Event for Button to calculate SimpleInterest



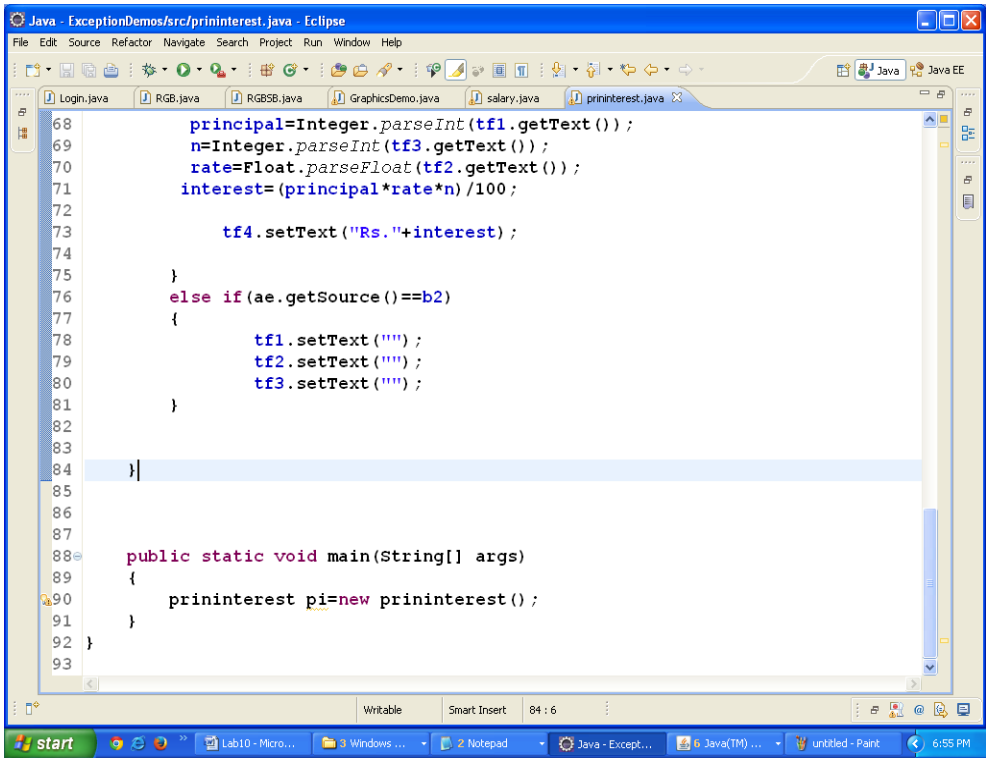
Continued…


Java - ExceptionDemos/src/prininterest.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Java
Java EE
Login.java
RGB.java
RGBSB.java
GraphicsDemo.java
salary.java
prininterest.java
b1=new Button(" Calculate ");
b2=new Button(" Reset ");
add(l1);
add(tf1);
add(l2);
add(tf2);
add(l3);
add(tf3);
add(l3);
add(tf3);
add(b1);
add(b2);
add(l4);
add(tf4);
l1.setBounds(50,50,100,30);
l2.setBounds(50,100,100,30);
l3.setBounds(50,150,100,30);
l4.setBounds(50,250,100,30);
tf1.setBounds(155,50,100,30);
tf2.setBounds(155,100,100,30);
tf3.setBounds(155,150,100,30);
tf4.setBounds(155,250,100,30);
Writable
Smart Insert
26 : 29
start
Lab10 - Micro...
3 Windows ...
2 Notepad
Java - Except...
5 Java(TM) ...
untitled - Paint
6:47 PM

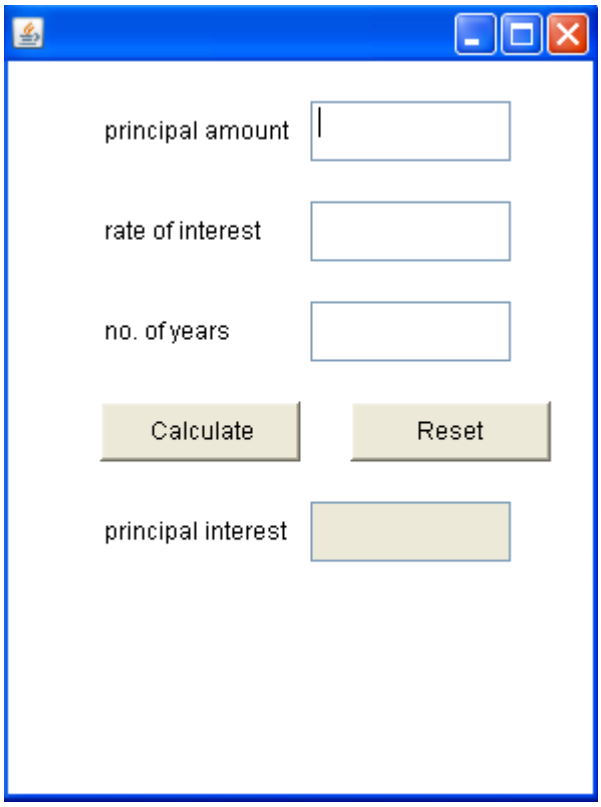
Continued…



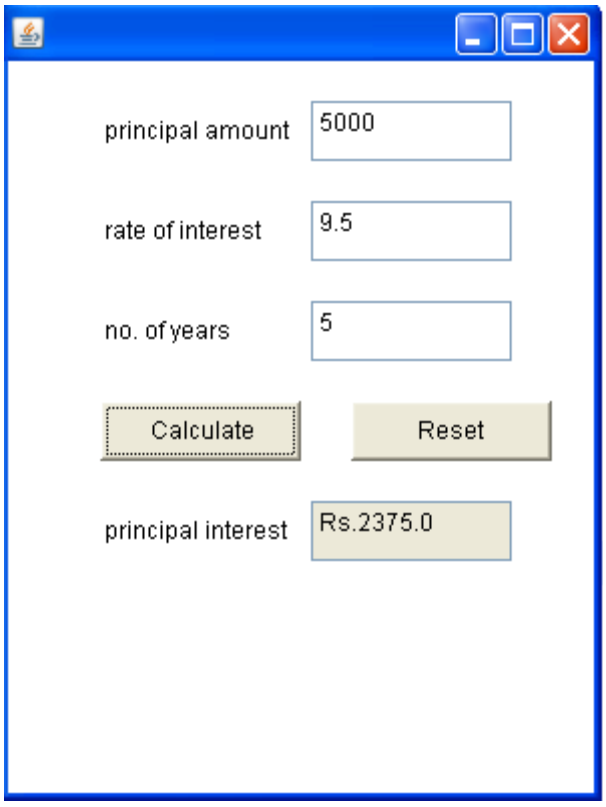
Continued…



Run as-> Java Application



Enter the values and click on Calculate Button(Event Handling)



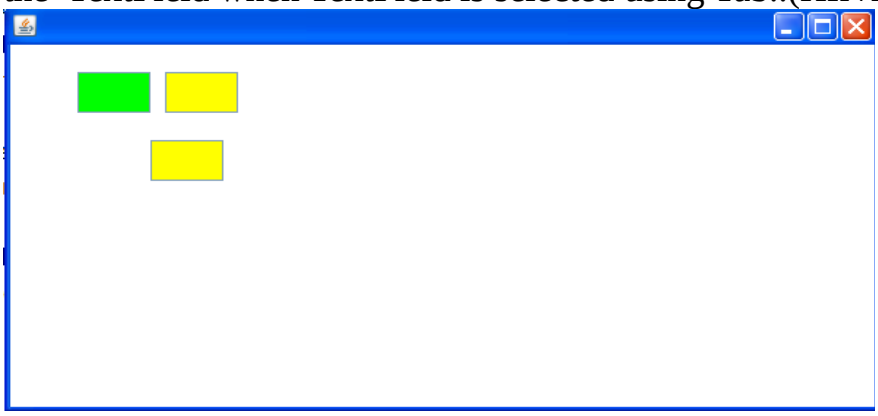
Assignments To Solve

1. Validate Username and Password for LoginPage created in the previous lab
   Print Access Granted on the Frame for valid data
   Print Access Denied on the Frame for Invalid data
   Handle this event for the click of **OK** Button
   Close the Frame at the click of **Cancel** Button

2. Write an event handling code to print your name on the frame at the click of Mouse Button.(hint : Mouse Event)

3. Handle an event for the Scrollbars created in the previous lab.
   Set the background color of the frame with the Scroll values of the 3 Scrollbars.

4. Handle an event for the TextFields shown below . Show the Focus gained for the TextField when TextField is selected using Tab..(HINT: Use FocusListener)

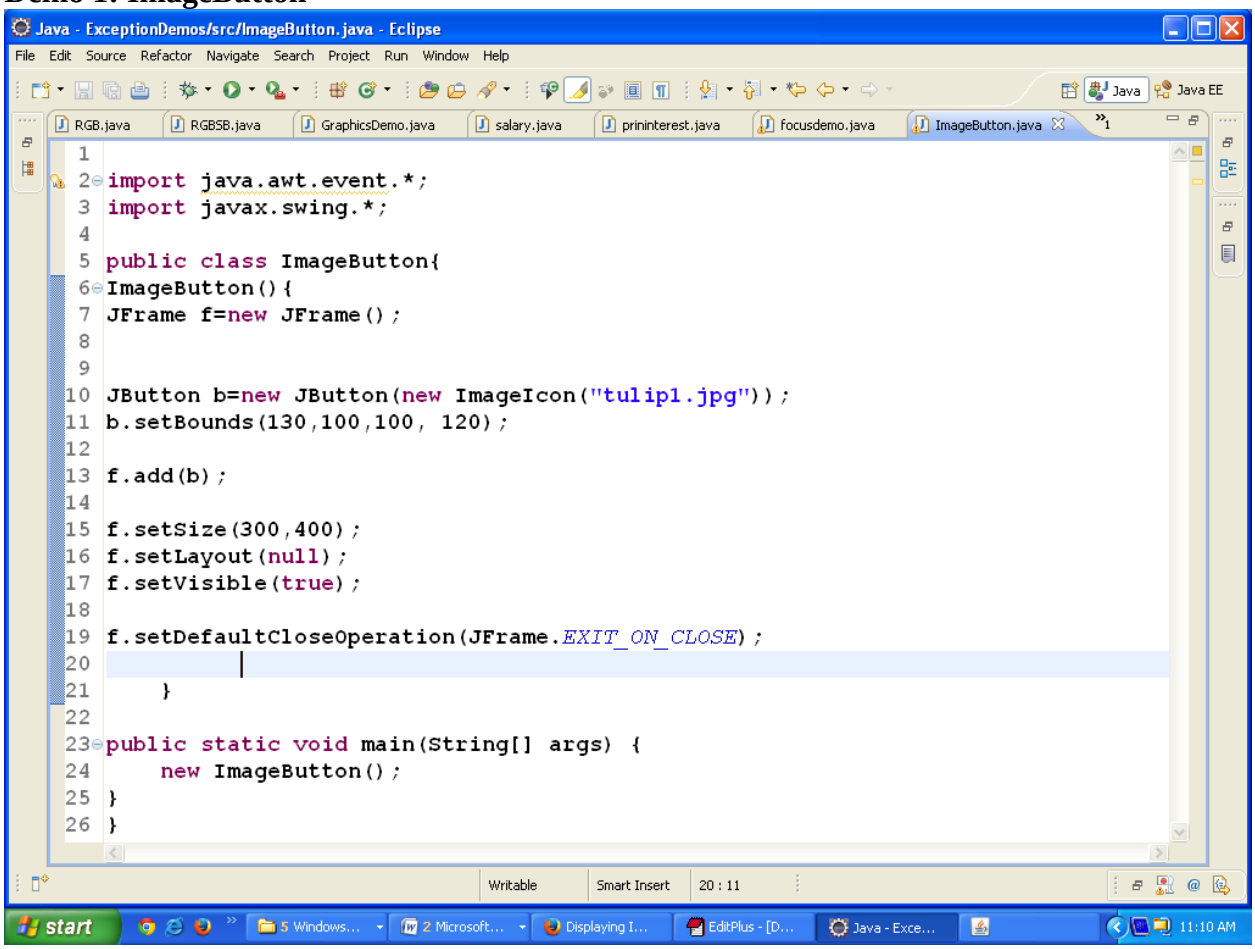


Lab-11 Assignments

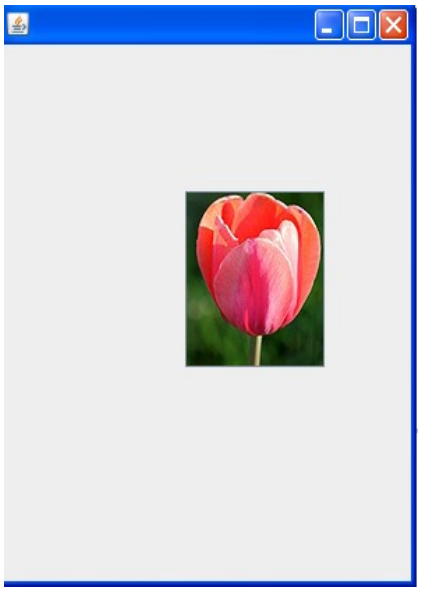
Swing Components

MenuBar

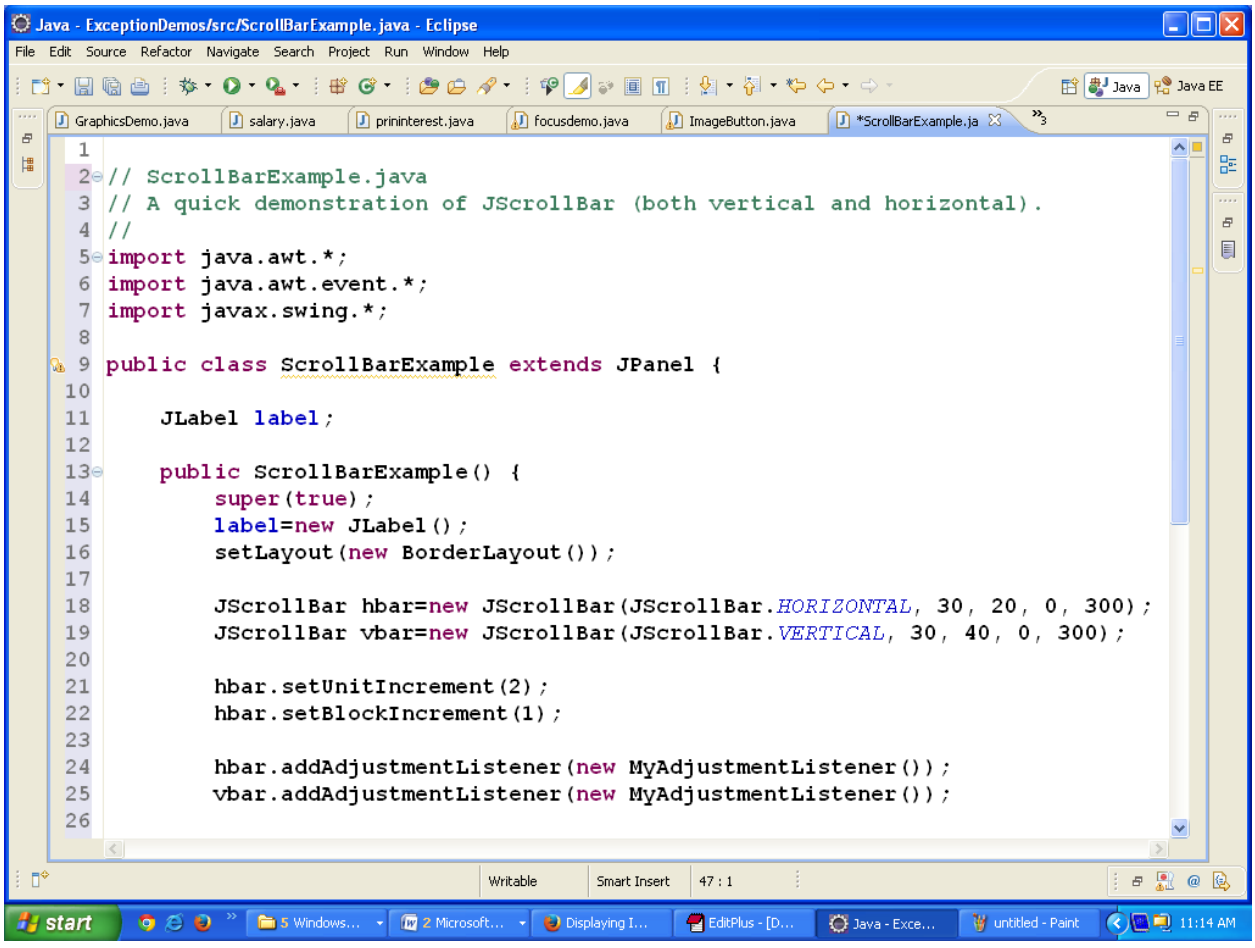
Demo 1: ImageButton



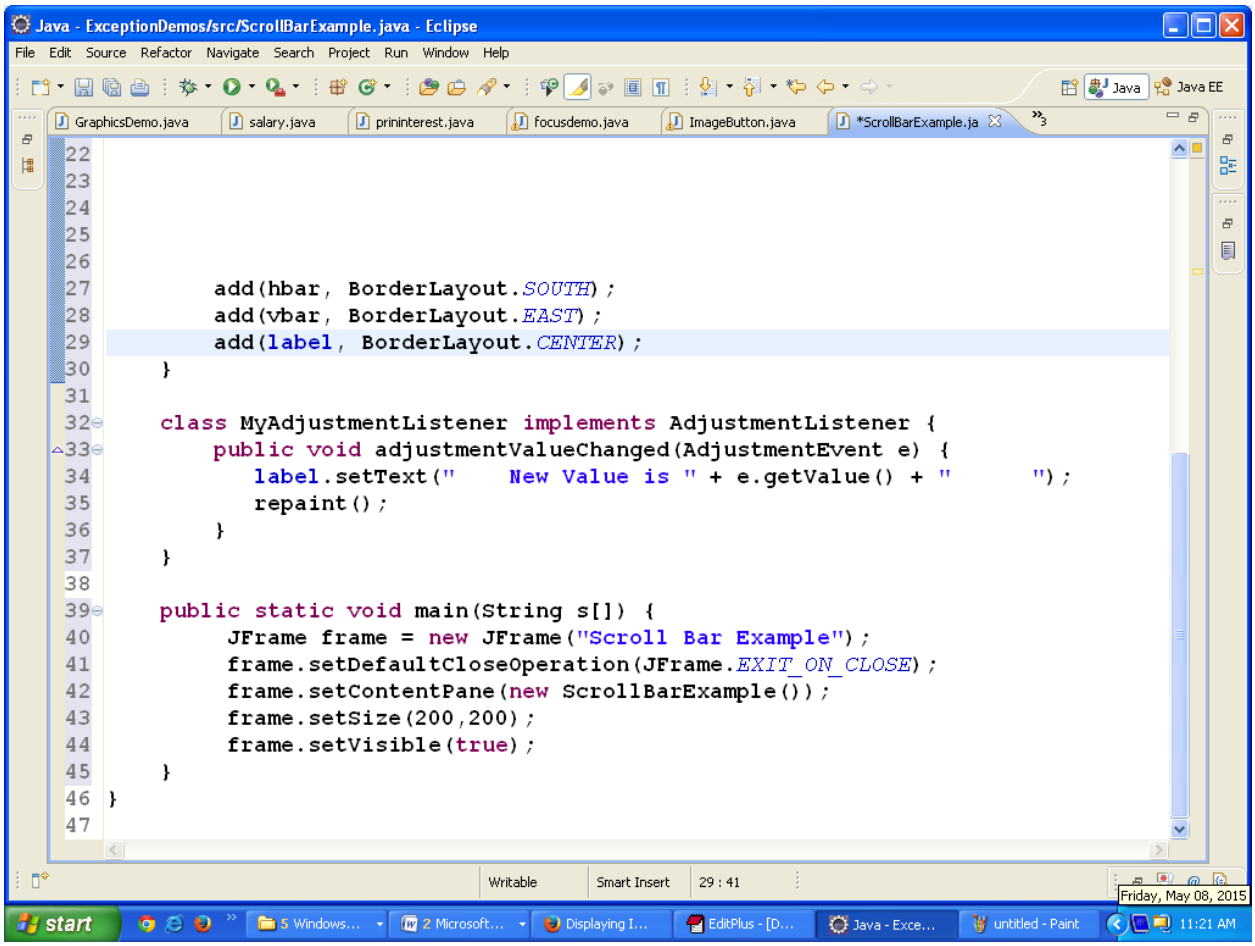
Run as -> Java Application



Demo 2: ScrollBarDemo



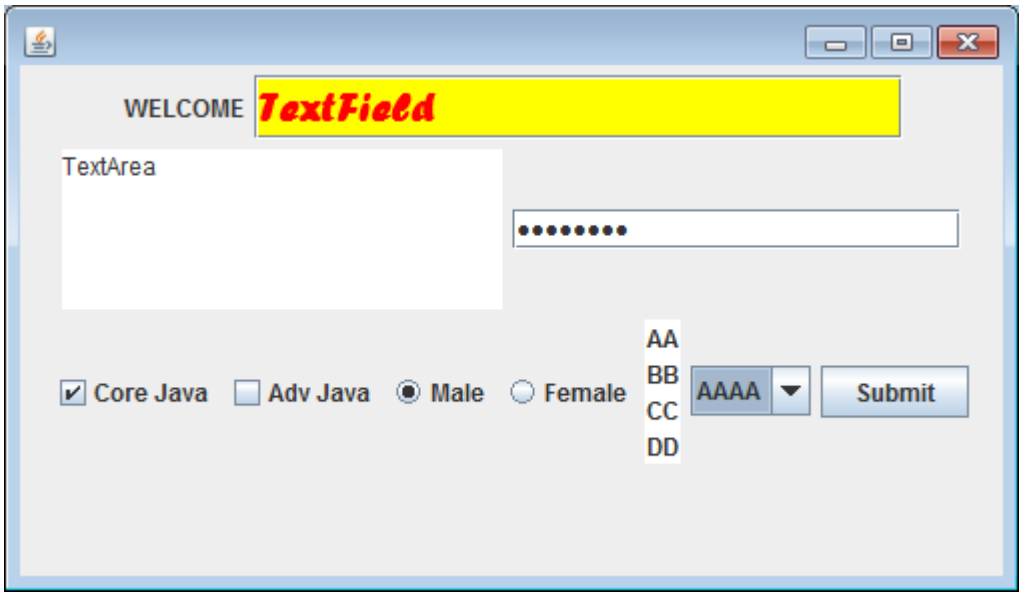
Continued…



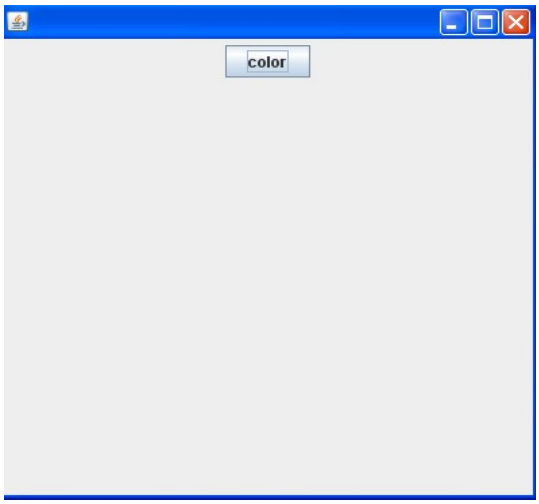
Run as -> Java Application

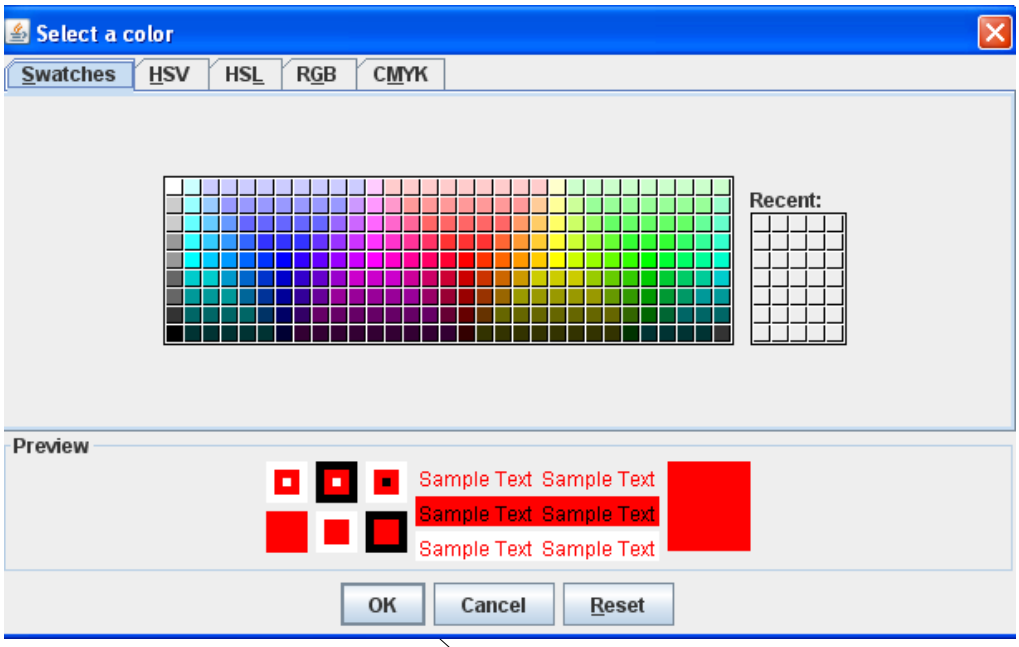
Assignments To Solve

1. **Design below form displaying basic swing controls.**

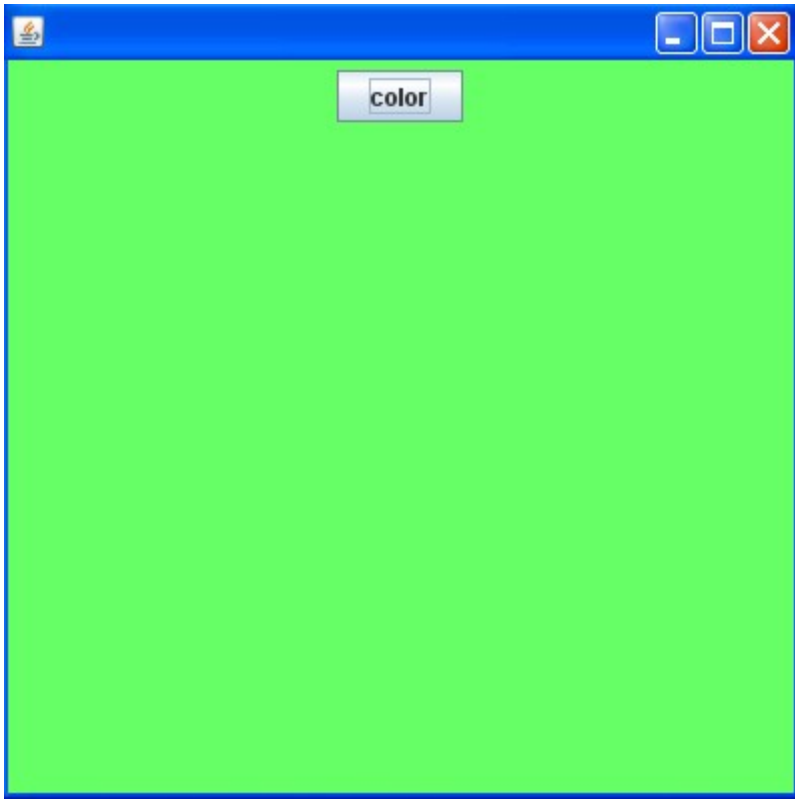


2. WAP to handle an event for a button to set the background color of the frame by selecting color from JColorChooser class.

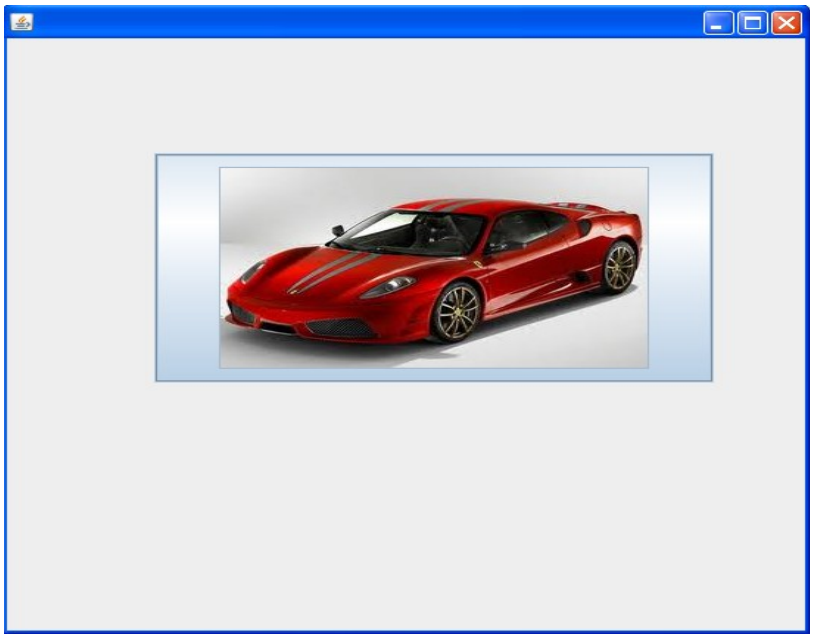




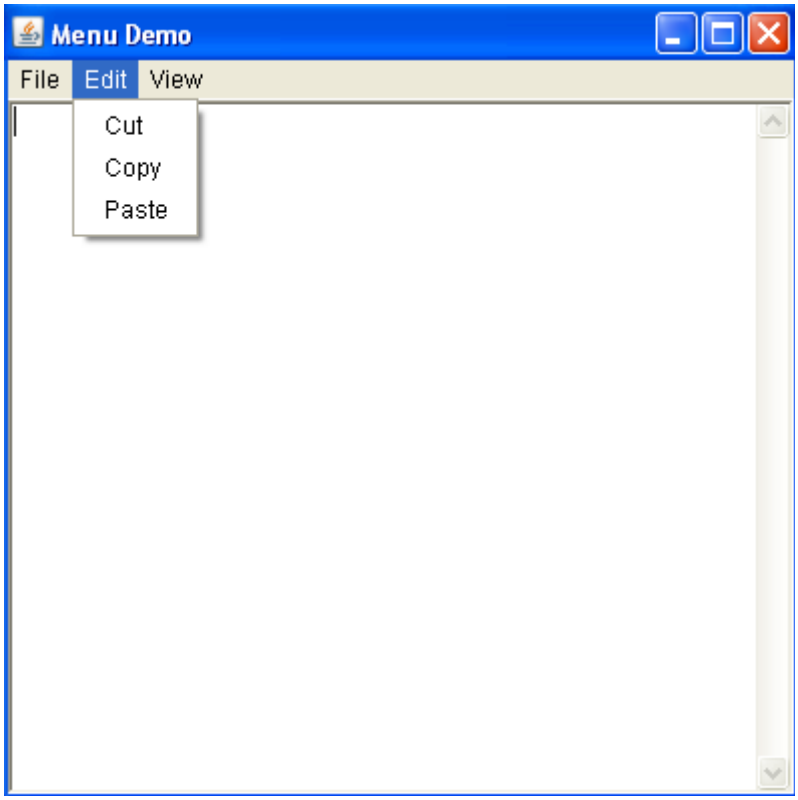
Select and press ok



3. Write an event driven program to move the car up, down,left and right (KeyEvent)



4. Write a java program to design notepad.(Use Jmenubar,Jmenu,JmenuItem,JtextArea)



Lab-12 Assignments
IO programming or File handling

1. Write java program to print file specification such as isFile,isDirectory,last modified date,file size,file patch etc

2. Write a java program to read & write the content to and from “myFile.txt” using FileInputStream and FileOutputStream API

3. Java program to read content from one file and write it into another file.

4. Write a java program to do read/write operation using buffered classes

5. Write a java program that takes a file name and a search string from the user, if the search string occures in the file, then it counts the no. Of occurences of the string
   Ex: output :Enter a file name : test.txt
   Enter a word : you

6. Write a java program to read console data using BufferedReader API

7. Write a Java program to count no. Of words into file

8. Write a java program to read & write the content to and from “myFile.txt” using RandomAccessFile

9. Write a java program to serialize the Student objects and deserialize it(Serializaion)

10. Write a java program to serialize the Employee objects(id,name,salary,address,email,contact) and restrict serialization of email and contact (use ObjectInputStream,ObjectOutputStream,transient)

Lab-13 Assignments

Collections & Generics

Collection APIs

List :ArrayList,LinkedList, Vector

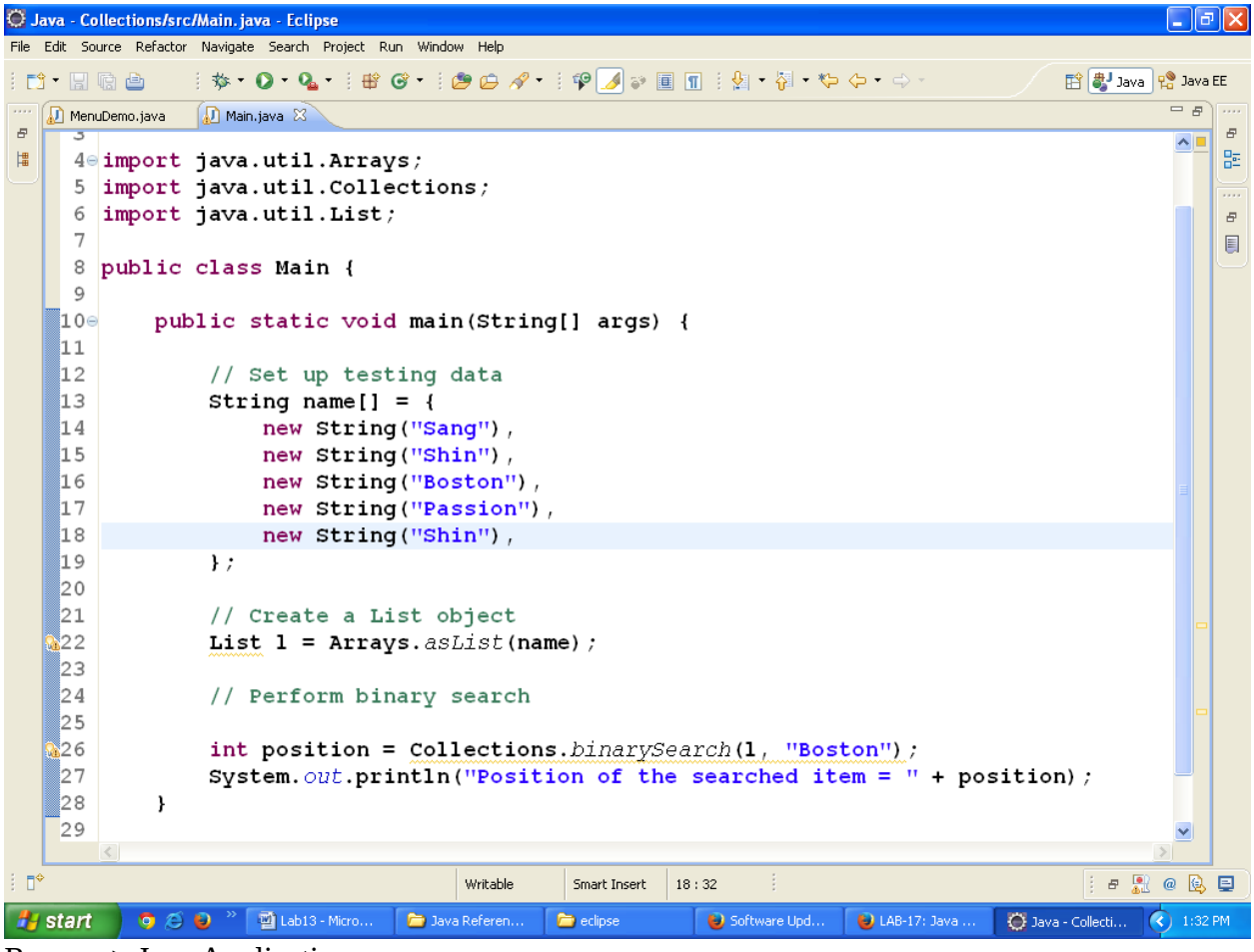
Set :HashSet,LinkedHashSet,TreeSet

Map : HashMap,LinkedHashMap, TreeMap, HashTable

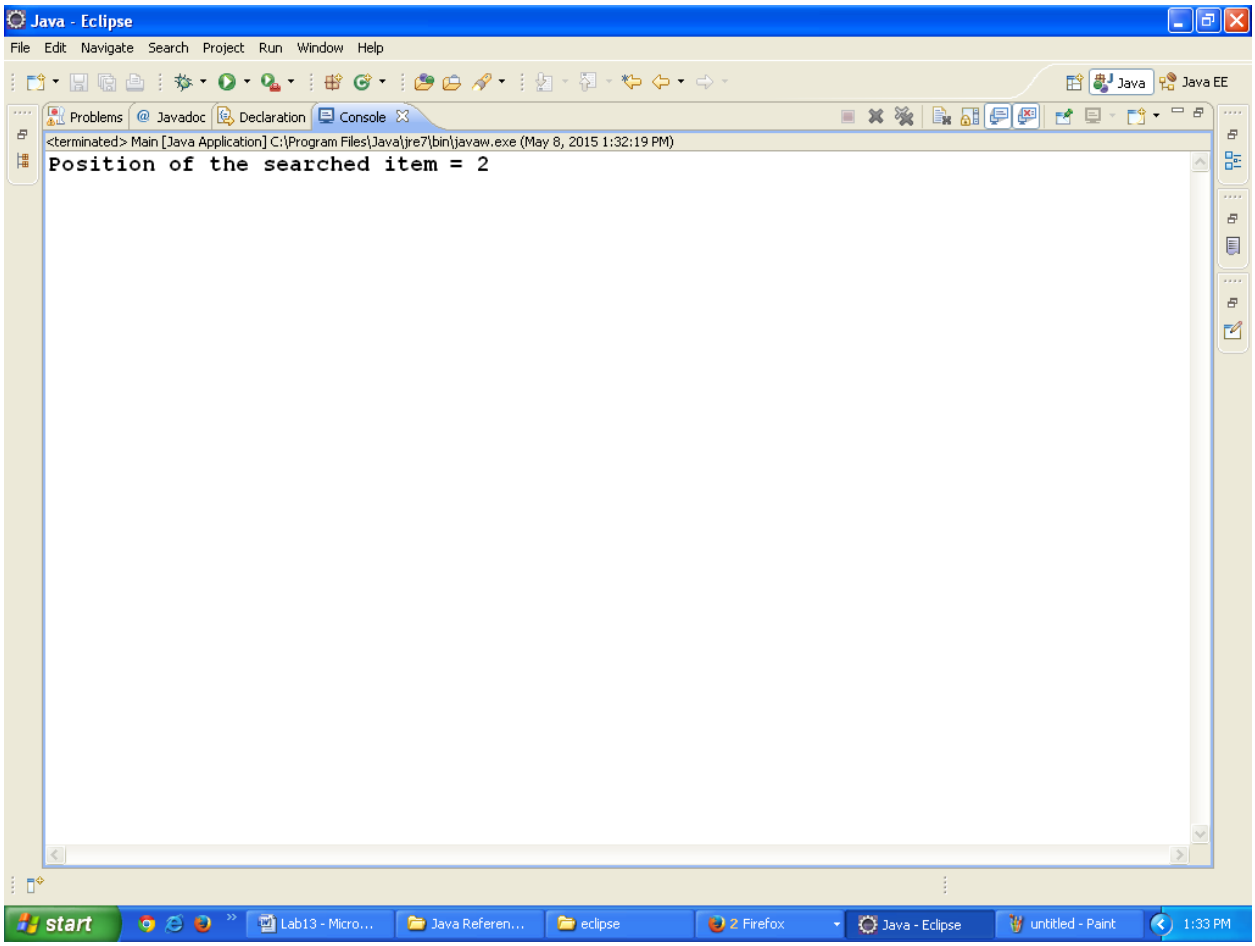
Comparable

Comparator

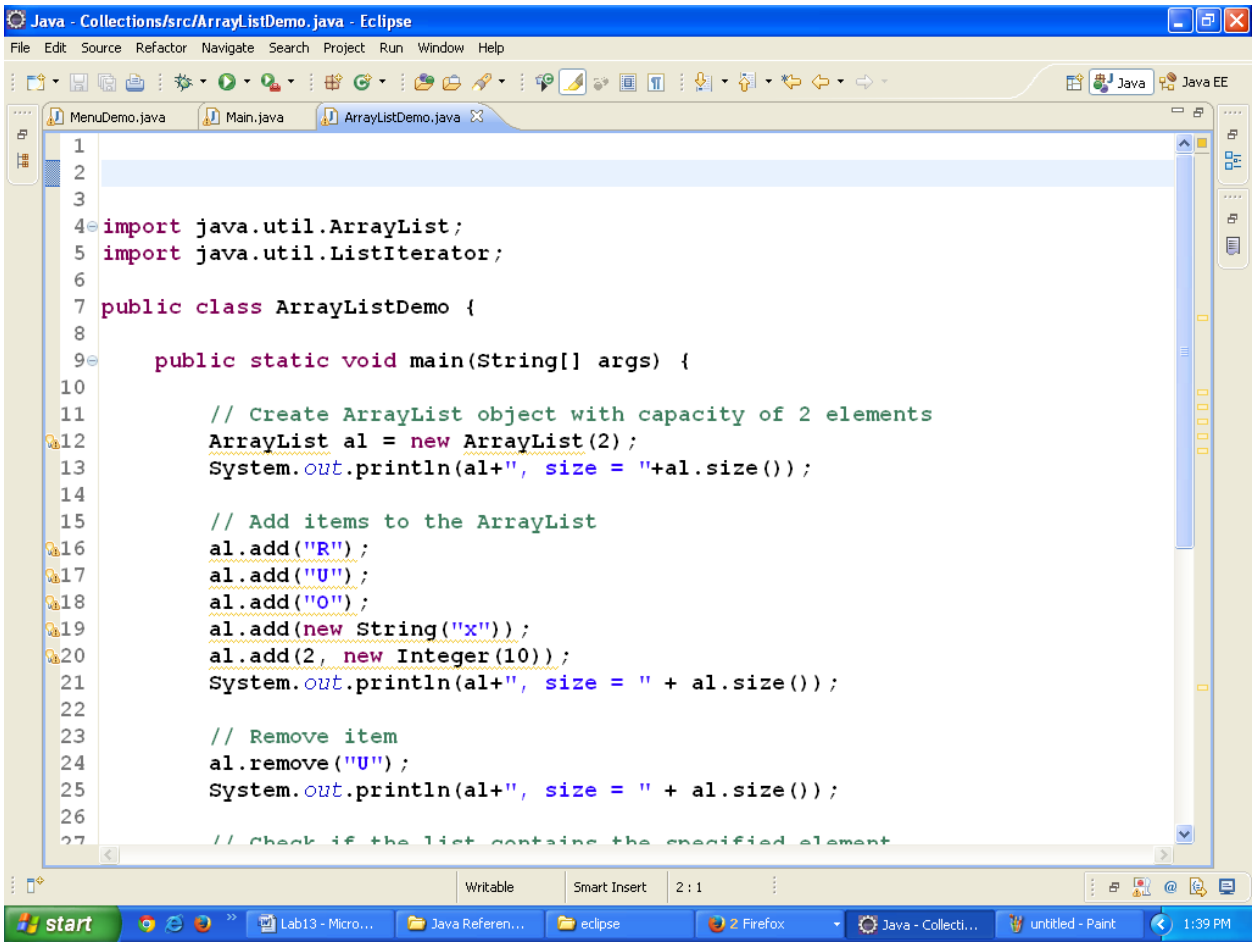
Demo 1:List



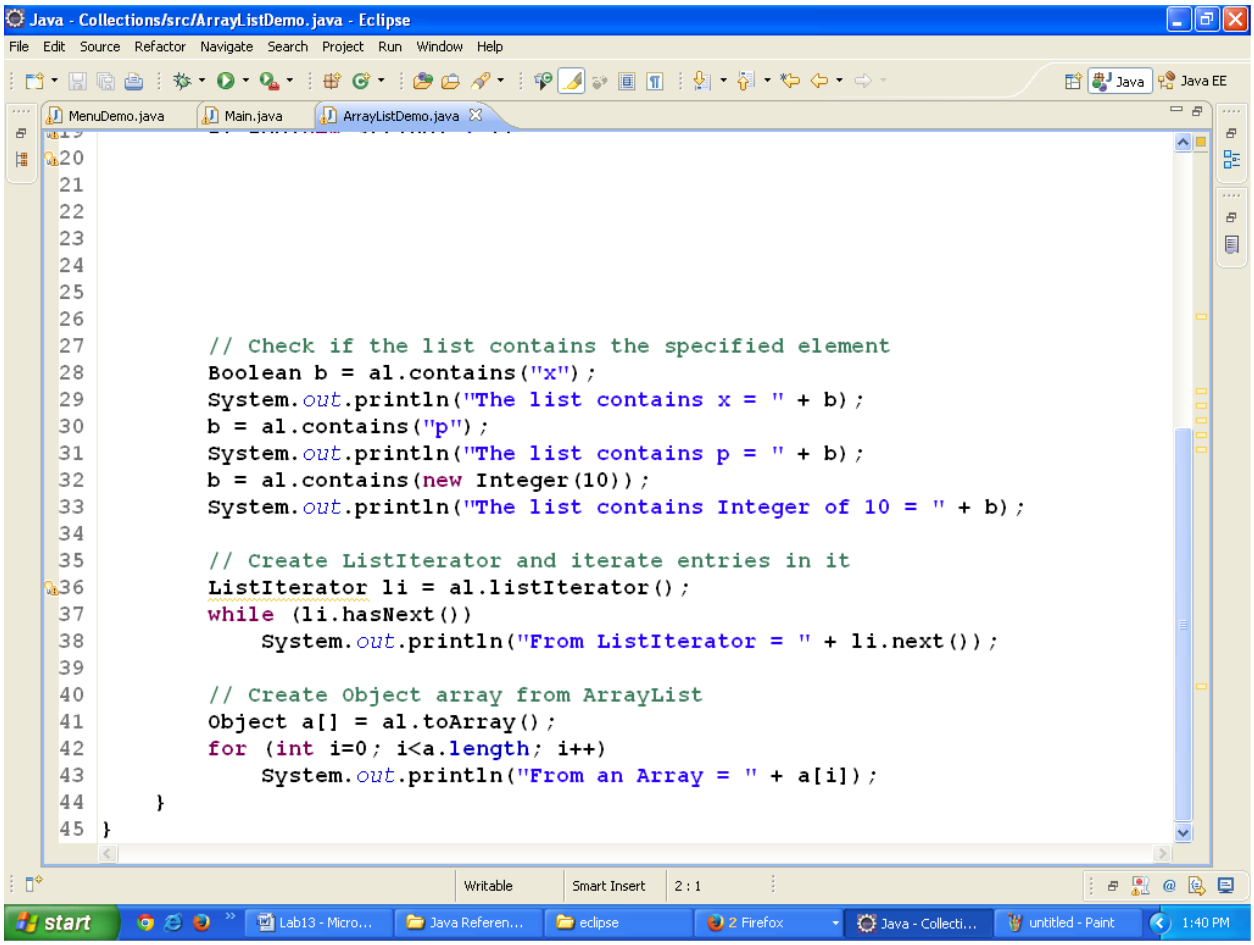
Run as -> Java Application



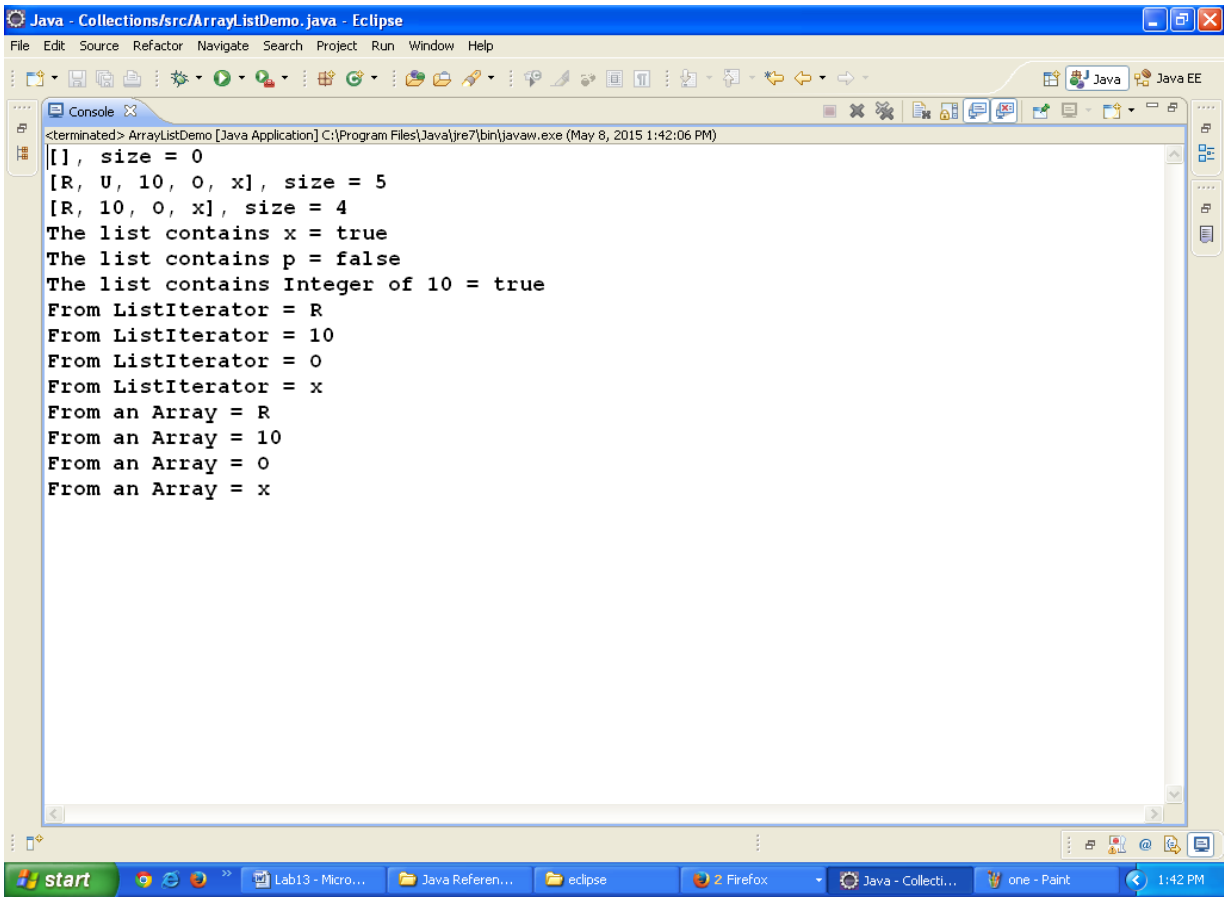
Demo 2: ArrayListDemo


Java - Collections/src/ArrayListDemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
MenuDemo.java
Main.java
ArrayListDemo.java
import java.util.ArrayList;
import java.util.ListIterator;
public class ArrayListDemo {
public static void main(String[] args) {
// Create ArrayList object with capacity of 2 elements
ArrayList al = new ArrayList(2);
System.out.println(al+", size = "+al.size());
// Add items to the ArrayList
al.add("R");
al.add("U");
al.add("O");
al.add(new String("x"));
al.add(2, new Integer(10));
System.out.println(al+", size = " + al.size());
// Remove item
al.remove("U");
System.out.println(al+", size = " + al.size());
Writable
Smart Insert
2 : 1
start
Lab13 - Micro...
Java Referen...
eclipse
2 Firefox
Java - Collecti...
untitled - Paint
1:39 PM

Continued…



Run as -> Java Application



Assignments To Solve :

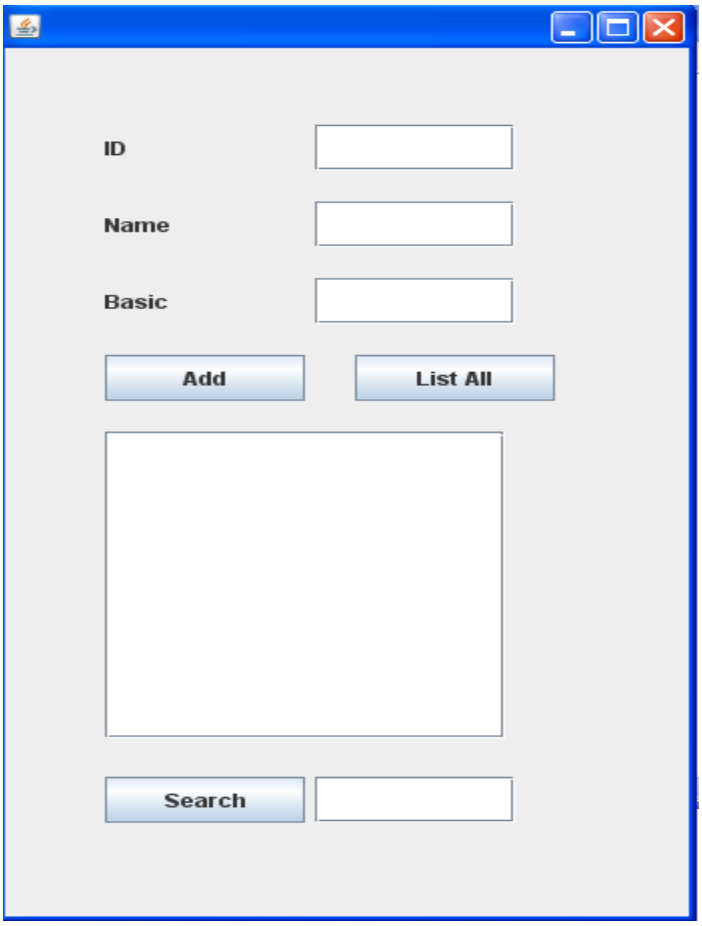
1. Write a TestBoxing class to check auto boxing and auto unboxing concepts

2. Write a program to test add(),get(),size(),isEmplty(),iterator(),remove() functions with basic collection APIs(List,Set)

2. Write a Java program to create a new array list, add some colors (string) and print out the collection(use iterator() and foreach() for printing elements).

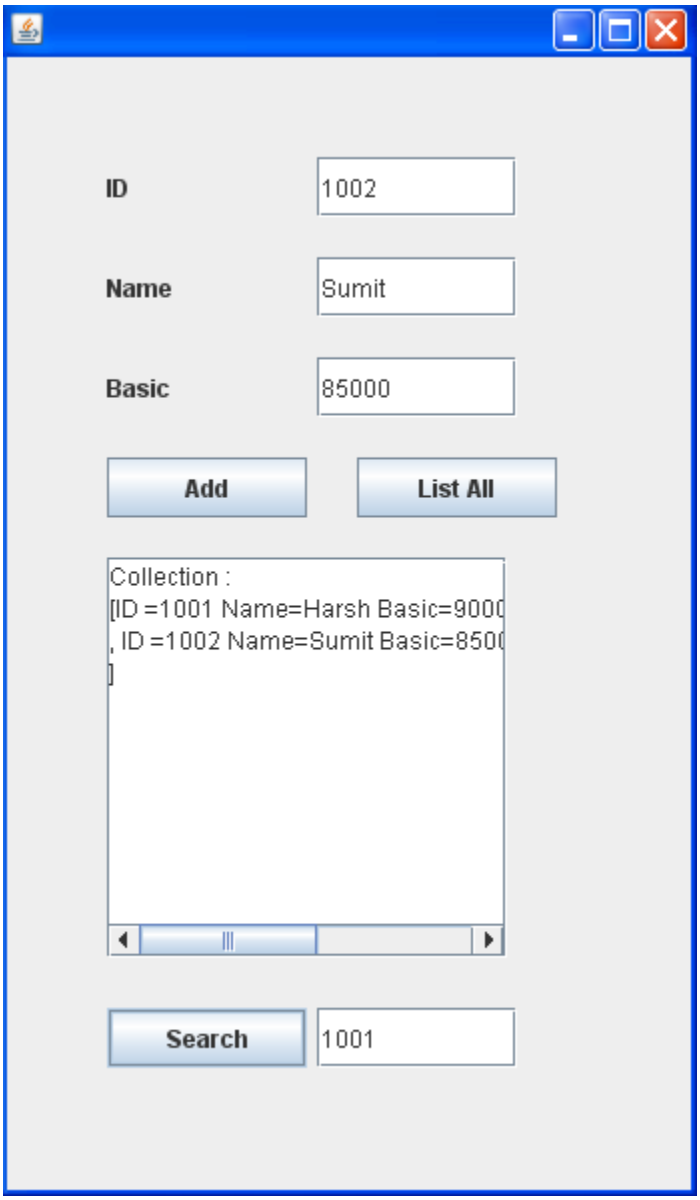
3.Write a Menu Driven program to do following
To Add elements into Arraylist
To retrieve an element (at a specified index) from a given array list
Toinsert an element into the array list at the first position
To update specific array element by given element.
To remove element from given position
To Search element in an arraylist
To reverse element of an arraylist
To sort element of an arraylist

4. WAP to generate the following GUI



Handle event for Add, List All and Search Buttons
Event for Add:
Retrieve the elements from the textfields, create an Object of Employee and add the Objects to Vector or ArrayList.

Event for List All
Display All Objects in TextArea
Event for Search
Iterate through the collection to find employee object for a particular id entered and display the details of that Object in TextArea..Or else print message Object not found for given id..



5. Create a Product class with Product Id & Product Name. Write a program to accept information of 10 products and store that in HashTable. Search a particular product in the Hash Table. Remove a particular product id and product name from the Hash Table. The product list is as follows:

Product Id	Product Name
P001	Maruti800
P002	MarutiZen
P003	MarutiEsteem

6. Write a Java program to Store the <telephone_no,Name_of_Customer> and sort the data by key of descending order

7. Write a Java program which take every entry as name and age of student,and save it as below specified way

Example :
Input :
Enter name and age of 1th :
Pravin 25
Enter name and age of 1th :
Prmod 20
Enter name and age of 1th :
Soham 25
Enter name and age of 1th :
Pravin 5

O/P:
Pravin 25,5
Prmod 20
Soham 25

8. Write a java program to save 5 customer object into TreeSet and print it using descending order. (use comparable or comparator)

Lab-14 Assignments

MultiThreading

Thread class and their methods

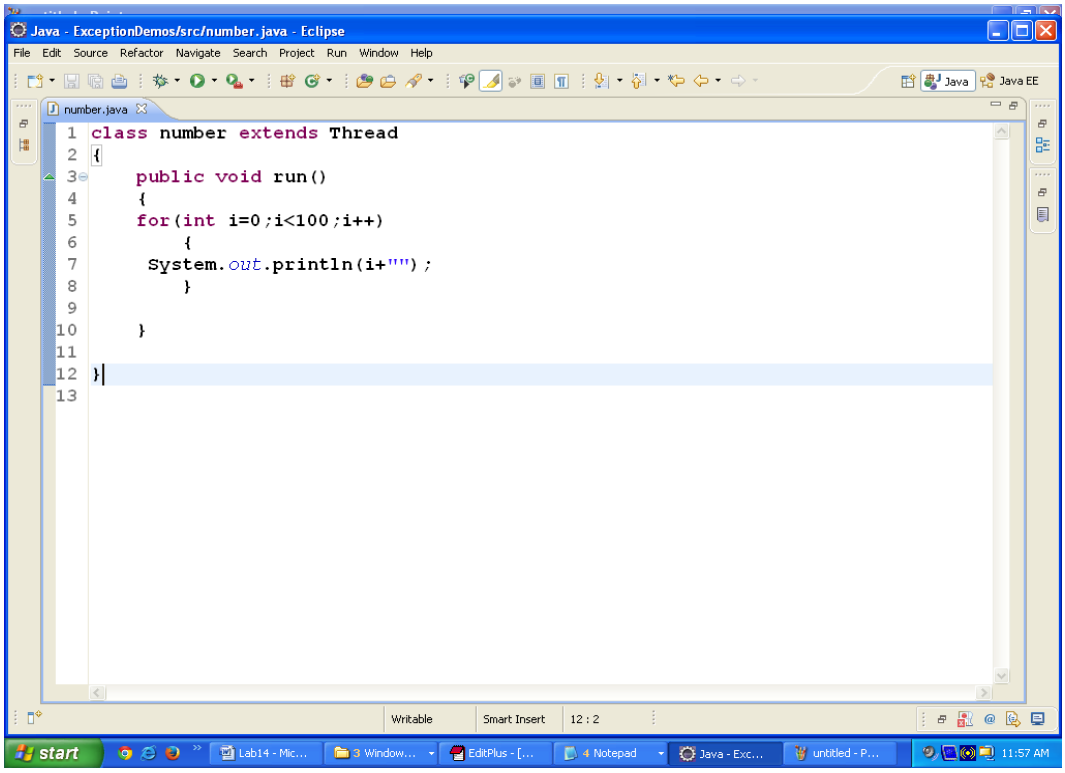
Runnable Interface

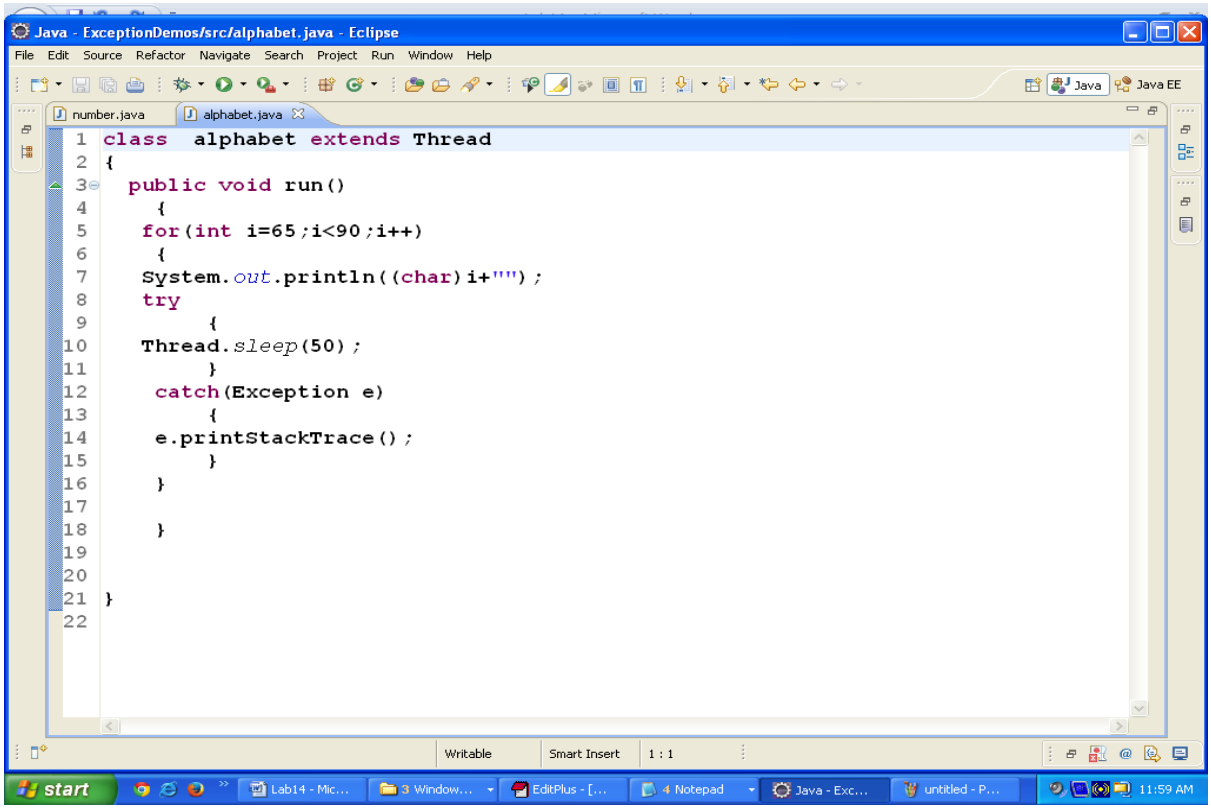
Synchronization

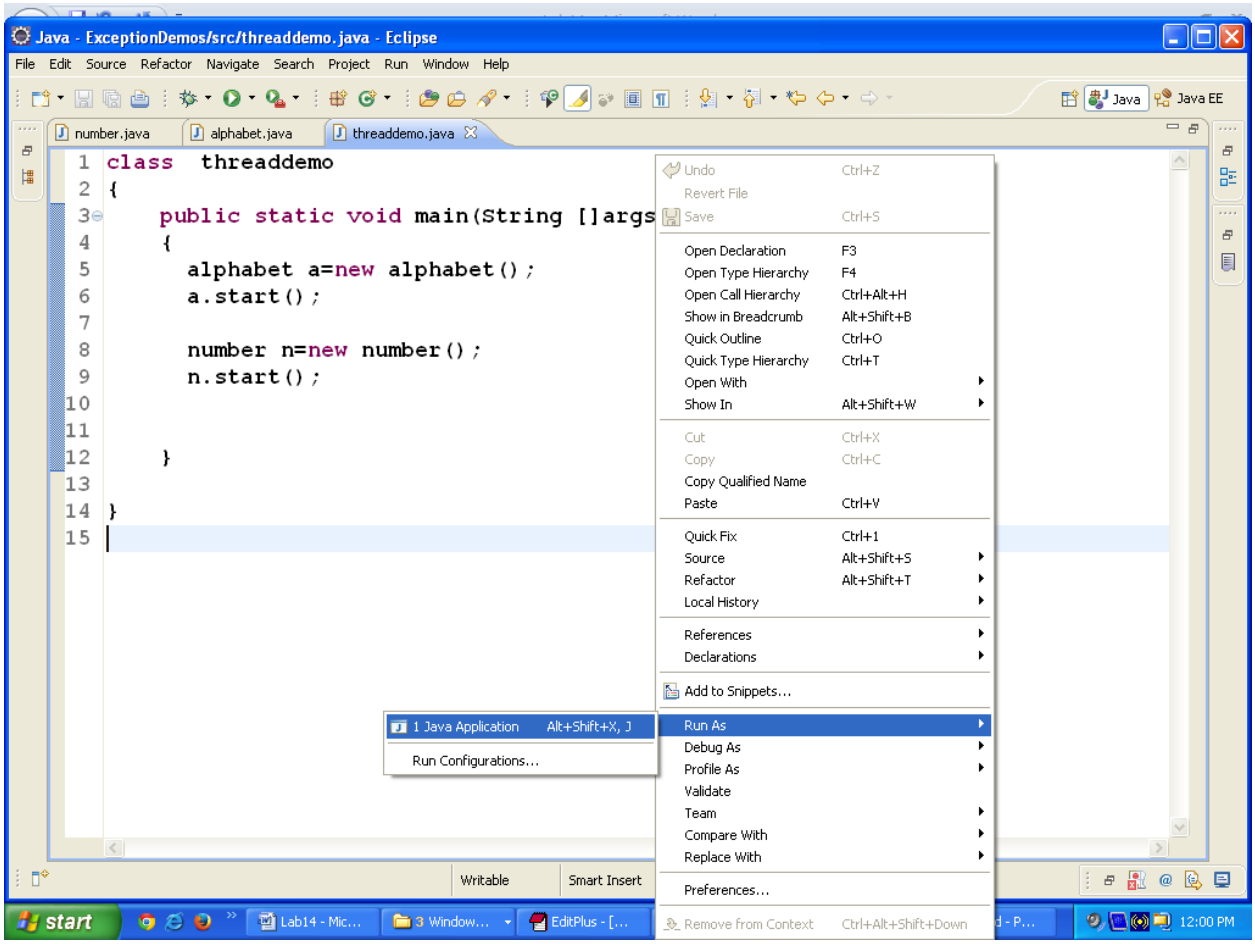
Interthread communication

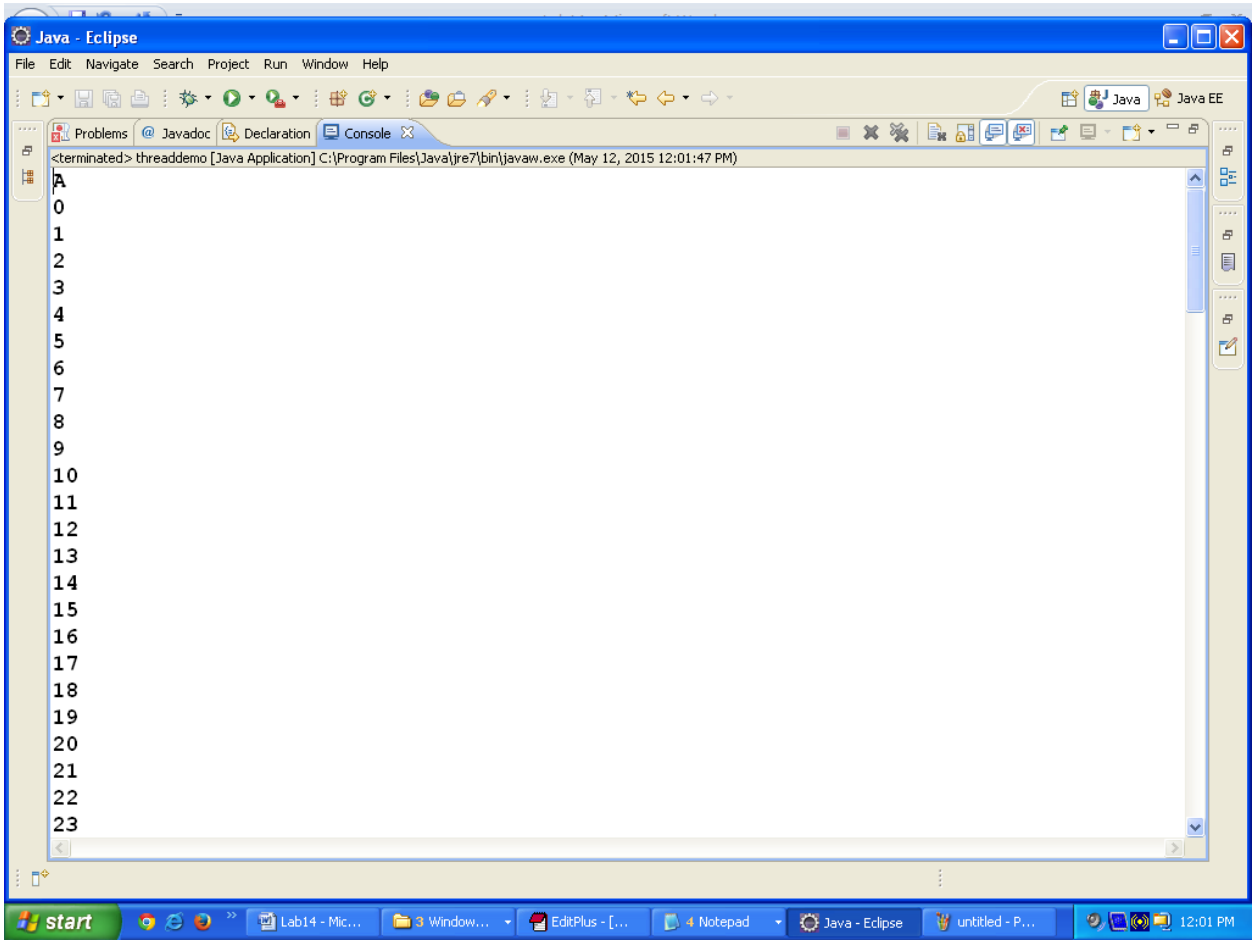
Lock api

Demo 1: ThreadDemo




Java - ExceptionDemos/src/alphabet.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
number.java
alphabet.java
1 class alphabet extends Thread
2 {
3 public void run()
4 {
5 for(int i=65;i<90;i++)
6 {
7 System.out.println((char)i+"");
8 try
9 {
10 Thread.sleep(50);
11 }
12 catch(Exception e)
13 {
14 e.printStackTrace();
15 }
16 }
17
18 }
19
20
21 }
22
Writable
Smart Insert
1 : 1
start
Lab14 - Mic...
3 Window...
EditPlus - [...
4 Notepad
Java - Exc...
untitled - P...
11:59 AM


Java - ExceptionDemos/src/threaddemo.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
number.java
alphabet.java
threaddemo.java
class threaddemo
{
public static void main(String []args
{
alphabet a=new alphabet();
a.start();
number n=new number();
n.start();
}
}
Undo Ctrl+Z
Revert File
Save Ctrl+S
Open Declaration F3
Open Type Hierarchy F4
Open Call Hierarchy Ctrl+Alt+H
Show in Breadcrumb Alt+Shift+B
Quick Outline Ctrl+O
Quick Type Hierarchy Ctrl+T
Open With
Show In Alt+Shift+W
Cut Ctrl+X
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Quick Fix Ctrl+1
Source Alt+Shift+S
Refactor Alt+Shift+T
Local History
References
Declarations
Add to Snippets...
Run As
Debug As
Profile As
Validate
Team
Compare With
Replace With
Preferences...
Remove from Context Ctrl+Alt+Shift+Down
1 Java Application Alt+Shift+X, J
Run Configurations...
Writable
Smart Insert
start
12:00 PM



Assignments To Solve

1. Write a Java program to create Thread and practice below mehtods.

getName(): It is used for Obtaining a thread's name

getPriority(): Obtain a thread's priority

isAlive(): Determine if a thread is still running

join(): Wait for a thread to terminate

run(): Entry point for the thread

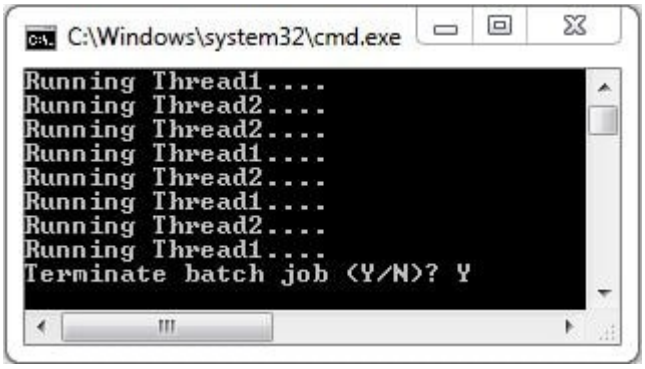
sleep(): suspend a thread for a period of time

start(): start a thread by calling its run() method

2. Write a Java program to create multiple threads using Runnabel interface and print currently running thread.

3. Write a program that creates 2 threads (Thread1,Thread2)- each displaying a message (Pass the message as a parameter to the constructor). The threads should display the messages continuously till the user presses ctrl+c. (use Thread.sleep(400))

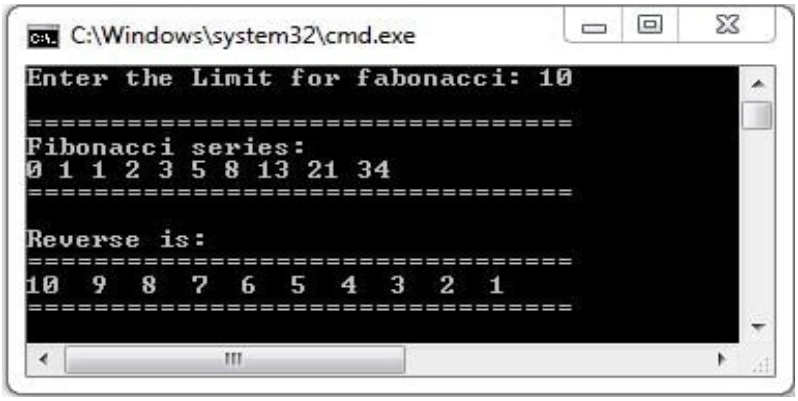
output :



4. Write a JAVA program which will generate the threads:

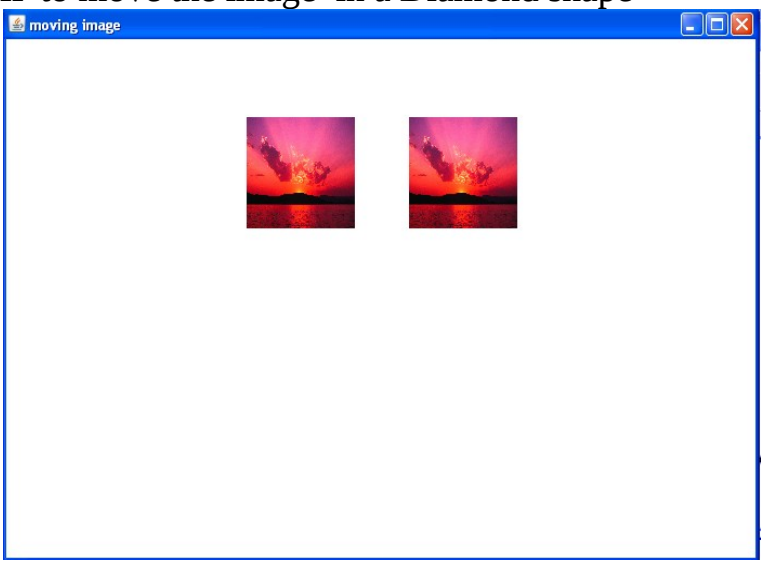
- To display 10 terms of Fibonacci series. (class Fibonacci extends Thread)
- To display 1 to 10 in reverse order. (class Reverse extends Thread)

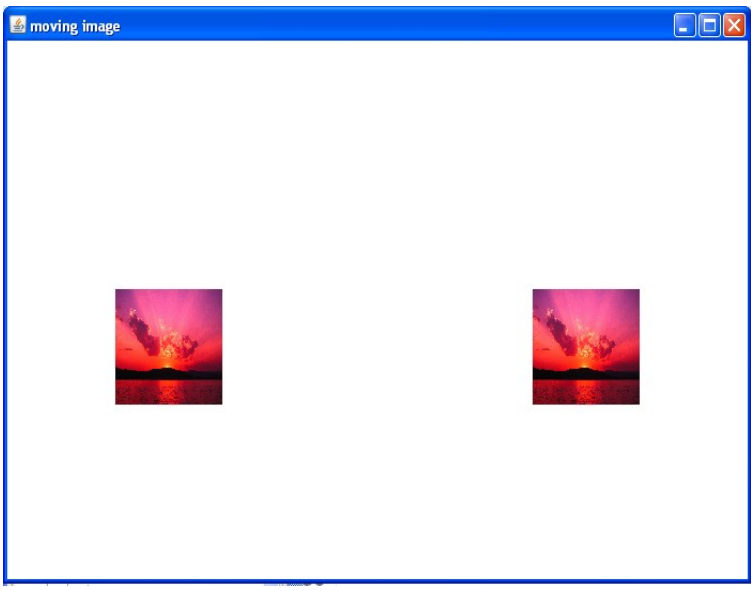
Output :

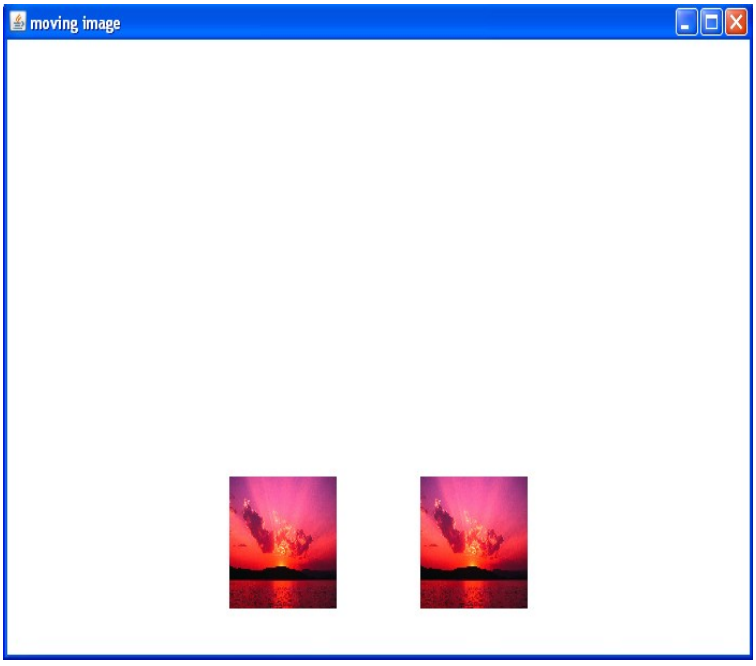


5. Writer a program to move the String ProFound.. As the String Moves the background colourshould change.

6. WAP to move the image in a Diamond shape







7. Write a Java program using Synchronized Threads, which demonstrates Producer Consumer concept.

8. Create a Banking Application simulating concurrent deposit and withdrawal. Use synchronized method for transaction.

Account Class

Attributes: acno & balance

Methods: getter setter methods and toString()

Method signature

public synchronized int withdraw(Account acc, int amount)
{….}

public synchronized int deposit(Account acc, int amount)
{…..}

9. Create a Banking Application simulating concurrent deposit and withdrawal. Use synchronized block for transaction

Method signature

public int withdraw (Account acc, int amount)
{

```
synchronized (this)
{…..}
}

public int deposit(Account account, int amount)
{
  synchronized {….}
}
```

10. WAP to move three balls (Use wait(), notify() and notifyAll() of Object class)

Lab-15 Assignments

1.JDBC Api

Connection
Statement
PreparedStatement
ResultSet

Assignments To Solve:

1. WAP perform CRUD operation on Student(rollno,name,marks,address) table using JDBC APIs .(use Statement)

2. WAP perform CRUD operation on Book(id,name,author,price,category) table using JDBC APIs .(use PreparedStatement)

Advanced JDBC

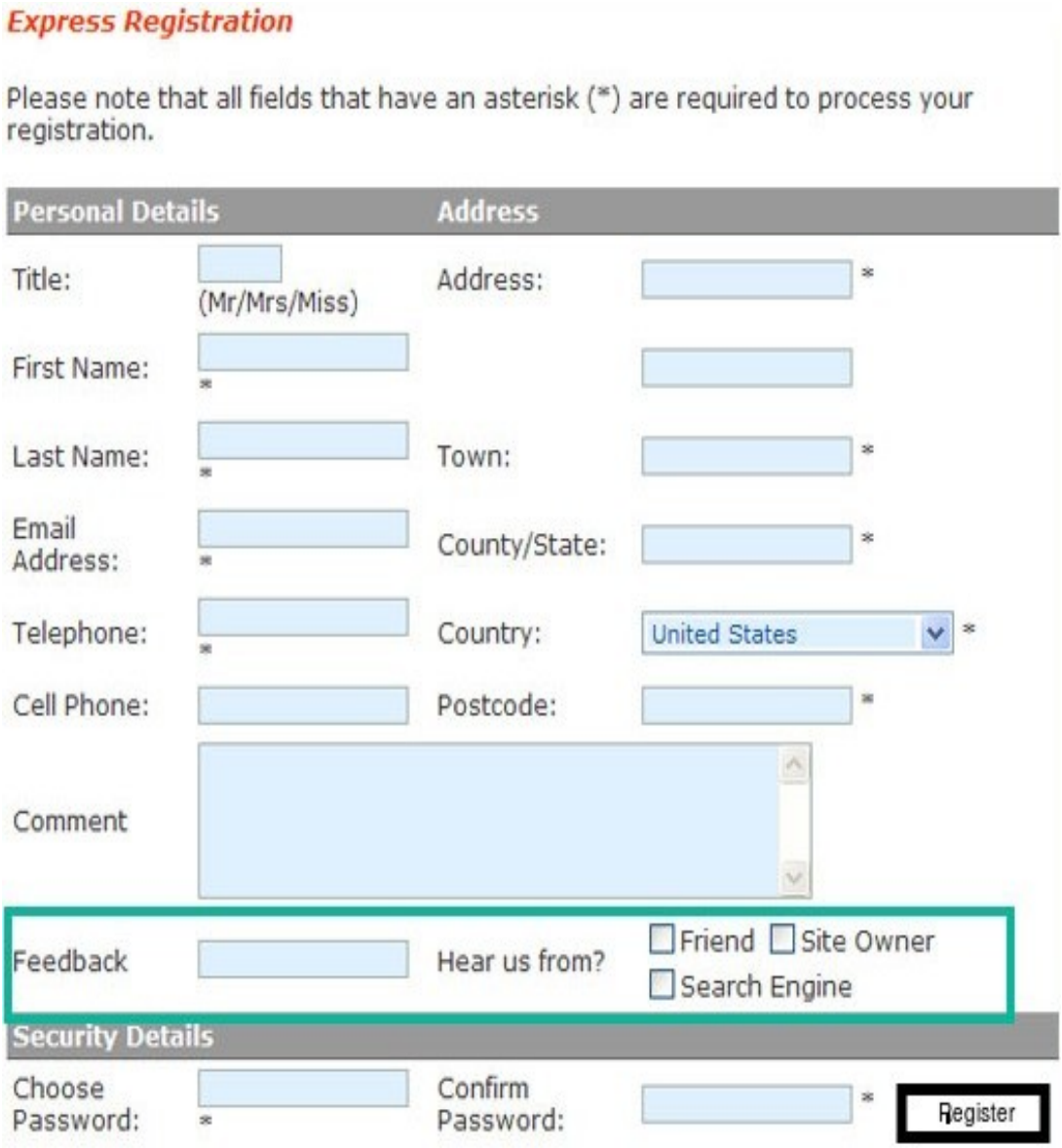
Assignment to solve:

1. Create procedures AddnewCustomer() and SearchNameById() in mysql database
   call above procedures by jdbc's Callable statement
2. Write a java application to test Scrollable and nonScrollable resultsets, with Book entity

3. Write a java application to to print the metadata above ResultSet as well as Database (use DatabaseMetadata,ResultSetMetaData Api)

HTML & CSS

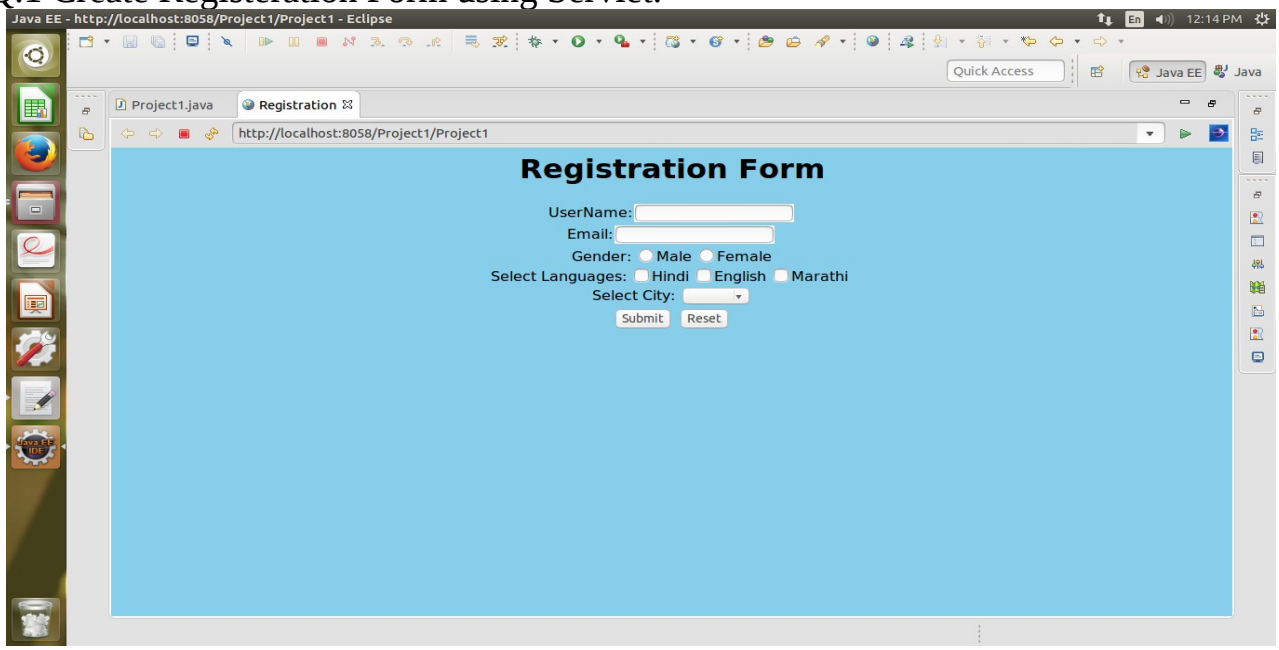
Q. Design the following registration form using HTML and CSS.



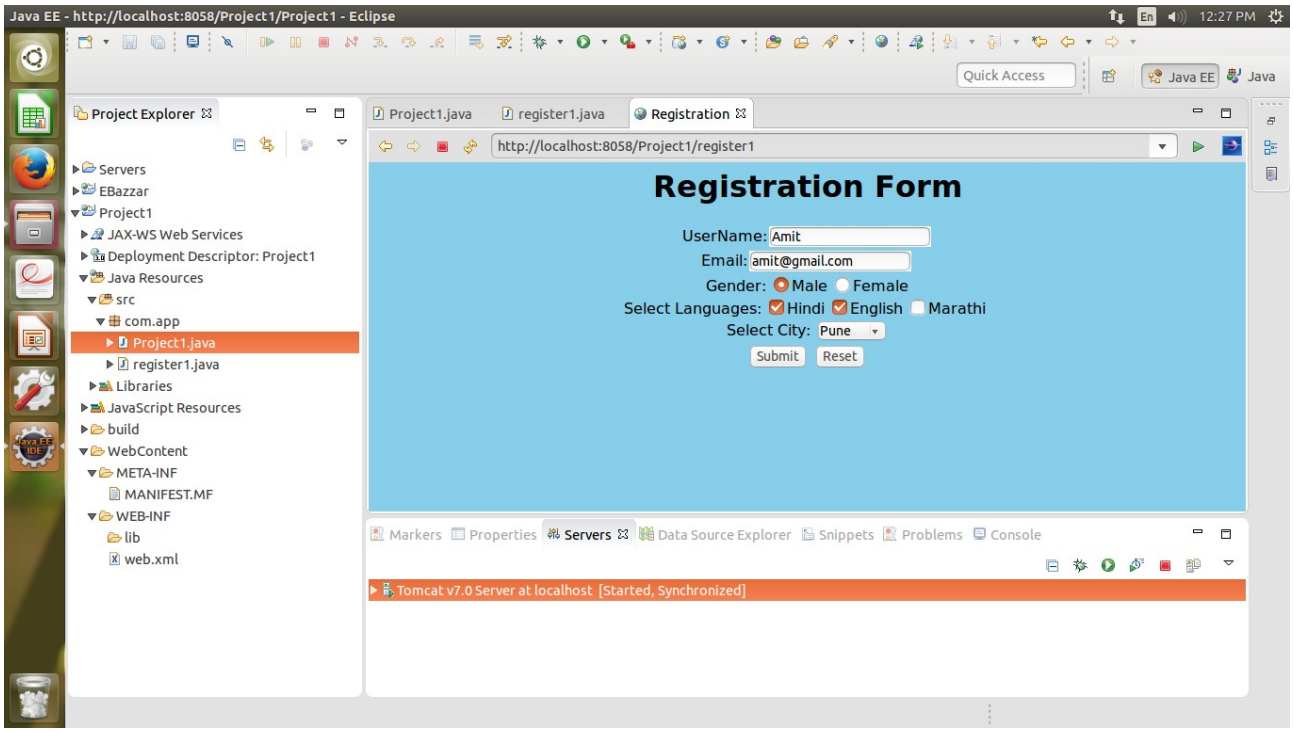
Q. Validate above required field by using form validation before click of register button.

Servlet Assignment

Q.1 Create Registeration Form using Servlet.



and Display in Next Servlet.



Q.2 Write a servlet that will take a string and show the reverse in the form. (create three servlets,
FirstServlet take text input and submit it to SecondServlet,

The SecondServlet print that input in reverse,it has original button when click,it navigate to ThirdServlet and then

ThirdServlet print original text
)

Feed in some text:
Press "submit" and get the form back with the text reversed:

Clicking "submit" again should give you the original text:
Using Servlet basic API.

Q.3 Store email in web.xml as init-param name value pair. Write a Servlet to extract the initialization parameters inside the init method of your Servlet and display it within your service method.

Q.4 Store database specification into web.xml as context-param name value pair. Write a Servlet to extract the context parameters ,and display it within your service method.

Q5. Develope LoginApp as directed below flow
Login.html[accept username , password and registration link]
ValidateServlet (check username and password is valid or not with database,depends

on that it will redirect to either HomeServlet or Login.html)
When you click on register link it will redirect you to Register.html and takall registration details, after submit it will save all detail using RegisterServlet

Q.6 Create web application to pay online telephone bill using (Servlet,ServletConfig,ServletContext & RequestDispatcher with Database.)
Hint:

Database BSNL
Table PayBill having Cust_No,Cust_Name & Current_Bill

Servlet
Login.html
User enter contact number and login

DisplayServlet.java
Display CustomerName & CurrentBill which fetch from DB using ServletConfig & ServletContext.
Click on "Pay" button.

Pay.java
Display Bill payment successful with DB update.
Else
Back to Login if contact number is incorrect.

Q.7 Display Customer BankDetails like Name,CustomerId Account_no using ServletFilter chaining concept.
CustomFilter—filter customerID
AuthenticateFilter – filter Account_no.

Q.8Develop BookCart Application using Servlet concept in incremental approach.

Scenario:

Cyber Publishing Company is a popular publishing company in India. All the transactions of books happen manually till now. They want to sell books online due to increase No. of customers.

They need to develop an web based application to purchase books online.

Login Id <Login Id >	Category <Category >

Book Id	Book Name	Author Name	Price	Purchase
---	----	----	----	☐
----	---	----	----	☐
	Submit [Hyperlink]	Logout [Hyperlink] [Hyperlink]		Show Cart

- Login ID must be taken from login page.
- Book Category must be either Computer or Engg Books.
- If Computer Books selected then display all the Books with all the details.
- If Engg. Books selected then display all the Books with all the details.
- Click on Logout Hyperlink will display the page with Thanks message and a hyperlink which will move the pointer to the login page again.

3. The customer needs to select the checkbox corresponding to **Purchase** if he is interested in that Book.

- Once the customer has selected all the Books for Purchase he needs to click the **Submit** Button.
- On click of the Hyperlink **Show Cart** the **ShowCartPage** will be called.

.

The screen of **ShowCartPage** is as follows:

Login Id		<Login Id >	
Book Id	Book Name	Price	Amount
-----	------	----	100
------	------	----	200
		Total Amount	300

JSP Assignments

Q.1 Write JSP Script which accepts user name and nick name from user. At first visit, display message “Hello user name” and for next successive requests, display “Hello nick name”.

Use username if visit count is odd and nick name if visit count is even. (use declaration

scripting elements)

Q.2 Write jsp code to use page directives tag's **isErrorPage** ,**errorPage,import** attribute.

Q.3 Write a jsp code to use all jsp implicit object to read and set content with different scope.

Q.4 Using Scripting Element & Directive tag.

Problem Statement 1:

ZBank is new generation bank they are planning to launch their website. They have approached you to create the banking application using jsp. As initial task to test your expertise in jsp programming they have asked you to create a simple welcome screen with a greeting message.

Requirement Description:

The welcome page should be named welcome.jsp. It should have the current date displayed “*dd-MM-yyyy*” format. The page should display a greeting based on the time of the day.

if the time is less than 12.00
Print “Good Morning”
Else Print “Good Evening”

The bank wishes to use 24hr time format.

Also as a matter of initial promo the bank wishes to keep track of the number of users

visiting the site and this value should be displayed to the user.

Develop a jsp page satisfying the above requirements.

The page should look like the screen shot design provided and the following test cases also needs to be executed.

Hints:

- Use the following code for date calculations and use scriptlets

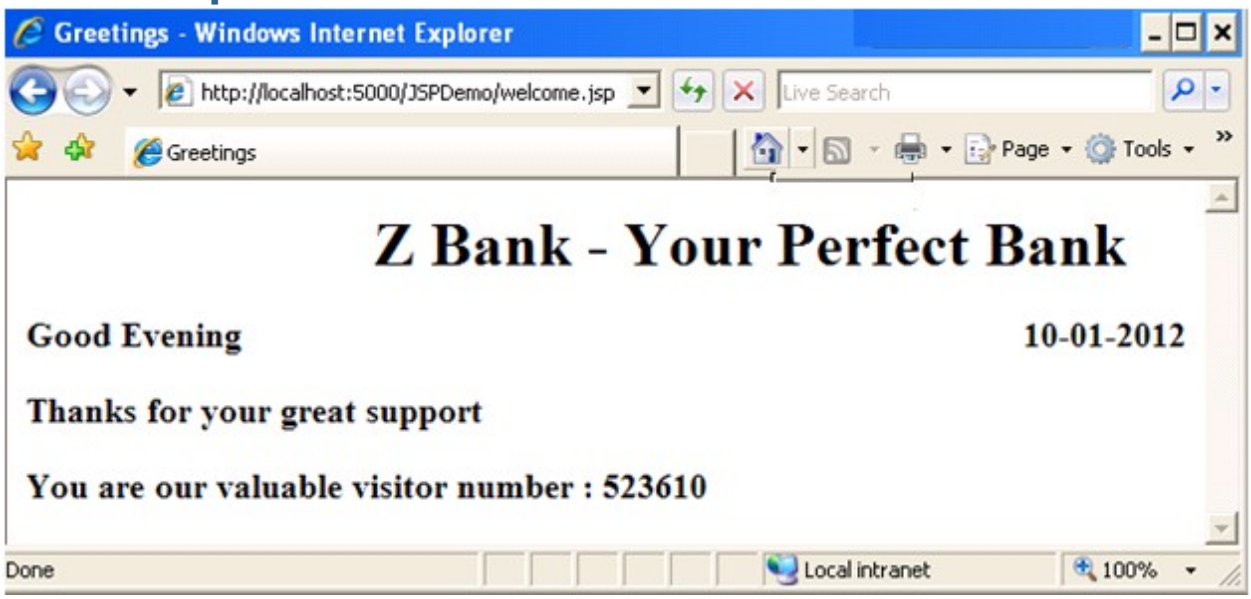
```
java.util.Calendar calendar = new java.util.GregorianCalendar();
java.util.Date date = calendar.getTime();
DateFormat format=new SimpleDateFormat("dd-MM-yyyy");
String dateval=format.format(date);
 int hour = calendar.get(Calendar.HOUR_OF_DAY);
```

- Use declaration to declare page count variable.

Test Cases: Execute the below test cases after development.

11. **Test case #1:** Access the welcome.jsp it should display the current system date and display the appropriate message.
12. **Test case #2:** As you refresh the JSP page the counter needs to be incremented and should increase for each refresh.

Screen Expected



Q.5 Write JSP code which adds the student education details in a database through Java bean.
Also display the student details who have secured first class in their graduation.

Q6. Write JSP code to perform CRUD operation on Product table.
Define links into index.html (add new Product,display products etc)

Q.7 Create custom tag which display User Name in its initial.
For eg Hrithik Rakesh Roshan ---> H.R.Roshan using either tag with body or attribute.

Q.8 Create custom tag which display UserName in uppercse with specified font color (use tag with body and attribute)

4. Ad Java Projcect

Using : Servlet + JSP+DB with MVC & DAO.

Problem Statement:

EBazaar is an online shopping portal where the users can purchase items online. They have approached you to create the application. In first phase they have asked you to create the Search module.

Requirement Description:

Search Module

The company deals with sales of three categories of products

9. Mobiles
10. Laptops
11. Watches

These are the sequence of steps the user does to purchase a product,

Step 1: User logs into the web application using the login.jsp. On successful login the page should be redirected to the home page.

Step 2: The home page (index.jsp) should list down the products Mobiles, Laptops and Watches as links.

Step 3: The user can select one of the above links and select the product they want to add to the cart.

Step 4: After adding the products to the cart the user can purchase the products in the cart.

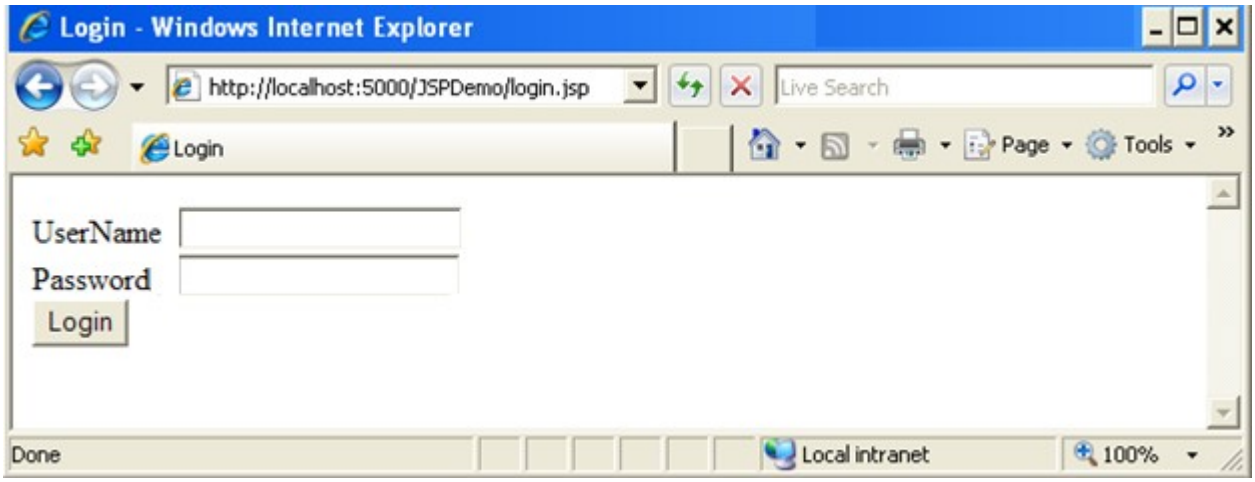
Features:

- Each product category page should provide an option for the user to navigate to the home page or to the cart page.
- The cart page should display all the products the user has added to his cart.
- The cart page should also provide the option to remove items from the cart..
- The cart page should also provide a feature for purchasing the products, clicking the purchase button should purchase all the products added by the user to the cart.
- The company also wishes to keep in track of the number of visitors visiting the site and to display it to all the users in all pages. The visitor count is incremented by for every new login. Say Jack logs into the application the count should show one. If he logs out and logs in again it should be incremented and displayed as 2.

Design:

Design the components as stated below.

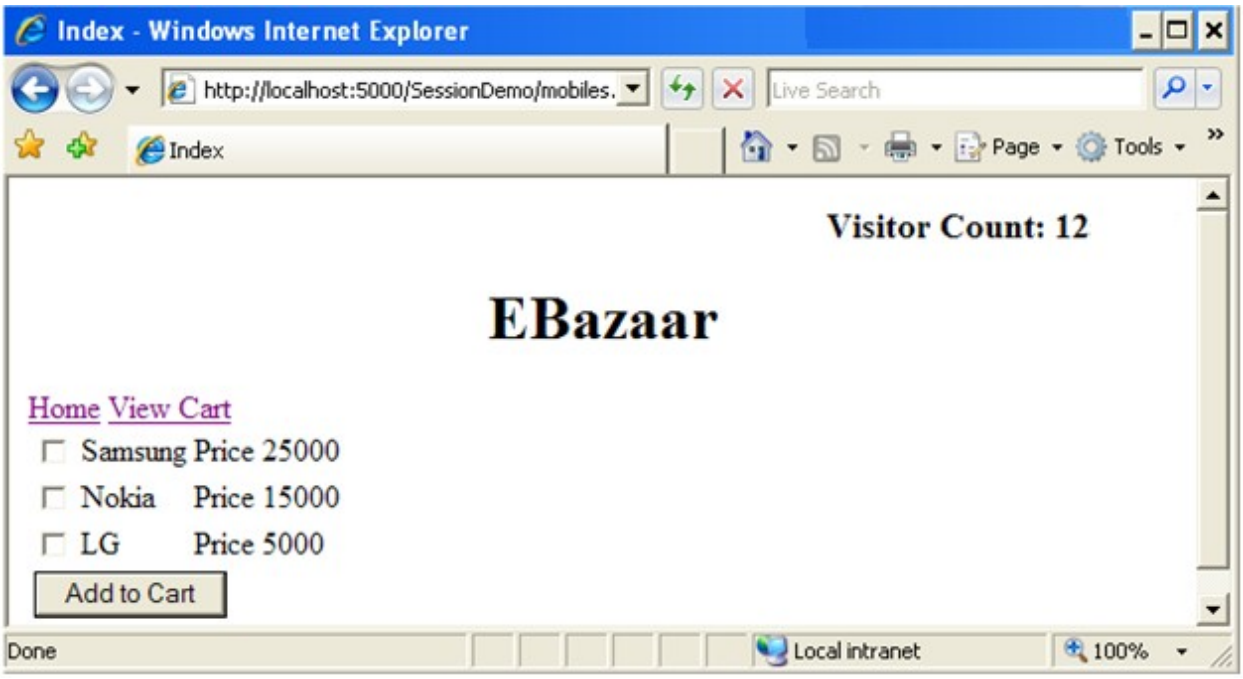
8. **Login.jsp :** Login Page for the user



9. **Index .jsp** : Displays various categories of products

10. **Watch.jsp** : Displays the list of watches available for Sale
11. **Laptop.jsp** : Displays the list of Laptops available for Sale
12. **Mobiles.jsp** : Displays list of mobiles available for Sale

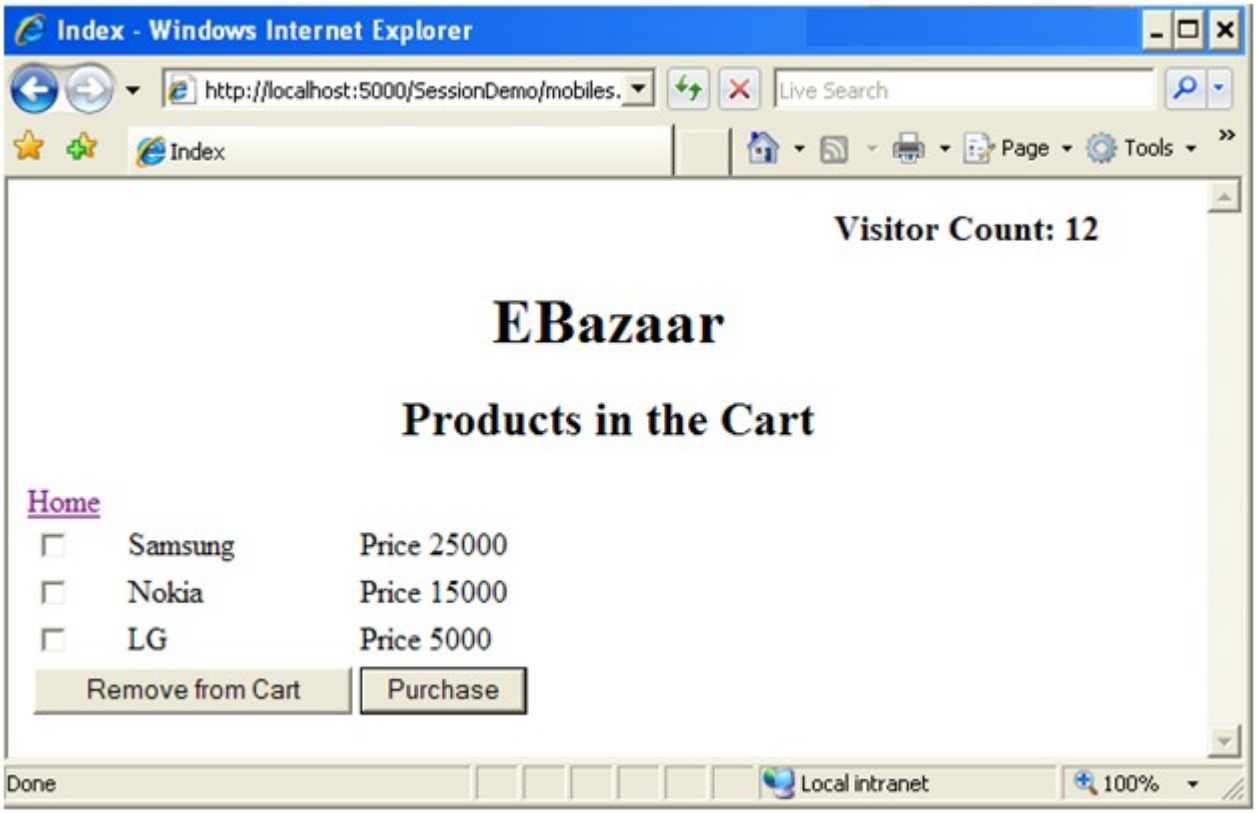
The ***Mobile.jsp*** should display a response similar to the below screen shot,



- User selects the appropriate device he needs to purchase using the check box and clicks Add to cart. This will add the selected products to carts page. Each product category page should provide an option for the user to navigate to the home page or to the cart page. A link "**home**" for the index page and "**View Cart**" for the Cart page should be included as per shown in the screen shot above.

  NOTE: The ***Laptop.jsp*** and ***Watches.jsp*** should display a response HTML similar to the above screen shot and display some laptops and mobiles models respectively

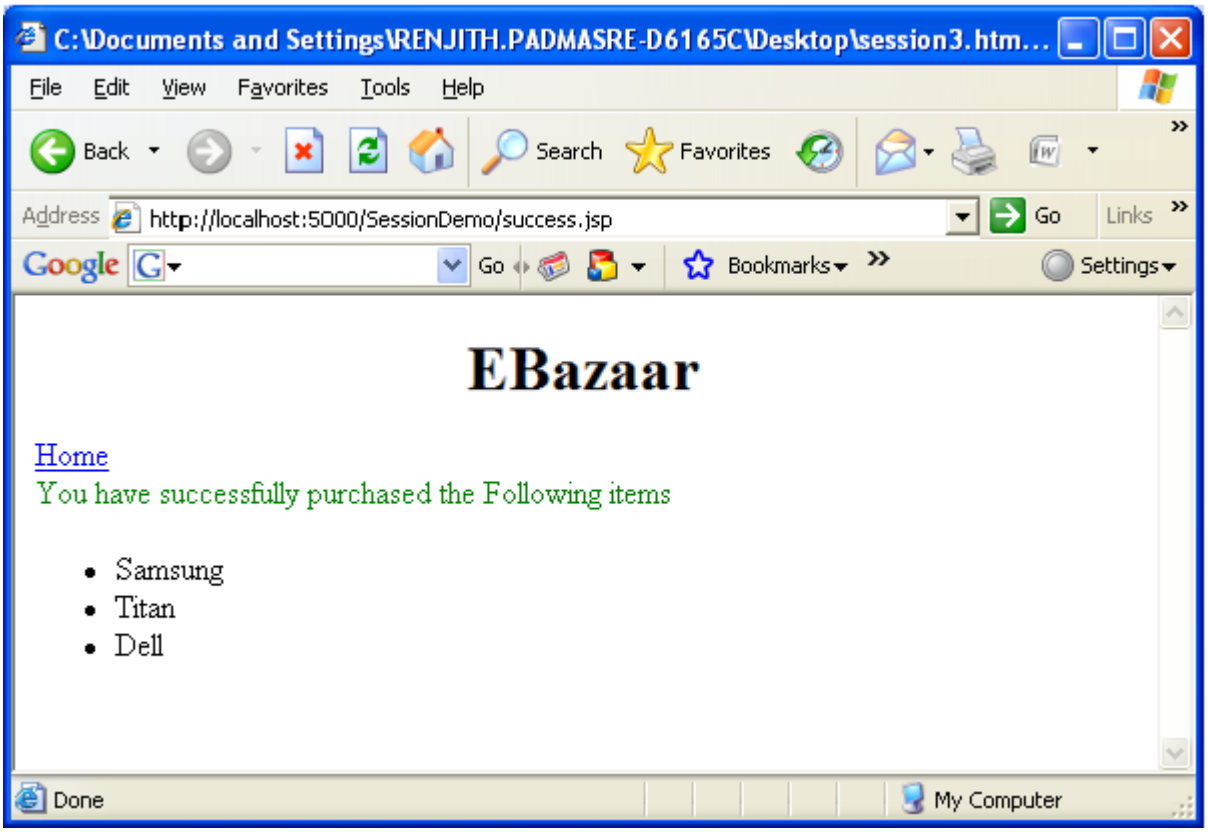
13. **Cart.jsp :** Displays the products which was added by the user in the cart.



The user can either select products and **remove from the cart** or **purchase** the products.

The user can also drop some products from the page by selecting them and clicking “remove from cart”. This should remove the product from the carts page.

On clicking the purchase button the response should be taken to a purchase successful page as illustrated below,



After purchase the products from the cart page should be automatically removed.

Hint:

4. Use HTTP session management technique to manage the products selected by user. Use Sessions add, remove API's to complete this case study.
5. Store the user count in application context to make it available across the application.
6. Use session to check whether the current user visits the index page for the first time in the current session.