

INSTALLATION OF JAVA SOFTWARE: STEPS:

1. Download JDK7.0 from www.java.sun.com/downloads
2. Install JDK by double clicking on setup file
3. Follow the installation wizard.
4. Upon successful installation the default location where java is installed is
C:\Program Files\Java directory.
5. To compile a java class we have to use javac (java compiler)
6. To execute java program we have to use jvm (java virtual machine).
7. To use javac and jvm we have to configure environment variables.

Environment Variables PATH:

- This environment variable is to locate the compiler.
- Right click MYCOMPUTER and find a properties tab, Click on Properties Tab and find Advanced Tab, Click on it to see environment variables.
- In this create a new environment variable called **PATH** and copy the location of java compiler as value to it. Default location:

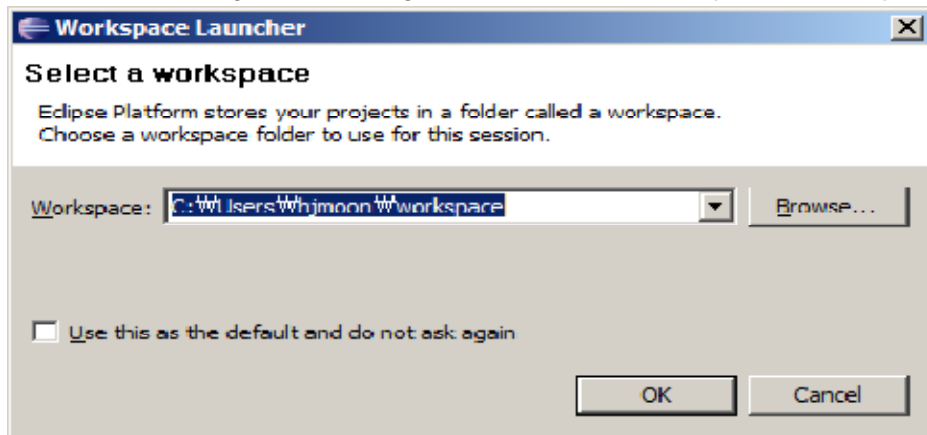
\$PATH= C:\Program Files\Java\Jdk1.6.0\bin;

- **CLASSPATH:** This environment variable will help the jvm to find the runtime environment of java i.e. all predefined classes and interfaces can be located using this variable. The location for runtime environment is JRE folder.

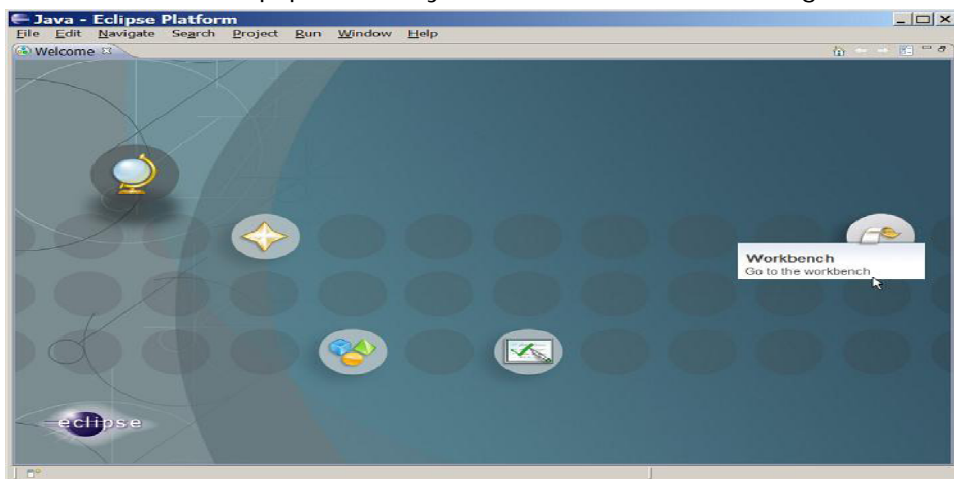
\$CLASSPATH= C:\Program Files\Java\Jre\bin;

ECLIPSE

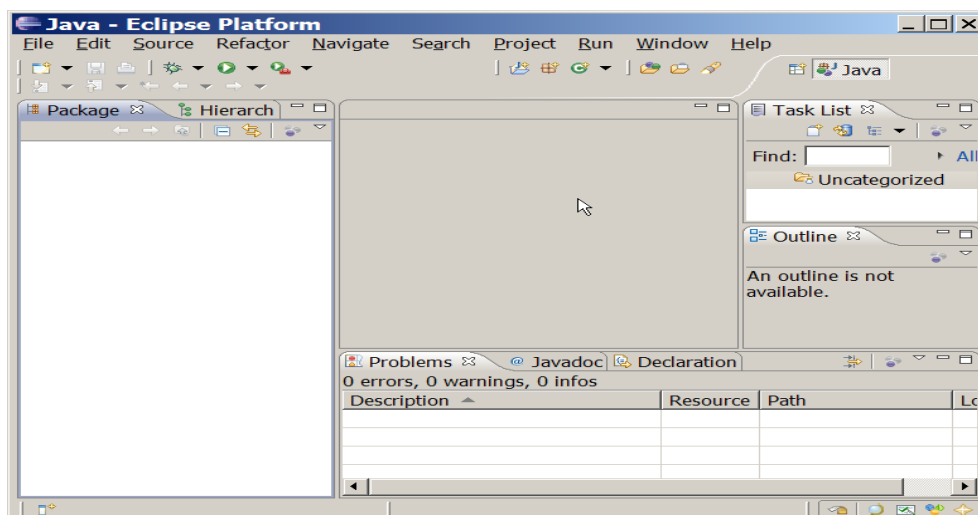
Initializing Eclipse When you develop Java applications in Eclipse, it stores all the created files in a directory called "workspace". When Eclipse is run for the first time, it will ask you where you want the workspace to be placed:



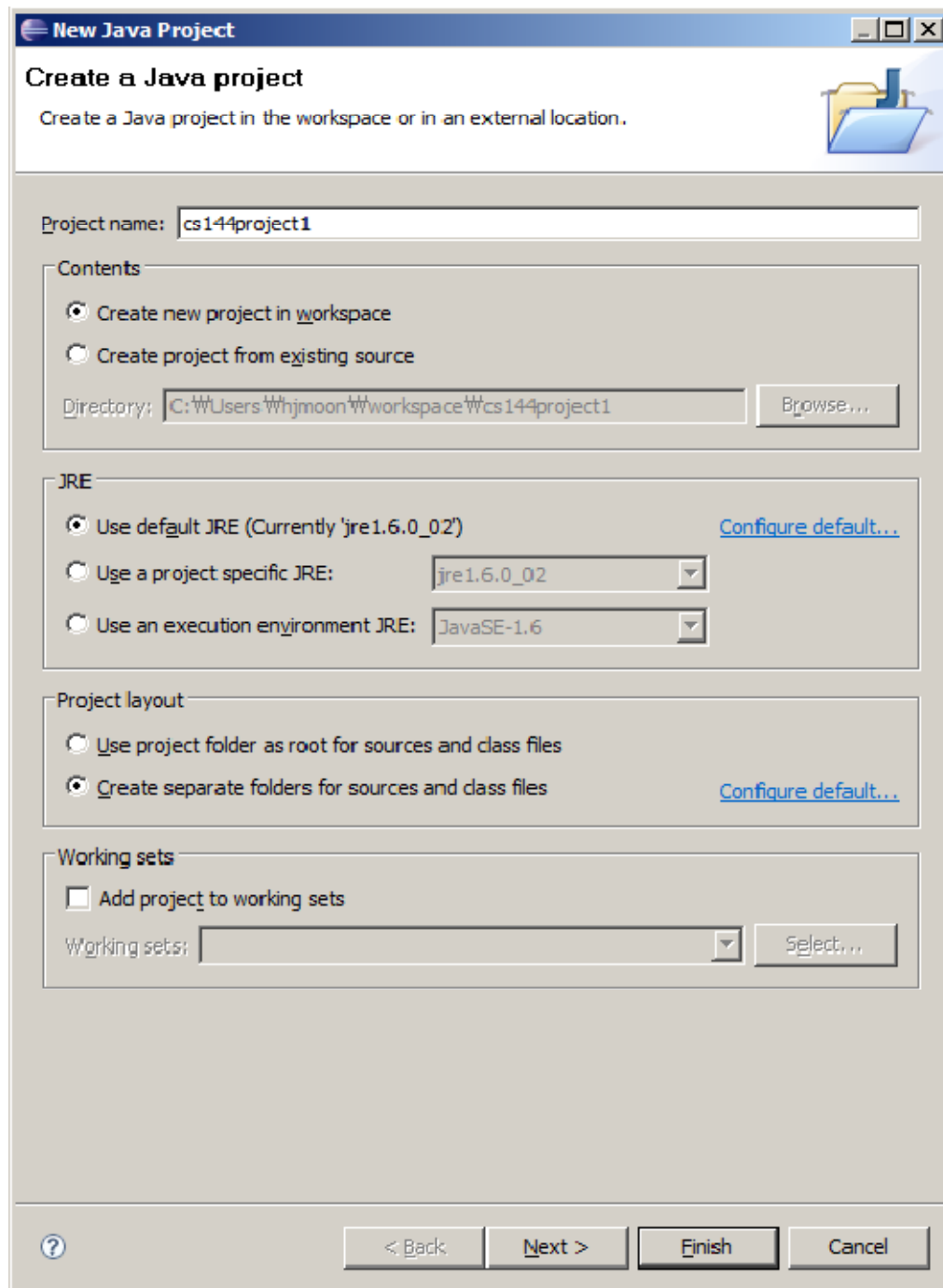
You can just use the default location or specify your preferred location. To avoid getting asked this question every time you start Eclipse, check "Use this as the default and do not ask again" option and press "OK" button. Once Eclipse finishes its startup process, you will see the following welcome window



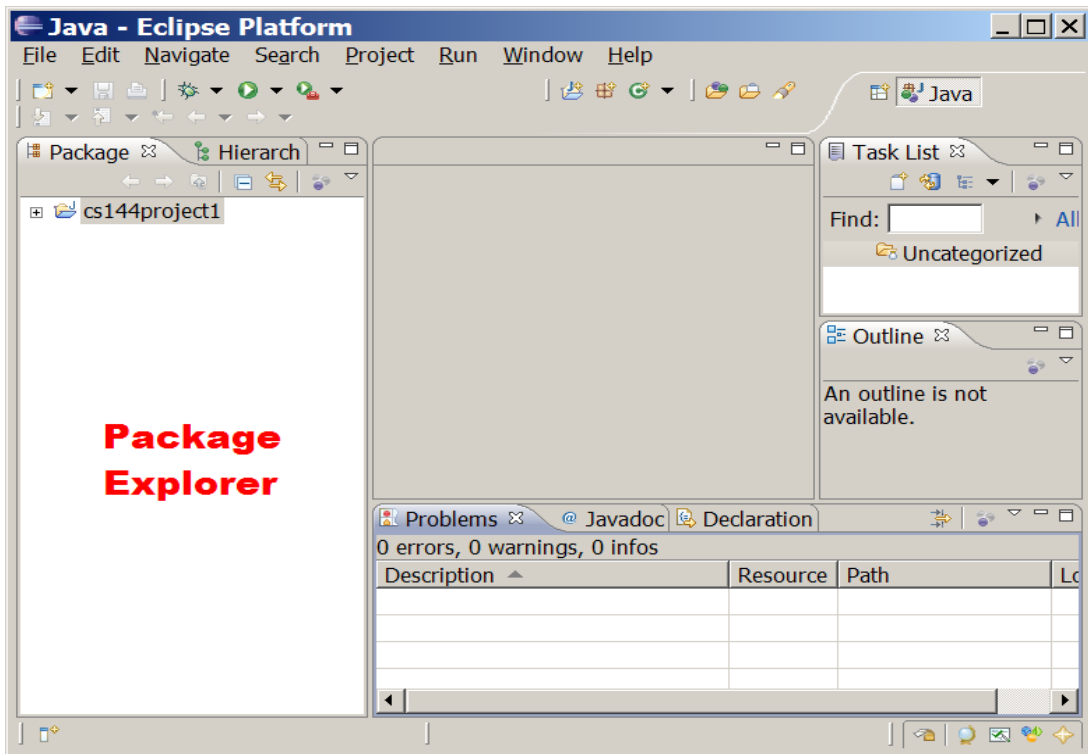
Click the "Workbench" icon on the right, which will lead you to the main Eclipse window:



Creating a Project Now that you've got Eclipse up and running, it's time to create your first Java project. To do this, you'll want to go File -> New -> Java Project. After doing so, you'll see a window like the following:

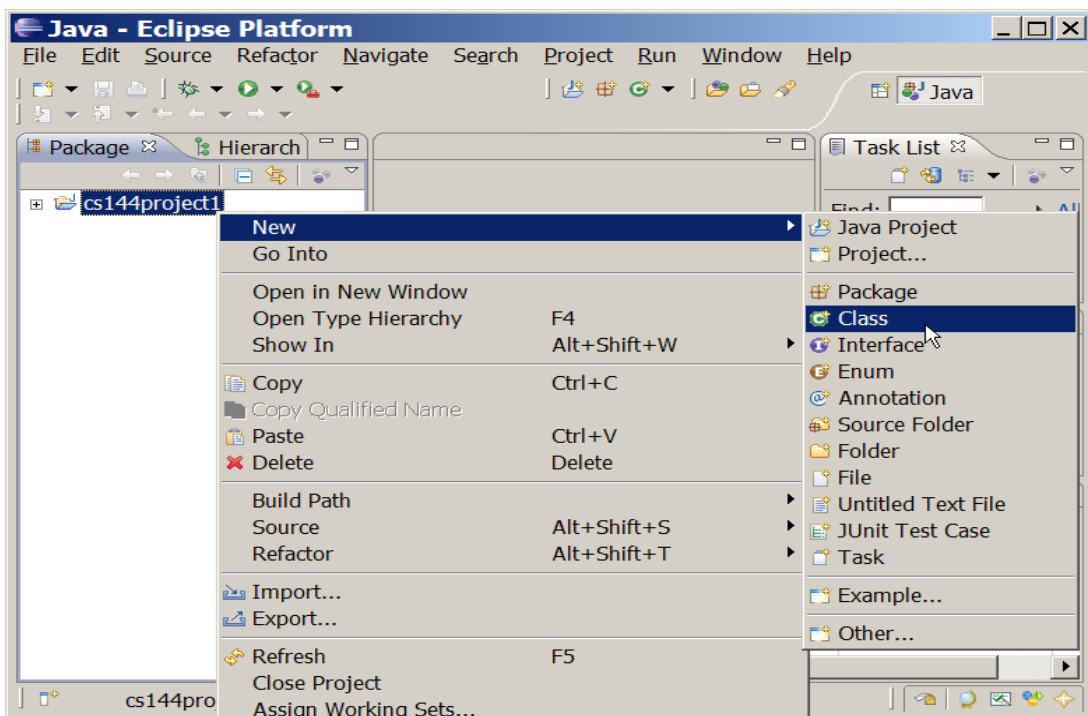


Type your project name (say, cs144project1) in the "Project name" field and click Finish. Then the name of your newly created project will appear on the left side of the Eclipse window (this part of the window is called "Package explorer pane"):

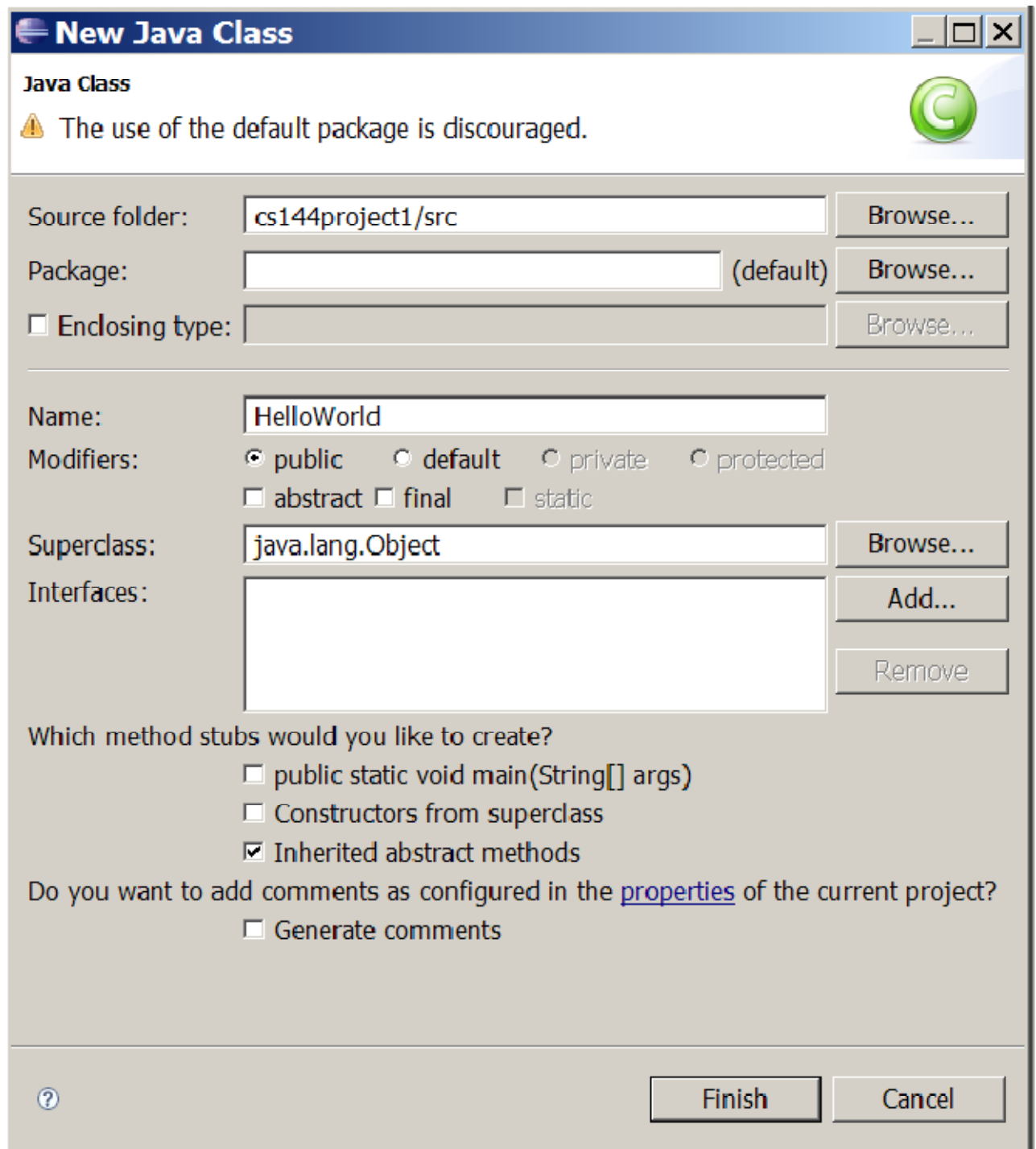


As you create more projects in Eclipse, other project names will appear in the Package explorer pane and you will be able to switch between your projects by clicking the name of a project.

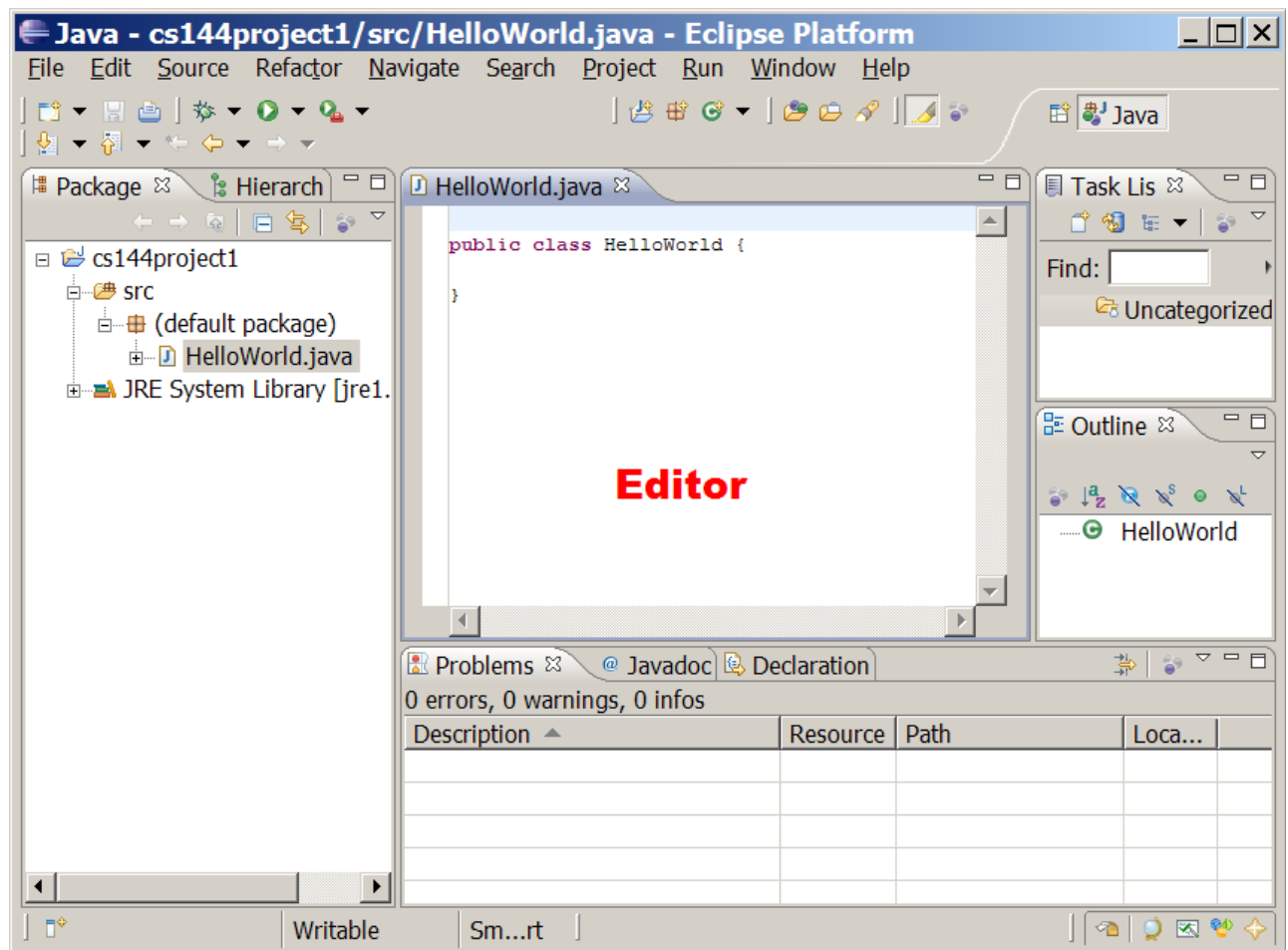
Adding a New File to a Project Now that you've created your first project, you now want to create a new Java file (with .java extension) and add it into your project. To create a new Java file, right click on the name of your project (cs144project1) in the Package explorer pane and select New -> Class as follows:



This command will show you a window that looks like the following:



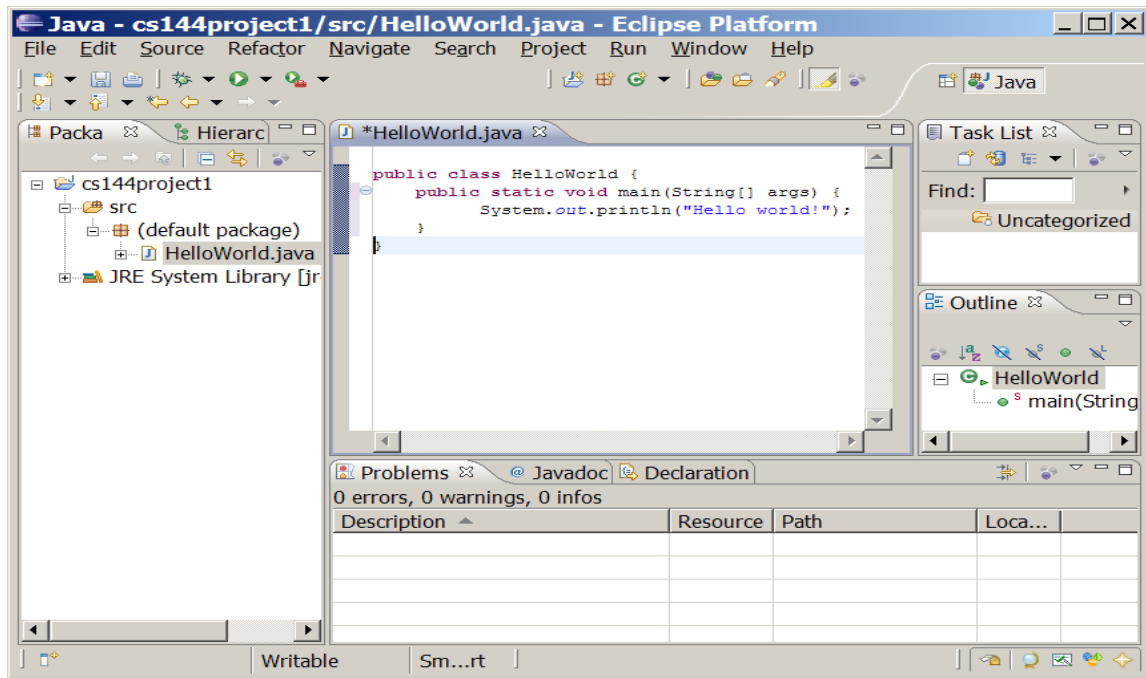
In the "Name: " section provide the name of the file (or the class) you want to create, HelloWorld, and click "Finish" button.



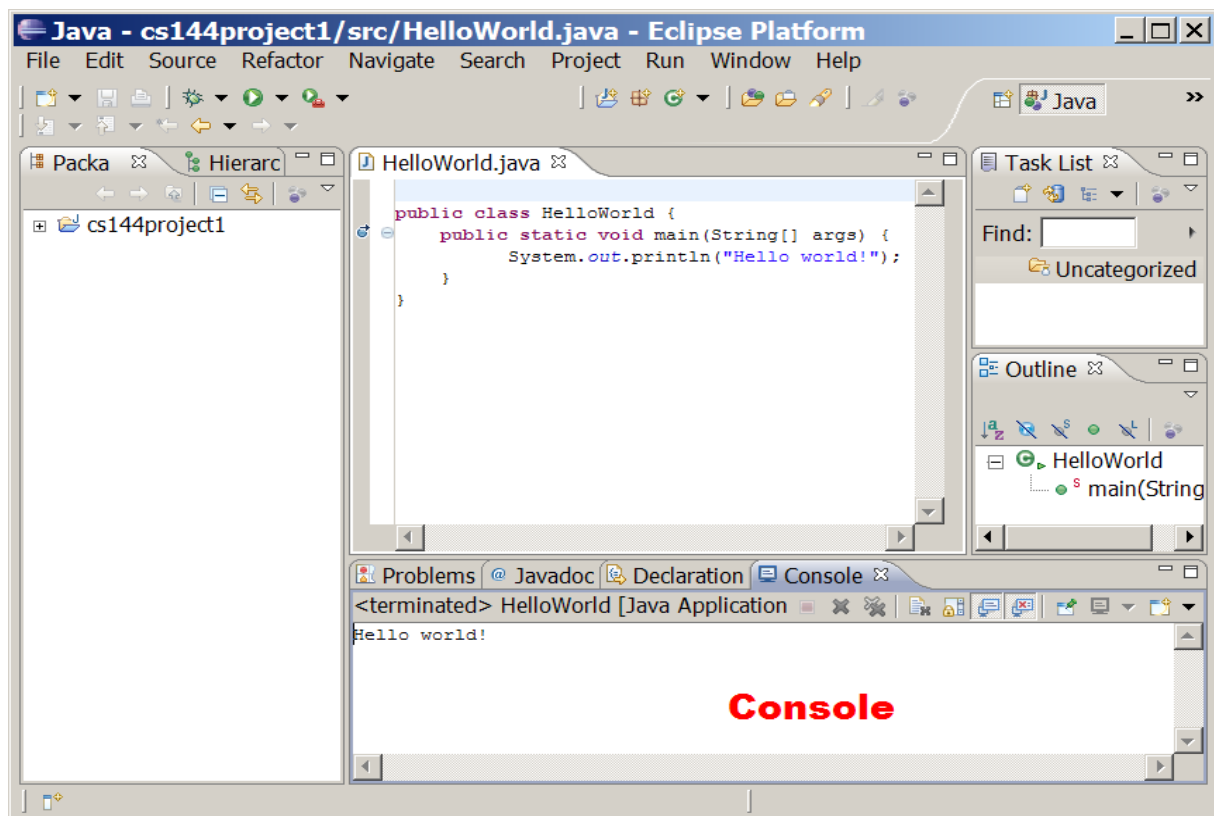
Now you have created your first Java code in eclipse. As you can see from the Package explore pane, your project now includes HelloWorld.java file. The "Editor pane" to the right of the Package explorer pane shows the actual content of the HelloWorld.java file, which simply declares HelloWorld as a public class. You can edit the content of the Java code inside the Editor pane.

Saving, compiling, and running Java code Now let us learn how to code, compile and run a Java program in Eclipse. First copy and paste the following method into the HelloWorld class definition:

```
public static void main(String[] args) {  
    System.out.println("Hello world!");  
}
```



Now save the file by selecting File->Save, or pressing Ctrl-S. When you save a Java file, Eclipse will automatically *compile* the file also, so that you don't need to compile it later when you want to run it. Now that your code has been saved and compiled, you can run your program by selecting Run -> Run, or by pressing Ctrl-F11 or by clicking on the "Run" button near the top of the window. Once your program finishes running, you will be able to see the output of your program by selecting the "Console tab" at the bottom of the window.



/*1. Write Java program to find prime numbers between 1 to n. */

```
import java.util.Scanner;
class PrimeNumbers{
    public static void main(String[] args) {
        System.out.println("***** PRIME NUMBERS *****");
        Scanner objScanner = new Scanner(System.in);
        System.out.print("\nEnter n Value:");
        long n = objScanner.nextInt();
        for (long i = 2; i <= n; i++)
        {
            boolean isprime = isNumPrime(i);
            if (isprime)
            {
                System.out.print(i + " ");
            }
        }
    }
    public static boolean isNumPrime(long number)
    {
        boolean result = true;
        for (long i = 2; i <= number / 2; i++)
        {
            if ((number % i) != 0)
            {
                result = true;
            }
            else
            {
                result = false;
                break;
            }
        }
        return result;
    }
}
```

O/P:

***** PRIME NUMBERS *****

Enter n Value: 100

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

/* 2. Write a java program that prints all real and imaginary solutions to the quadratic equation $ax^2+bx+c=0$. Read in a, b, c and use the quadratic formula */

```
import java.io.*;
class Quadratic{
    public static void main(String[] args) throws IOException{
        double x1,x2,disc,a,b,c;
        InputStreamReader obj=new InputStreamReader(System.in);
        BufferedReader br=new BufferedReader(obj);
        System.out.println("enter a,b,c values");
        a=Double.parseDouble(br.readLine());
        b=Double.parseDouble(br.readLine());
        c=Double.parseDouble(br.readLine());
        disc=(b*b)-(4*a*c);
        if(disc==0)
        {
            System.out.println("roots are real and equal ");
            x1=x2=-b/(2*a);
            System.out.println("roots are "+x1+", "+x2);
        }
        else if(disc>0)
        {
            System.out.println("roots are real and unequal");
            x1=(-b+Math.sqrt(disc))/(2*a);
            x2=(-b-Math.sqrt(disc))/(2*a);
            System.out.println("roots are "+x1+", "+x2);
        }
        else
        {
            System.out.println("roots are imaginary");
        }
    }
}
```

O/P:

enter a,b,c values

2

-4

2

roots are real and equal

roots are 1.0,1.0

```
/* 3. Write a Java program for sorting a given list of names in ascending order */
import java.lang.*;
import java.io.*;
class StringAscending{
    public static void main(String args[ ])throws IOException{
        InputStreamReader obj=new InputStreamReader(System.in);
        BufferedReader br=new BufferedReader(obj);
        System.out.print("enter how many strings");
        int n=Integer.parseInt(br.readLine( ));
        String[ ] str=new String[n];
        System.out.println("Enter "+n+" strings");
        for(int i=0;i<n;i++)
        {
            str[i]=br.readLine( );
        }
        String temp=new String( );
        for(int i=0;i<n;i++)
        {
            for(int j=i;j<n;j++)
            {
                if(str[i].compareTo(str[j])>0)
                {
                    temp=str[i];
                    str[i]=str[j];
                    str[j]=temp;
                }
            }
        }
        System.out.println("Strings in alphabetical order are");
        for(String x:str)
        {
            System.out.println(x);
        }
    }
}
```

O/P:

enter how many strings

4

Enter 4 strings

java

cprog

dotnet

sql

Strings in alphabetical order are

cprog

dotnet

java

sql

/*4. Write a java program to accept a string from user and display number of vowels, consonants, digits and special characters present in each of the words of the given text. */

```
import java.io.*;
class CountVowels{
    public static void main(String[ ] args) throws IOException{
        InputStreamReader obj=new InputStreamReader(System.in);
        BufferedReader br=new BufferedReader(obj);
        String str=new String();
        System.out.println("Enter a string");
        str=br.readLine();
        int ln=str.length();
        System.out.println("String length is"+ln);
        int d=0,v=0,s=0,c=0;
        for(int i=0;i<ln;i++)
        {
            char ch=str.charAt(i);
            if(((ch>='a') && (ch<='z')) || ((ch>='A') && (ch<='Z')))
            {
                if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')
                    v++;
                else if(ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')
                    v++;
                else
                    c++;
            }
            else if(ch>='0' && ch<='9')
                d++;
            else
                s++;
        }
        System.out.println("Digits "+d);
        System.out.println("Vowels "+v);
        System.out.println("Consonents "+c);
        System.out.println("Special Characters "+s);
    }
}
```

O/P:

Enter a string

This program counts number of vowels, consonants, digits 1234 and special characters @\$

String length is88

Digits 4

Vowels 22

Consonents 45

Special Characters 17

/*5. Write a java program to add two Matrices*/

```
import java.io.*;
class Matrix{
    int i,j;
    void Read(int mat[][],int r,int c)throws IOException{
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        for( i=0;i<r;i++)
        {
            for(j=0;j<c;j++)
            {
                mat[i][j]=Integer.parseInt(br.readLine());
            }
        }
    }
    void Display(int sum[][],int r,int c){
        for( i=0;i<r;i++)
        {
            for( j=0;j<c;j++)
            {
                System.out.print(sum[i][j]+" ");
            }
            System.out.println();
        }
    }
}
public class MatrixADD {
    public static void main(String args[])throws IOException{
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        int[][] a=new int[10][10];
        int[][] b=new int[10][10];
        int[][] c=new int[10][10];
        int i,j;
        Matrix m=new Matrix();
        System.out.println("enter no of rows and cloumns of matrix 1");
        int row1=Integer.parseInt(br.readLine());
        int col1=Integer.parseInt(br.readLine());
        System.out.println("enter no of rows and cloumns of matrix 2");
        int row2=Integer.parseInt(br.readLine());
        int col2=Integer.parseInt(br.readLine());
        if(row1==row2&&col1==col2)
        {
            System.out.println("Matrix addition is possibel");
            System.out.println("Enter elements of Matrix1 row wise");
            m.Read(a,row1,col1);
            System.out.println("Enter elements of Matrix2 row wise");
            m.Read(b,row2,col2);
            for(i=0;i<row1;i++){
                for(j=0;j<col1;j++){
                    c[i][j]=a[i][j]+b[i][j];
                }
            }
        }
    }
}
```

```
        System.out.println("Sum of two matrices is :");
        m.Display(c,row1,col1);
    }
    else
    {
        System.out.println("Matrix addition is not possible");
    }
}
}
```

O/P:

```
enter no of rows and cloumns of matrix 1
2
2
enter no of rows and cloumns of matrix 2
2
2
Matrix addition is possibel
Enter elements of Matrix1 row wise
1
1
1
1
Enter elements of Matrix2 row wise
3
3
3
3
Sum of two matrices is :
4 4
4 4
```

O/P:

```
enter no of rows and cloumns of matrix 1
2
2
enter no of rows and cloumns of matrix 2
2
3
Matrix addition is not possible
```

/*6. Write a java program to find factorials of numbers in a given range */

```
import java.io.*;
class Fact {
    // This is a recursive function.
    long factR(int n) {
        long result;
        if(n==1) return 1;
        result = factR(n-1) * n;
        return result;
    }
}
class Factorial {
    public static void main(String[] args) throws IOException{
        Fact f = new Fact();
        int i,n;
        InputStreamReader obj=new InputStreamReader(System.in);
        BufferedReader br=new BufferedReader(obj);
        System.out.println("Enter n ");
        n=Integer.parseInt(br.readLine());
        System.out.println("Factorials from 1 to "+ n);
        for(i=1;i<=n;i++)
        {
            System.out.println("Factorial of "+i+" is " + f.factR(i));
        }
    }
}
```

O/P:

Enter n

10

Factorials from 1 to 10

Factorial of 1 is 1

Factorial of 2 is 2

Factorial of 3 is 6

Factorial of 4 is 24

Factorial of 5 is 120

Factorial of 6 is 720

Factorial of 7 is 5040

Factorial of 8 is 40320

Factorial of 9 is 362880

Factorial of 10 is 3628800

/*7. Write a java program to make rolling a dice 10,000 times and counts the number of times each face has rolled *Hint: Math.random()* */

```
import java.util.Random;
public class RollDice{
    public static void main( String[ ] args){
        Random randomNumbers = new Random( );
        int frequency1 = 0;
        int frequency2 = 0;
        int frequency3 = 0;
        int frequency4 = 0;
        int frequency5 = 0;
        int frequency6 = 0;
        int face;
        for ( int roll = 1; roll <= 10000; roll++ )
        {
            face = 1 + randomNumbers.nextInt( 6 );
            switch(face)
            {
                case 1:    ++frequency1;
                           break;
                case 2:    ++frequency2;
                           break;
                case 3:    ++frequency3;
                           break;
                case 4:    ++frequency4;
                           break;
                case 5:    ++frequency5;
                           break;
                case 6:    ++frequency6;
                           break;
            }
        }
        System.out.println( "Face\tFrequency" ); // output headers
        System.out.printf("1\t%d\n2\t%d\n3\t%d\n4\t%d\n5\t%d\n6\t%d\n" ,
            frequency1, frequency2, frequency3, frequency4,frequency5, frequency6 );
    }
}
```

O/P:

Face	Frequency
1	1595
2	1683
3	1679
4	1703
5	1708
6	1632

/*8. Write a java program to make rolling a pair of dice 10,000 times and counts the number of times doubles of are rolled for each different pair of doubles. Hint: Math.random() */

```
import java.util.Random;
public class RollDoubles{
    public static void main( String[ ] args){
        Random randomNumbers = new Random( );
        int frequency1 = 0;
        int frequency2 = 0;
        int frequency3 = 0;
        int frequency4 = 0;
        int frequency5 = 0;
        int frequency6 = 0;
        int dice1,dice2;
        for ( int roll = 1; roll <= 10000; roll++ )
        {
            dice1 = 1 + randomNumbers.nextInt( 6 );
            dice2 = 1 + randomNumbers.nextInt( 6 );
            if(dice1==dice2)
            {
                switch(dice1)
                {
                    case 1:    ++frequency1;
                               break;
                    case 2:    ++frequency2;
                               break;
                    case 3:    ++frequency3;
                               break;
                    case 4:    ++frequency4;
                               break;
                    case 5:    ++frequency5;
                               break;
                    case 6:    ++frequency6;
                               break;
                }
            }
        }
        System.out.println( "Doubles\tFrequency" );
        System.out.printf( "1\t%d\n2\t%d\n3\t%d\n4\t%d\n5\t%d\n6\t%d\n",
            frequency1, frequency2, frequency3, frequency4,frequency5, frequency6 );
    }
}
```

O/P:

Doubles	Frequency
1	260
2	301
3	292
4	267
5	288
6	313

/*9. Write a java program to make rolling a pair of dice 10,000 times and counts the number of times each pair are rolled Hint: Math.random() */

```
import java.util.Random;
public class RollPair{
    public static void main( String[] args){
        Random randomNumbers = new Random();
        int[][] count=new int[6][6];
        int dice1,dice2;
        for ( int roll = 1; roll <= 10000; roll++ )
        {
            dice1 = 1 + randomNumbers.nextInt( 6 );
            dice2 = 1 + randomNumbers.nextInt( 6 );
            count[dice1-1][dice2-1]++;
        }
        System.out.println("Each pair has rolled the following number of times");
        for(int i=0;i<6;i++)
        {
            for(int j=0;j<6;j++)
                System.out.print("pair "+(i+1)+","+(j+1)+"="+count[i][j]+" ");
            System.out.println();
        }
    }
}
```

O/P:

Each pair has rolled the following number of times

pair 1,1=283 pair 1,2=293 pair 1,3=287 pair 1,4=304 pair 1,5=297 pair 1,6=274
pair 2,1=260 pair 2,2=285 pair 2,3=279 pair 2,4=291 pair 2,5=289 pair 2,6=276
pair 3,1=270 pair 3,2=280 pair 3,3=275 pair 3,4=300 pair 3,5=326 pair 3,6=243
pair 4,1=272 pair 4,2=292 pair 4,3=259 pair 4,4=260 pair 4,5=260 pair 4,6=275
pair 5,1=244 pair 5,2=284 pair 5,3=295 pair 5,4=284 pair 5,5=282 pair 5,6=271
pair 6,1=284 pair 6,2=256 pair 6,3=267 pair 6,4=255 pair 6,5=254 pair 6,6=294

/* 10. Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read display the complete set of unique values input after the user enters each new value.*/

```
import java.util.Scanner;
public class Duplicate{
    public static void main(String[ ] args){
        int a[]={0,0,0,0,0},t,i,j,s=0,r=0;
        Scanner z=new Scanner(System.in);
        System.out.println("Enter 5 unique values between 10 & 100 ");
        for(j=0;j<5;j++)
        {
            t=z.nextInt();
            if(t>=10&&t<=100)
            {
                for(i=0;i<r;i++)
                {
                    if(a[i]==t)
                    s++;
                }
                if(s>0)
                {
                    System.out.println("Duplicate value found retry");
                    s--;
                    j--;
                    continue;
                }
                else
                {
                    a[j]=t;
                    r++;
                }
            }
            else
            {
                System.out.println("value must be in between 10 & 100");
                j--;
            }
        }
        System.out.print("The five unique values are ");
        for(i=0;i<5;i++)
        {
            System.out.print(a[i]+" ");
        }
    }
}
```

O/P-1:

Enter 5 unique values between 10 & 100

10

20

30

40

50

The five unique values are 10 20 30 40 50

O/P-2:

Enter 5 unique values between 10 & 100

10

8

value must be in between 10 & 100

20

30

20

Duplicate value found retry

40

6

value must be in between 10 & 100

50

The five unique values are 10 20 30 40 50

O/P-3:

Enter 5 unique values between 10 & 100

5

value must be in between 10 & 100

6

value must be in between 10 & 100

90

46

58

58

Duplicate value found retry

90

Duplicate value found retry

45

87

The five unique values are 90 46 58 45 87

/* 11. Write a java program to read the time intervals (HH:MM) and to compare system time if the system time between your time intervals print correct time and exit else try again to repute the same thing. By using StringTokenizer class */

```
import java.util.*;
import java.text.*;
class Tokenizer{
    static int[] cal(String y) {
        String a,b,x=":";
        int i[] = {0,0};
        StringTokenizer st=new StringTokenizer(y,x);
        a=(String) st.nextElement();
        b=(String) st.nextElement();
        i[0]=Integer.parseInt(a);
        i[1]=Integer.parseInt(b);
        return i;
    }
}
public class GetCurrentDateTime{
    public static void main(String[] args){
        SimpleDateFormat dateFormat = new SimpleDateFormat("HH:mm");
        Calendar cal = Calendar.getInstance();
        String y =dateFormat.format(cal.getTime());
        while(true)
        {
            String x,t1,a,b;
            int minute,hour;
            int HH[]={0,0},MM[]={0,0};
            t1=dateFormat.format(cal.getTime());
            int time1[]=Tokenizer.cal(t1);
            hour=time1[0];
            minute=time1[1];
            System.out.println("Enter the time intervals in HH MM format");
            Scanner z=new Scanner(System.in);
            String t2=z.next();
            String t3=z.next();
            int time2[]=Tokenizer.cal(t2);
            HH[0]=time2[0];
            MM[0]=time2[1];
            int time3[]=Tokenizer.cal(t3);
            HH[1]=time3[0];
            MM[1]=time3[1];
            if(HH[0]>HH[1])
            {
                int t=HH[0];
                HH[0]=HH[1];
                HH[1]=t;
            }
            if(HH[0]==HH[1]&&MM[0]>MM[1])
            {
                int t=MM[0];
```

```
        MM[0]=MM[1];
        MM[1]=t;
    }

    if((hour>=HH[0]&&hour<HH[1]) || (hour==HH[0]&&
        hour==HH[1])&& (minute>=MM[0]&&minute<=MM[1]))
    {
        System.out.println("Current time is "+hour+" : "+minute);
        break;
    }
    else
    {
        System.out.println("Try again");
    }
}
}
```

O/P:

Enter the time intervals in HH MM fommat

05:22

09:30

Try again

Enter the time intervals in HH MM fommat

15:30

20:30

Try again

Enter the time intervals in HH MM fommat

18:30

22:20

Current time is 20 : 4

/*12. Write java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. The program derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub class overridden area() so that it returns the area of a rectangle and a triangle respectively */

```
import java.io.*;
// Using abstract methods and classes.
abstract class Figure {
    double dim1;
    double dim2;
    Figure(double a, double b) {
        dim1 = a;
        dim2 = b;
    }
    // area is now an abstract method
    abstract double area();
}
class Rectangle extends Figure {
    Rectangle(double a, double b) {
        super(a, b);
    }
    // override area for rectangle
    double area() {
        System.out.println("Inside Area for Rectangle.");
        return dim1 * dim2;
    }
}
class Triangle extends Figure {
    Triangle(double a, double b) {
        super(a, b);
    }
    // override area for right triangle
    double area() {
        System.out.println("Inside Area for Triangle.");
        return dim1 * dim2 / 2;
    }
}
class AbstractAreas {
    public static void main(String args[]) throws IOException{
        BufferedReader br=new BufferedReader(new
            InputStreamReader(System.in));
        System.out.println("Enter length and breadth of Rectangle");
        double len=Double.parseDouble(br.readLine());
        double bdt=Double.parseDouble(br.readLine());
        System.out.println("Enter height and side of Triangle");
        double ht=Double.parseDouble(br.readLine());
        double sd=Double.parseDouble(br.readLine());
        Rectangle r = new Rectangle(len, bdt);
        Triangle t = new Triangle(ht, sd);
    }
}
```

```
        Figure figref; // this is OK, no object is created
        figref = r;
        System.out.println("Area is " + figref.area());
        figref = t;
        System.out.println("Area is " + figref.area());
    }
}
```

O/P 1:

Enter length and breadth of Rectangle

4

2

Enter height and side of Triangle

4

2

Inside Area for Rectangle.

Area is 8.0

Inside Area for Triangle.

Area is 4.0

O/P 2:

Enter length and breadth of Rectangle

50

20

Enter height and side of Triangle

20

15

Inside Area for Rectangle.

Area is 1000.0

Inside Area for Triangle.

Area is 150.0

/* 13. Write a JAVA program to append the contents to an existing file */

```
import java.io.*;

class Append {
    public static void main(String[] args)
    {
        String str;
        BufferedReader br = new BufferedReader(new
            InputStreamReader(System.in));

        System.out.println("Enter text ('stop' to quit).");
        try ( FileWriter fw = new FileWriter("knr.txt",true)){
            do {
                System.out.print(": ");
                str = br.readLine();
                if(str.compareTo("stop") == 0) break;
                str = str + "\r\n"; // add newline
                fw.write(str);
            } while(str.compareTo("stop") != 0);
        }
        catch(IOException exc) {
            System.out.println("I/O Error: " + exc);
        }
    }
}
```

Note: The program assumes that there exists a file called knr.txt.

O/P:

Program executes successfully. The contents entered from keyboard are appended to the file knr.txt


```
/* 14. Write a JAVA program to merge two files to a third file */
import java.io.*;

class Merge{
    public static void main(String[] args)
    {
        String str;
        try (BufferedReader f1 = new BufferedReader(new
            FileReader("file1.txt"));
            BufferedReader f2 = new BufferedReader(new
            FileReader("file2.txt"));
            FileWriter f3 = new FileWriter("file3.txt",true)){
            do {
                str = f1.readLine();
                if(str!= null) {
                    str = str + "\r\n"; // add newline
                    f3.write(str);
                }
            } while(str!=null);
            do {
                str = f2.readLine();
                if(str!= null) {
                    str = str + "\r\n"; // add newline
                    f3.write(str);
                }
            } while(str!=null);
        }
        catch(IOException exc) {
            System.out.println("I/O Error: " + exc);
        }
    }
}
```

Note: The program assumes that there exists two files called file1 and file2

O/P:

Program executed successfully. The contents of file and file2 are merged to file3

/* 15. Write a JAVA program to compare two files*/

```
import java.io.*;

class Compare{
    public static void main(String[] args)
    {
        int c1,c2;
        try ( BufferedReader f1 = new BufferedReader(new
                                                    FileReader("file1.txt"));
            BufferedReader f2 = new BufferedReader(new
                                                    FileReader("file2.txt"))){
            do {
                c1 = f1.read();
                c2 = f2.read();
                if(c1!=c2) break;
            } while(c1!=-1&&c2!=-1);
            if(c1!=c2)
                System.out.println("two files are different");
            else
                System.out.println("two files are same");
        }
        catch(IOException exc) {
            System.out.println("I/O Error: " + exc);
        }
    }
}
```

If the program assumes two file file1 and file2 with same contents then output is:

O/P:

two files are same

If the program assumes two file file1 and file2 with different contents then output is:

O/P:

two files are different

/* 16. Write a java program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file. */

```
import java.io.*;
import java.util.Scanner;
public class Split {
    public static void main(String args[]) {
        try{

            // Reading file and getting no. of files to be generated
            String inputfile = "test.txt"; // Source File Name.
            // No. of lines to be split and saved in each output file.
            double nol = 5.0;
            File file = new File(inputfile);
            Scanner scanner = new Scanner(file);
            int count = 0;
            while (scanner.hasNextLine())
            {
                scanner.nextLine();
                count++;
            }

            // Displays no. of lines in the input file.
            System.out.println("Lines in the file: " + count);
            double temp = (count/nol);
            int temp1=(int)temp;
            int nof=0;
            if(temp1==temp)
            {
                nof=temp1;
            }
            else
            {
                nof=temp1+1;
            }
            System.out.println("No. of files to be generated :"+nof);

            // Actual splitting of file into smaller files

            BufferedReader br = new BufferedReader(new
                FileReader(inputfile));

            String strLine;

            for (int j=1;j<=nof;j++)
            {
                // Destination File Location
                FileWriter fw= new FileWriter("File"+j+".txt");
                for (int i=1;i<=nol;i++)
                {
                    strLine = br.readLine();
```

```
        if (strLine!= null)
        {
            strLine=strLine+"\r\n";
            fw.write(strLine);
        }
    }
    fw.close();
}
br.close();
}
catch (Exception e) {
    System.err.println("Error: " + e.getMessage());
}
}
}
```

OUTPUT:

The program assumes a file called test.txt.
Program executed successfully and every 5 lines from test.txt have been written to file1, file2, file3 respectively

/*17. Write a Java program that creates three threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds */

```
class A implements Runnable
{
    Thread thrd;

    // Construct a new thread.
    A(String name) {
        thrd = new Thread(this, name);
        thrd.start(); // start the thread
    }
    synchronized public void run()
    {
        try{
            while(true)
            {
                Thread.sleep(1000);
                System.out.println("good morning");
            }
        }
        catch(Exception e)
        {
            System.out.println(thrd.getName() + " interrupted.");
        }
    }
}

class B implements Runnable
{
    Thread thrd;

    // Construct a new thread.
    B(String name) {
        thrd = new Thread(this, name);
        thrd.start(); // start the thread
    }
    synchronized public void run()
    {
        try{
            while(true){
                Thread.sleep(2000);
                System.out.println("hello");
            }
        }
        catch(Exception e)
        {
            System.out.println(thrd.getName() + " interrupted.");
        }
    }
}
```

```
class C implements Runnable
{
    Thread thrd;

    // Construct a new thread.
    C(String name) {
        thrd = new Thread(this, name);
        thrd.start(); // start the thread
    }
    synchronized public void run()
    {
        try{
            while(true)
            {
                Thread.sleep(3000);
                System.out.println("welcome");
            }
        }
        catch(Exception e){
            System.out.println(thrd.getName() + " interrupted.");
        }
    }
}
class ThreadDemo
{
    public static void main(String args[]){
        A t1=new A("thread1");
        B t2=new B("thread2");
        C t3=new C("thread3");
    }
}
```

O/P:

The above program gives the following output continuously

```
good morning
hello
good morning
welcome
good morning
hello
good morning
good morning
welcome
hello
good morning
good morning
hello
good morning
welcome
```

/ 18. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication */

```
class Q {
    int n;
    boolean valueSet = false;

    synchronized int get() {
        if(!valueSet)
            try {
                wait();
            }
            catch(InterruptedException e) {
                System.out.println("InterruptedException caught");
            }

        System.out.println("Got: " + n);
        valueSet = false;
        notify();
        return n;
    }

    synchronized void put(int n) {
        if(valueSet)
            try {
                wait();
            }
            catch(InterruptedException e) {
                System.out.println("InterruptedException caught");
            }

        this.n = n;
        valueSet = true;
        System.out.println("Put: " + n);
        notify();
    }
}

class Producer implements Runnable {
    Q q;
    Producer(Q q) {
        this.q = q;
        new Thread(this, "Producer").start();
    }
    public void run() {
        int i = 0;
        while(true) {
            q.put(i++);
        }
    }
}
```

```
class Consumer implements Runnable {
    Q q;
    Consumer(Q q) {
        this.q = q;
        new Thread(this, "Consumer").start();
    }

    public void run() {
        while(true) {
            q.get();
        }
    }
}

class ProducerConsumer {
    public static void main(String args[]) {
        Q q = new Q();
        new Producer(q);
        new Consumer(q);
        System.out.println("Press Control-C to stop.");
    }
}
```

O/P:

Put: 0

Press Control-C to stop.

Got: 0

Put: 1

Got: 1

Put: 2

Got: 2

Put: 3

Got: 3

Put: 4

Got: 4

Put: 5

Got: 5

Put: 6

Got: 6

Put: 7

Got: 7

Put: 8

Got: 8

Put: 9

Got: 9

Put: 10

Got: 10