**Exercise - 1 (Basics)**

a) Write a JAVA program to display default value of all primitive data type of JAVA.

**Program:**

import java.lang.\*;

class DefaultValues

{

static byte b;

static short s;

static int i;

static long l;

static float f;

static double d;

static char c;

static boolean bl;

public static void main(String[] args)

{

System.out.println("Default Value for Byte: "+b);

System.out.println("Default Value for Short: "+s);

System.out.println("Default Value for Int: "+i);

System.out.println("Default Value for Long: "+l);

System.out.println("Default Value for Float: "+f);

System.out.println("Default Value for Double: "+d);

System.out.println("Default Value for Char: "+c);

System.out.println("Default Value for Boolean: "+bl);

}

}

**Output:**

Default Value for Byte: 0

Default Value for Short: 0

Default Value for Int: 0

Default Value for Long: 0

Default Value for Float: 0.0

Default Value for Double: 0.0

Default Value for Char:

Default Value for Boolean: false

b) Write a case study on public static void main (250 words).

**Steps:**

Step-1: Click start+run and then type notepad in run dialog box and click OK. It displays Notepad.

Step-2: In run dialogbox type cmd and click OK. It displays command prompt.

Step-3: Type the following program in the Notepad and save the program as “example.java” in a current working directory.

Step-4 (Compilation): To compile the program type the following in the current working directory and then click enter.

c:\xxxx >javac example.java

Step-5 (Execution): To run the program type the following in the current working directory and then click enter.

c:\xxxx>java example

class example

{

public static void main(String args[])

{

System.out.println(“Welcome”);

}

}

**Explanation:**

Generally, the file name and class name should be the same. If it is not the same then the java file can be compiled but it cannot be executed. That is when execution gives the following error

Exception in thread “main” java. lang.NoClassDefFoundError: ex

In the “public static void main(String args[])” statement

public is an access specifier. If a class is visible to all classes then the public is used main() must be declared as public since it must be called by outside of its class.

The keyword static allows main() to be called without creating an object of the class.

The keyword void represents that main( ) does not return a value.

The main method contains one parameter String args[].

We can send some input values (arguments) at run time to the String args[] of the main method . These arguments are called command-line arguments. These command-line arguments are passed at the command prompt.

In System.out.println(“Welcome”); statement

System is a predefined class that provides access to the system.

out is the output stream.

println() method display the output in different lines. If we use print() method it display the output in the same line

c) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

**Program:**

import java.io.\*;

import java.util.\*;

class BikeRacers

{

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

int speed[]=new int[5];

for(int i=0;i<5;i++)

{

System.out.print("\nEnter the speed of Racer-"+(i+1)+": ");

speed[i]=sc.nextInt();

}

int sum=0;

for(int i=0;i<5;i++)

sum+=speed[i];

System.out.print("\nSum of Speed of Bikeracers is: "+sum);

double avg=sum/5;

System.out.print("\nAverage Speed of Bikeracers is: "+avg);

System.out.print("\nThe speed of qualifying racers are: ");

for(int i=0;i<5;i++)

{

if(speed[i]>=avg)

System.out.print("\nRacer-"+(i+1)+": "+speed[i]);

}

}

}

**Output:**

Enter the speed of Racer-1: 20

Enter the speed of Racer-2: 30

Enter the speed of Racer-3: 40

Enter the speed of Racer-4: 50

Enter the speed of Racer-5: 60

Sum of Speed of Bikeracers is: 200

Average Speed of Bikeracers is: 40.0

The speed of qualifying racers are:

Racer-3: 40

Racer-4: 50

Racer-5: 60

**Exercise - 2 (Control -flow)**

a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.

**Program:**

import java.util.Scanner;

class BinarySearch

{

public static void main(String args[])

{

int c,first,last,middle,n,search,array[];

Scanner in=new Scanner(System.in);

System.out.print("Enter number of elements: ");

n=in.nextInt();

array=new int[n];

System.out.println("Enter " + n + " elements: ");

for(c=0;c<n;c++)

array[c]=in.nextInt();

System.out.print("Enter value to find: ");

search=in.nextInt();

first=0;

last=n-1;

middle=(first+last)/2;

while(first<=last)

{

if(array[middle]<search)

first=middle+1;

else if(array[middle]==search )

{

System.out.println("Element " + search + " is found at location " + (middle + 1));

break;

}

else if(array[middle]>search)

last=middle-1;

middle=(first+last)/2;

}

if(first>last )

System.out.println("Element " + search + " is not present in the list!!!");

}

}

**Output:**

Enter number of elements: 5

Enter 5 elements: 10 20 30 40 50

Enter value to find: 40

Element 40 is found at location 3

b) Write a JAVA program to sort for an element in a given list of elements using bubble sort

**Program:**

import java.lang.\*;

public class BubbleSort

{

static void bubbleSort(int[] arr)

{

int n=arr.length;

int temp=0;

for(int i=0;i<(n-1);i++)

{

for(int j=1;j<n;j++)

{

if(arr[j-1]>arr[j])

{

temp=arr[j-1];

arr[j-1]=arr[j];

arr[j]=temp;

}

}

}

}

public static void main(String[] args)

{

int arr[]={3,60,35,2,45,320,5};

System.out.print("Array Before Bubble Sort: ");

for(int i=0;i<arr.length;i++)

{

System.out.print(arr[i] + " ");

}

System.out.println();

bubbleSort(arr);//sorting array elements using bubble sort

System.out.print("Array After Bubble Sort: ");

for(int i=0;i<arr.length;i++)

{

System.out.print(arr[i] + " ");

}

}

}

**Output:**

Array Before Bubble Sort: 3 60 35 2 45 320 5

Array After Bubble Sort: 2 3 5 35 45 60 320