***OPERATORS AND EXPRESSIONS :***

**//program of arithmetic operators- + - \* / %**

class Operators\_arithmetic

{

public static void main(String [] args)

{

int a=6,b=3;

//1.Addition

int c=a+b;

System.out.println("Addition of a & b is (a+b) :"+c);

//2.substraction

System.out.println("Substraction of a & b is (a-b) :"+(a-b));

//3.Multiplication

System.out.println("Multiplication of a & b is (a\*b) :"+(a\*b));

//4.Division

System.out.println("Divsion of a & b is (a/b) :"+(a/b));

//5.Modulo Division (remainder)

int md=a%b;

System.out.println("Modulo Division of a & b is (a%b) :"+md);

}

}

/\*output :

g:\Java> javac Operators\_arithmetic.java

g:\Java> java Operators\_arithmetic

Addition of a & b is (a+b) :9

Substraction of a & b is (a-b) :3

Multiplication of a & b is (a\*b) :18

Divsion of a & b is (a/b) :2

Modulo Division of a & b is (a%b) :0

\*/

**//program of relational operator- <, > ,<= ,>= ,== ,!=**

class relational\_operator

{

public static void main(String []args)

{

int a=10,b=10;

//1.less than operator : <

boolean c= a<b;

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

//2.greater than operator : >

System.out.println("a>b :"+(a>b));

//3.less than equal to operator : <=

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

//4.greater than equal to operator : >=

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

//5.equal to operator : ==

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

//6.not equal to operator : !=

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

}

}

/\* output:

g:\Java> javac relational\_operator.java

g:\Java>java relational\_operator

a<b :true

a>b :false

a<=b :true

a>=b :false

a==b :false

a!=b :true

\*/

**//program for assignment and shorthand operators- +=, -=,\*= ,/=, %=**

class assignment\_operator

{

public static void main(String []args)

{

int x=10;

//1. x=x+10;

x+=10;

System.out.println("Shorthand addition operator:"+x);

//2. x=x-10;

x-=10;

System.out.println("Shorthand substraction operator:"+x);

//3. x=x\*10;

x\*=10;

System.out.println("Shorthand multiplication operator:"+x);

//4. x=x/10;

x/=10;

System.out.println("Shorthand division operator:"+x);

//5. x=x%10;

x%=10;

System.out.println("Shorthand modulo division operator:"+x);

}

}

/\*OUTPUT :

g:\Java> javac assignment\_operator.java

g:\Java> java assignment\_operator

Shorthand addition operator:20

Shorthand substraction operator:10

Shorthand multiplication operator:100

Shorthand division operator:10

Shorthand modulo division operator:0

\*/

**//program for logical operator - && , ||, !**

class logical\_operator

{

public static void main(String []args)

{

int x=10,y=20;

//1.logical AND &&

boolean result= x>y && x<50;

System.out.println("Result is :"+result);

//2.logical OR ||

result= x>y || x>5;

System.out.println("Result is :"+result);

//3.logical NOT !

result= !(x<y);

System.out.println("Result is :"+result)

}

}

/\*

OUTPUT :

g:\Java> javac logical\_operator.java

g:\Java> java logical\_operator

Result is :false

Result is :true

Result is :false

\*/

**//program for increment and decrement operator- ++,--**

class increment\_decrement

{

public static void main(String [] args)

{

int x=5;

//1.pre-increment

++x;

System.out.println("Pre-increment :"+x);

//2.post-increment

x++;

System.out.println("Post-increment :"+x);

//3.pre-decrement

--x;

System.out.println("Pre-decrement :"+x);

//4.post-decrement

x--;

System.out.println("Post-decrement :"+x);

}

}

/\*

OUTPUT :

g:\Java>javac increment\_decrement.java

g:\Java>java increment\_decrement

Pre-increment :6

Post-increment :7

Pre-decrement :6

Post-decrement :5

\*/

**//program for ternary operator - ? :**

class conditional\_operator

{

public static void main(String [] args)

{

int x=45,y=50;

//expression\_1 ? expression\_2 : expression\_3

String xx= x<y ? "true" : "false";

System.out.println("XX : "+xx);

x = (y<25) ? y : 5555;

System.out.println("X : "+x);

}

}

/\*output :

XX: true

X: 5555

\*/

/\* works like if else loop

if(cond)

{

}

else

{

}

\*/

**//program for bitwise operator - & , | , <<, >> , >>>, ,^**

class bitwise\_operator

{

public static void main(String args[])

{

//1.Right shift - 10/2^2=10/4=2

System.out.println("Right Shift : "+(10>>2));

//2.Left shift - 10\*2^3=10\*8=80

System.out.println("Left shift : "+(10<<3));

//3.Right shift with zero fill - works same as right shift, only for negative number it works differnetly

System.out.println("Right shift with zero fill : "+ (10>>>2));

//4.bitwise and &

System.out.println("bitwise and : "+(10 & 2));

//5.bitwise or |

System.out.println("bitwise or : "+(10 | 2));

//6.bitwise ex-or ^

System.out.println("bitwise ex-or : "+(10 ^ 2));

}

}

/\* OUTPUT :

g:\Java> javac bitwise\_operator.java

g:\Java> java bitwise\_operator

Right Shift : 2

Left shift : 80

Right shift with zero fill : 2

bitwise and : 2

bitwise or : 10

bitwise ex-or : 8

\*/

**//Question paper programs**

**//Write a program to compute sum of digits of a given integer numbers**

import java.lang.\*;

import java.util.Scanner;

class sumofdigits

{

public static void main(String [] args)

{

int sum=0,remainder;

//taking input from user

System.out.println("Enter the number");

Scanner sc=new Scanner(System.in);

int num=sc.nextInt();

//direct input

// int num=56;

while(num>0)

{

// by % operator we get remainder i.e. last digit of a number

remainder=num%10;

//we add one by one digit, startig from end

sum=sum+remainder;

// by / operator we get quotient i.e. remaining digits

num=num/10;

}//end of while loop

//printing the result

System.out.println("sum of digits of a given number is : "+sum);

}

}

/\* OUTPUT :

g:\Java> javac sumofdigits.java

g:\Java>java sumofdigits

Enter the number

63

sum of digits of a given number is : 9

\*/

**//Write a program to find number and sum of all integers greater than 100 and less than 200 that are divisible by 7**

import java.lang.\*;

public class sum\_div

{

public static void main(String []args)

{

// variable declaration and value assigning

int sum=0,count=0;

//for loop - start from 101 as number should be greater than 100 and ends at 200 as number should be less than 200

for(int i=101;i<200;i++)

{

//to check whether number is divisible by 7 or not

if(i%7==0)

{

//adding elements and storing result back to sum

sum=sum+i; //sum=+i;

//counting number of elements which are divisible by 7

count++;

}

}

System.out.println("Number of elemeents : "+count);

System.out.println("Sum of elemeents : "+sum);

}

}

/\* OUTPUT :

C:\users> d:

D:> cd d:\prti

D:\prti> set path="C:\Program Files\Java\jdk1.8.0\_144\bin";

D:\prti> javac sum\_div.java

D:\prti java sum\_div

g:\Java> set path="C:\Program Files\Java\jdk1.8.0\_144\bin";

g:\Java> set classpath="C:\Program Files\Java\jre1.8.0\_144\lib\rt.jar";

g:\Java> javac sum\_div.java

g:\Java>java sum\_div

Number of elemeents : 14

Sum of elemeents : 2107

\*/

**//Arrays of objects**

**//Define Employee class with data members empid,name,salary. accept data of 5 objects and display it.**

import java.lang.\*;

import java.util.\*;

import java.util.Scanner.\*;

class Employee

{

// data members of class- instance variables

int empid;

String name;

float salary;

// Create object of scanner class – new keyword is used to create objects and //system.in – in is object of system class

Scanner sc=new Scanner(System.in);

public static void main(String [] args)

{

// Employee e=new Employee();

//int arr[5]; or int arr[]=new int[5]

//create array of object of employee class

Employee e[]=new Employee[5];

//initialize all objects by new keyword

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

{

e[i]=new Employee();

}

//call method getdata 5 times for 5 objects

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

{

e[i].getdata();

}

//call method display 5 times for 5 objects of employee to display data

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

{

e[i].display();

}

}

void getdata()

{

System.out.println("Enter Id : ");

empid=sc.nextInt();

System.out.println("Enter Name : ");

name=sc.next();

System.out.println("Enter Salary : ");

salary=sc.nextFloat()

}

void display()

{

System.out.println("Id = "+empid);

System.out.println("Name = "+name);

System.out.println("Salary = "+salary);

}

}

/\* output :

g:\Java>java Employee

Enter Id :

1

Enter Name :

apl

Enter Salary :

100

Enter Id :

2

Enter Name :

pll

Enter Salary :

200

Enter Id :

3

Enter Name :

sss

Enter Salary :

500

Enter Id :

4

Enter Name :

rtg

Enter Salary :

542

Enter Id :

5

Enter Name :

fff

Enter Salary :

203

Id = 1

Name = apl

Salary = 100.0

Id = 2

Name = pll

Salary = 200.0

Id = 3

Name = sss

Salary = 500.0

Id = 4

Name = rtg

Salary = 542.0

Id = 5

Name = fff

Salary = 203.0

\*/

**//program of mathematical functions - min(),max(),sqrt(),pow(),abs()**

//package com.example;

import java.lang.\*;

class mathematical

{

int a=5,b=3,Result;

float x=5.3,y=9.9,R;

public static void main(String [] args)

{

//1. min()-used to find minimum of two numbers

Result=Math.min(a,b);

R=Math.min(x,y);

System.out.println("Minimum value is : "+Result);

System.out.println("Minimum value is : "+R);

//2. max()-used to find maximum of two numbers

Result =Math.max(5,10);

System.out.println("Maximum value is : "+Result);

//3. sqrt()-find square root.-compulsary double data type

double rr=Math.sqrt(64.0);

System.out.println("Square root is: "+rr);

//4. pow()- raised to function -compulsary double data type

rr=Math.pow(2.0,3.0)

System.out.println("power is: "+rr);

//5.abs()-absoultly

Result=Math.abs(-962555);

System.out.println("abs() is : "+Result);

}

}

**//program to take input from command line and check for even or odd number.**

import java.lang.\*;

class even\_odd

{

public static void main(String [] args)

{

// taking input from command line

int a=1000; //Integer.parseInt(args[0]);

// even odd logic- if number is divisible by 2 then even otherwise odd

if(a%2==0)

{

System.out.println("Number is even : "+a);

}

else

{

System.out.println("Number is odd : "+a);

}

}

}

/\* output:

g:\Java>javac even\_odd.java

g:\Java>java even\_odd

Number is even : 1000

\*/

**//program to check whether given number is prime number or not.take command line argument**

class cmd\_prime

{

public static void main(String args[])

{

int i,m=0,flag=0;

//take input from command line

int n=Integer.parseInt(args[0]);

//m - denotes number of times for loop should excute- normally half of given //number

m=n/2;

//prime number is divisible by 1 and the number itself,so for loop starts from 2 and //ends at m(half of n)

for(i=2;i<=m;i++)

{

if(n%i==0)

{

flag=1;

}

}

if(flag==1)

System.out.println("Number is not prime : "+n);

else

System.out.println("Number is prime : "+n);

}

}

/\*output :

g:\Java>javac cmd\_prime.java

g:\Java>java cmd\_prime 46

Number is not prime

\*/

**//program to take input from command line and check for even or odd number.**

import java.lang.\*;

class even\_odd

{

public static void main(String [] args)

{

// taking input from command line

int a=Integer.parseInt(args[0]);

// even odd logic- if number is divisible by 2 then even otherwise odd

if(a%2==0)

{

System.out.println("Number is even : "+a);

}

else

{

System.out.println("Number is odd : "+a);

}

}

}

/\* output :

g:\Java> javac even\_odd.java

g:\Java> java even\_odd 56

Number is even : 56

\*/

**// Write a program to accept a number as command line arguments and print //addition of it**

import java.lang.\*;

class cmd\_addition

{

public static void main(String []args)

{

// taking input as command line argument

int a=Integer.parseInt(args[0]);

int b=Integer.parseInt(args[1]);

int result=0;

result=a+b;

System.out.println("Addition is : "+result);

}

}

/\* output :

g:\Java> javac cmd\_addition.java

g:\Java> java cmd\_addition 10 20

Addition is : 30

\*/

**// define student class and data members name,roll,marks and display //percentage.**

**//program multiple inheritance with interface**

**//program for interface**

**//program from this and supper keyword**

import java.io.\*;

import java.lang.\*;

class MultipleInheritance

{

//instance variables declare

String name;

int roll;

double m1,m2;

//parameterized constructor

MultipleInheritance(String name,int roll,double m1,double m2)

{

this.name=name;

this.roll=roll;

this.m1=m1;

this.m2=m2;

}

public static void main(String []args)

{

//create object of child class i.e. student

student s=new student("abc",1,50,60);

s.calculate\_per();

s.display();

}

}

//interface-it is an class which contains abstract (incomplete)methods

interface exam

{

public void calculate\_per();

public void display();

}

//child class-inheritance concept

//extends+implements->multiple inheritance

class student extends MultipleInheritance implements exam

{

double result;

student(String name,int roll,double m1,double m2)

{

//cal to parent class constructor

super(name,roll,m1,m2);

}

public void calculate\_per()

{

result=((m1+m2)\*100)/200;

}

public void display()

{

System.out.println("Name : "+name);

System.out.println("Percentage is : "+result);

}

}

/\* output :

g:\Java>javac MultipleInheritance.java

g:\Java>java MultipleInheritance

Name : abc

Percentage is : 55.0

\*/