**Name Allahdad**

**Section D**

**CMS(023-22-0056)**

**Lab 05**

Exercise 1 Bank.java

Write a program by creating Bank class having the following methods

getInfo() Which takes user name, type of account, and balance.

Cashdeposit( ) which takes the deposit amout, and date of deposit.

Withdraw( ) which takes the withdraw amount, and date of withdraw

Statement( ) that will display complete statement. Display final balance

**Input**

import java.util.Scanner;

class Bank

{

public static Scanner input=new Scanner(System.in);

static String name;

static String type\_account;

static double balance;

static double deposit\_amount;

static String date\_deposit;

static double withdraw\_amount;

static String date\_withdraw;

static double fbalance;

public static void getinfo()

{

System.out.println("Enter the name:");

name=input.nextLine();

System.out.println("Enter the Type account:");

System.out.println("1.Saving\n2.default\n3.current\nselect....");

type\_account=input.nextLine();

System.out.println("Enter the balance:");

balance=input.nextDouble();

}

public static void cashdeposit()

{

System.out.println("Enter the deposit amount:");

deposit\_amount=input.nextDouble();

System.out.println("Enter the date of deposit:");

input.nextLine();

date\_deposit=input.nextLine();

}

public static void withdraw()

{

System.out.println("Enter the withdraw amount:");

withdraw\_amount=input.nextDouble();

System.out.println("Enter the date of withdraw:");

input.nextLine();

date\_withdraw=input.nextLine();

}

public static void statement()

{

System.out.println("\n...........BANK..........");

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

System.out.println("Type account:"+type\_account);

System.out.println("balance:"+balance);

System.out.println("Deposit amount:"+deposit\_amount);

System.out.println("Date of deposit:"+date\_deposit);

System.out.println("With draw amount:"+withdraw\_amount);

System.out.println("Date of withdraw amount:"+date\_withdraw);

fbalance=(balance+deposit\_amount)-withdraw\_amount;

System.out.println("......................");

System.out.println("Final balance:"+fbalance);

System.out.println("......................");

}

public static void interferance()

{

getinfo();

cashdeposit();

withdraw();

statement();

}

public static void main(String args[])

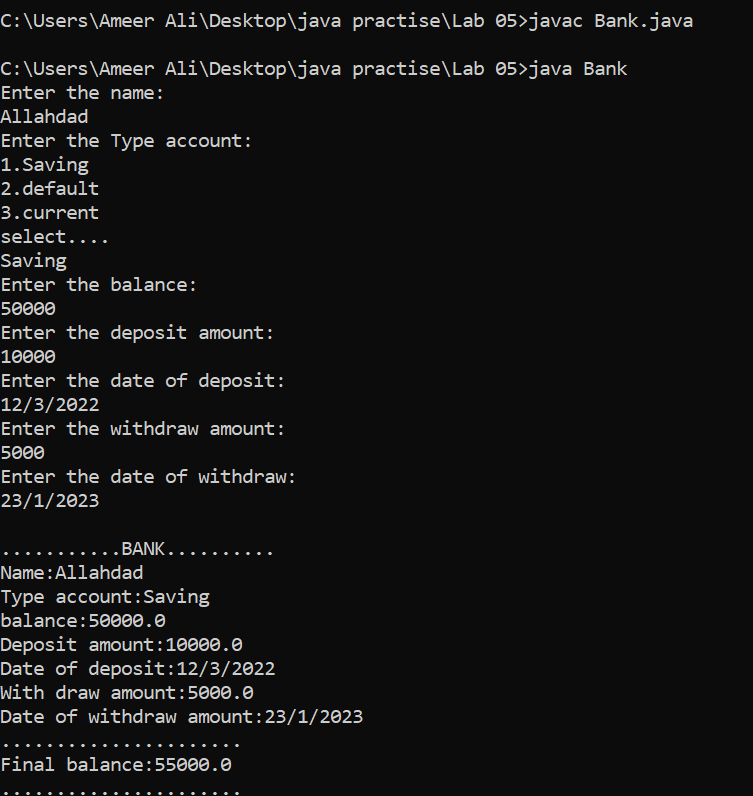
{

interferance();

}

}

**Output**



Exercise 2 (Method Overloading) Methods.java

Write a program by creating class MethodsOverloading that contains following methods

AreaRectangle(int length, int width) It takes length and width and display Area of Rectangle.

Intro(String name, int age) It takes name, and age.Display name, and age.

Force(double mass, double accleration ) It takes mass and acceleartion and display force

**Input**

import java.util.Scanner;

class Methodoverloading

{

public static Scanner input= new Scanner(System.in);

static int length,width;

static String name;

static int age;

static double force;

static double acceleration;

static double mass;

static int rectanglearea;

public static void arearectangle(int length,int width)

{

rectanglearea=length\*width;

System.out.println("Area of length is "+rectanglearea);

}

public static void intro(String name,int age)

{

System.out.println("name:"+name+"\nAge:"+age);

}

public static void force(double mass,double acceleration)

{

force=mass\*acceleration;

System.out.println("Force is :"+force);

}

public static void main(String args[])

{

System.out.println("\n");

System.out.print("Enter the length:");

length=input.nextInt();

System.out.print("Enter the width:");

width=input.nextInt();

arearectangle(length,width);

System.out.println("\n");

System.out.print("Enter the Name:");

input.nextLine();

name=input.nextLine();

System.out.print("Enter the Age:");

age=input.nextInt();

intro(name,age);

System.out.println("\n");

System.out.print("Enter the Mass:");

mass=input.nextInt();

System.out.print("Enter the acceleration:");

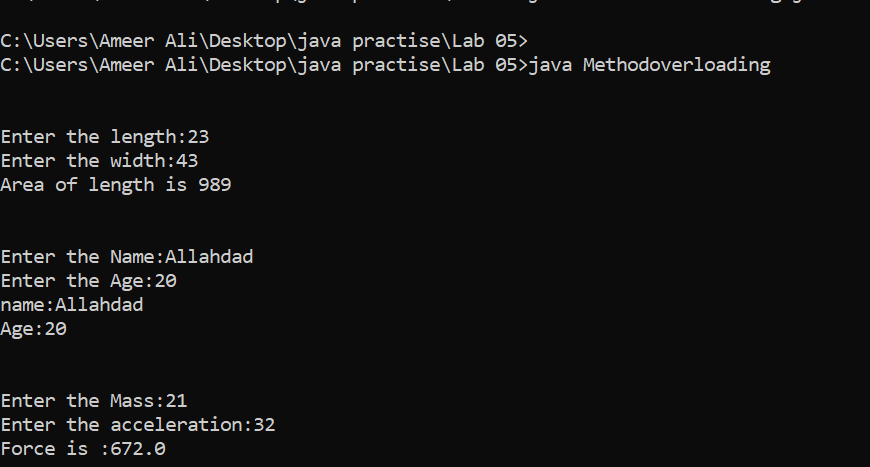
acceleration=input.nextInt();

force(mass,acceleration);

}

}

**Output**



Exercise 3 (Static Method) FindFirstLastDigit.java

Write a static method named lastDigit that returns the last digit of an integer. For example, FirstlastDigit(3852) should return 3 and 2.

**Input**

import java.util.Scanner;

class Firstlastdigit

{

public static Scanner input=new Scanner(System.in);

static String num;

static char a,b;

public static void lastdigit()

{

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

num=input.nextLine();

a=num.charAt(0);

System.out.println("First number is :"+a);

b=num.charAt(num.length()-1);

System.out.println("Last number is :"+b);

}

public static void main(String args[])

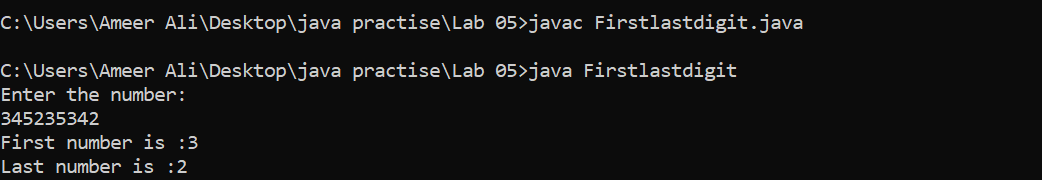
{

lastdigit();

}

}

**Output**



Exercise 4 (Factorial) FactorialDetector.java

Write a program that takes input from the user. It tells the factorial number of the input.

Output

Enter any input: 24

24 is the factorial of : 4

Enter any input: 6

6 is the factorial of : 3

Enter any input: 10

10 has no any factorial!

**Input**

import java.util.Scanner;

class EX\_4{

static int fact(long num){

if(num>1)

return (int)(num\*fact(num-1));

else

return 1;

}

public static void main(String[] agrs)

{

Scanner ip = new Scanner(System.in);

System.out.print("Enter Number :");

long fact =ip.nextLong();

for(int i=1;i<=17;i++)

{int x =0;

if(fact(i)==fact)

{x++;

System.out.println(fact+":Is factorial of:"+(i));break;

}else if(i==17 && x ==0)

{

if(fact==0){System.out.println(fact+":is factorial of:1");break;}

System.out.println(fact+":Has No factorial.");

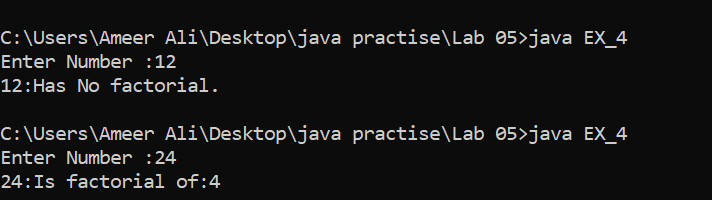
}

}

}

}

**Output**



Exercise 5 (Recursion) FibonacciSeries.java

Write a program that find fibonacci series using recursive function and also using loop. Also, takes input and check number at particular index of fibonacci series

**Input**

class EX\_5{

static int x=0,y=1,z;

public static void main(String args[])

{

System.out.print(x+" "+y+" ");

EX\_4 obj = new EX\_4();

int num=5;

obj.mymethod(num);

}

void mymethod(int n)

{

if(n>=1)

{

z=x+y;

System.out.print(" "+z);

x=y;

y=z;

mymethod(n-1);

}

}

}

**Output**

