**EXPEREIMENT-1**

**Title - CONSTRUCTOR OVERLOADING**

**1.**

public class Cons{

int id;

String name;

int rollno;

Cons(int i,String n){

id = i;

name = n;

}

Cons(int i,String n,int r){

id = i;

name = n;

rollno=r;

}

public void display()

{

System.out.println(id+" "+name+" "+rollno);

}

public static void main(String args[]){

Cons c1 = new Cons(58,"Valli");

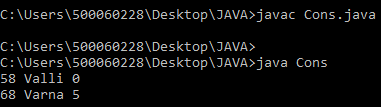
Cons c2 = new Cons(68,"Varna",05);

c1.display();

c2.display();

}

}

****

**2.**

**ALGORITHM-**

1. Write the class , inside the class declare the variable a,s,id with the respective datatypes.
2. Create a parameterized void method named “hello” with parameter id in it.
3. Inside the method print the id along with the statement “Hello World”.
4. Create another parameterized void method named “hello” but with different parameters a,s.
5. And print the two parameters.
6. In the main() , create an object for the class.
7. Call the methods with the object created, and pass values in the methods.

public class Demo2{

int a;

String s;

int id;

public void hello(int id){

System.out.println("Hello World" +id);

}

public void hello(int a,String s){

System.out.println("id is " +a);

System.out.println("name is " +s);

}

public static void main(String args[]){

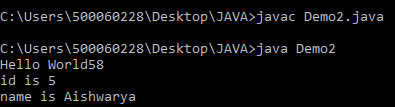
Demo2 d=new Demo2();

d.hello(58);

d.hello(5,"Aishwarya");

}

}

****

**3.**

class Student{

String name;

Student(String n){

name = n;

}

Student(){

name = "unknown";

}

public void printName(){

System.out.println(name);

}

public static void main(String[] args){

Student a = new Student("Valli");

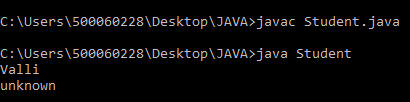
Student b = new Student();

a.printName();

b.printName();

}

}



**EXPEREIMENT-2**

**Title - METHOD OVERLOADING AND OVERRIDING**

**1.**

class Start{

public void sum(int a,int b){

System.out.println((a+b));

}

public int div(int a,int b){

return a/b;

}

}

class Calculate extends Start{

public int mul(int a,int b){

return a\*b;

}

public void sum(int a,int b){

System.out.println((a-b));

}

}

public class Demo4{

public static void main(String args[]){

Calculate s=new Calculate();

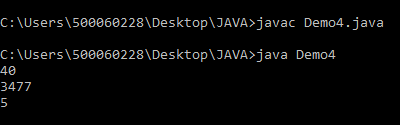
s.sum(50,10);

System.out.println(s.mul(61,57));

System.out.println(s.div(10,2));

}

}



**2.**

class Pets{

String s="Dog";

String s1="cat";

String s2="Rabbit";

public void Display(){

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

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

System.out.println(" "+s1);

System.out.println(" "+s2);

}

}

class Wild extends Pets{

String a="Tiger";

String b="Lion";

String c="leopard";

public void display(){

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

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

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

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

}

}

public class Demo5{

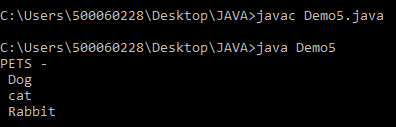
public static void main(String args[]){

Wild w=new Wild();

w.Display();

}

}



**3.**

class Jungles{

public void Wild(){

System.out.println("Tiger");

System.out.println("Lion");

}

public void Wild(String s,String s1){

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

System.out.println(" "+s1);

}

}

public class Jungle extends Jungles{

public void Sum(int a,int b){

int c=a+b;

System.out.println("Sum = "+c);

}

public void Sums(int z,int y){

int x=z\*y;

System.out.println("Multiply = "+x);

}

public static void main(String args[]){

Jungle j=new Jungle();

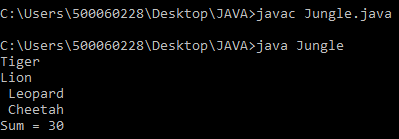
j.Wild();

j.Wild("Leopard","Cheetah");

j.Sum(10,20);

}

}



**EXPEREIMENT-3**

**Title - ENCAPSULATION AND INHERITANCE**

1. **Encapsulation**

class SCS{

private int sapid,salary;

//private double no;

String f1="Kalpana rangra";

String e1="krangra@ddn.upes.ac.in";

String f2="Dhirendra kumar";

String e2="dkumar@ddn.upes.ac.in";

String f3="Hitesh kumar";

String e3="hkumar@ddn.upes.ac.in";

String f4="Sunil kumar";

String e4="skumar@ddn.upes.ac.in";

public void Setsalary(int salary){

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

System.out.println(this.salary=salary);

}

public void Setsapid(int sapid){

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

System.out.println(this.sapid=sapid);

}

public int getsalary(){

return salary;

}

public int getsapid(){

return sapid;

}

}

class Dept extends SCS{

public void Cyber(){

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

System.out.println("JAVA - " +f2);

System.out.println("email- " +e2);

System.out.println("SPM - " +f1);

System.out.println("email- " +e1);

System.out.println("Build & Release - " +f3);

System.out.println("email- " +e3);

}

}

class Dept1 extends Dept{

public void Noncyber(){

System.out.println("NON-CYBER -");

System.out.println("Electronics - " +f4);

System.out.println("email- " +e4);

}

}

public class Encapdemo extends Dept1{

public static void main(String args[]){

Encapdemo e=new Encapdemo();

e.Cyber();

e.Noncyber();

e.Setsalary(25000);

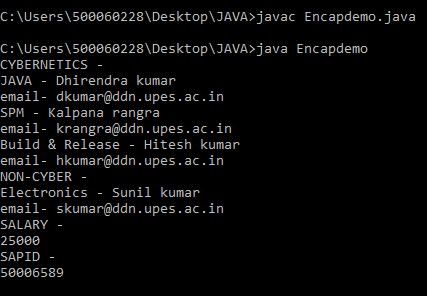
e.Setsapid(50006589);

e.getsalary();

e.getsapid();

}

}

****

1. **Inheritance**

**ALGORITHM-**

1. Create a class named “Sum” and declare the variable c.
2. Create a parameterized ”int” method named “sum” and declare the variable a,b in the method.
3. Inside the method add the two variables a,b and assign to the variable c.
4. And return the value c with “this” keyword.
5. Now create another class “Multiply” and inherit the Sum class in it.
6. Declare the variable z.
7. Create the parameterized “int” method named”multi” with x,y variables declared.
8. Inside this method multiply the both x,y values and assign it into the z variable.
9. Return the variable z using “this” keyword.
10. Now in the public class “Yum” inherit the “Multiply” class.
11. Create an object for the Yum and call the methods that were written above inside the printing statement.

class Sum{

int c;

public int sum(int a,int b){

c=a+b;

return (this.c);

}

}

class Multiply extends Sum{

int z;

public int multi(int x,int y){

z=x\*y;

return (this.z);

}

}

public class Yum extends Multiply{

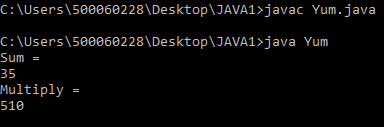
public static void main(String args[]){

Yum y=new Yum();

System.out.println(y.sum(10,25));

System.out.println(y.multi(51,10));

}}



**EXPEREIMENT-4**

**Title - BANK MANAGEMENT SYSTEM**

import java.util.Scanner;

class ExBank

{

private String accno;

private String name;

private long balance;

Scanner KB=new Scanner(System.in);

//method to open an account

void openAccount()

{

System.out.print("Enter Account No: ");

accno=KB.next();

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

name=KB.next();

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

balance=KB.nextLong();

}

//method to display account details

void showAccount()

{

System.out.println(accno+","+name+","+balance);

}

//method to deposit money

void deposit()

{

long amt;

System.out.println("Enter Amount U Want to Deposit : ");

amt=KB.nextLong();

balance=balance+amt;

}

//method to withdraw money

void withdrawal()

{

long amt;

System.out.println("Enter Amount U Want to withdraw : ");

amt=KB.nextLong();

if(balance>=amt)

{

balance=balance-amt;

}

else

{

System.out.println("Less Balance..Transaction Failed..");

}

}

//method to search an account number

boolean search(String acn)

{

if(accno.equals(acn))

{

showAccount();

return(true);

}

return(false);

}

}

class Bank

{

public static void main(String args[])

{

Scanner KB=new Scanner(System.in);

//create initial accounts

System.out.print("How Many Customer U Want to Input : ");

int n=KB.nextInt();

ExBank C[]=new ExBank[n];

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

{

C[i]=new ExBank();

C[i].openAccount();

}

//run loop until menu 5 is not pressed

int ch;

do

{

System.out.println("Main Menu 1.Display 2.Search 3.Deposit 4.Withdrawal 5.Exit");

System.out.println("Ur Choice :");

ch=KB.nextInt();

switch(ch)

{

case 1:

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

{

C[i].showAccount();

}

break;

case 2:

System.out.print("Enter Account No U Want to Search...: ");

String acn=KB.next();

boolean found=false;

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

{

found=C[i].search(acn);

if(found)

{

break;

}

}

if(!found)

{

System.out.println("Search Failed..Account Not Exist..");

}

break;

case 3:

System.out.print("Enter Account No : ");

acn=KB.next();

found=false;

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

{

s found=C[i].search(acn);

if(found)

{

C[i].deposit();

break;

}

}

if(!found)

{

System.out.println("Search Failed..Account Not Exist..");

}

break;

case 4:

System.out.print("Enter Account No : ");

acn=KB.next();

found=false;

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

{

found=C[i].search(acn);

if(found)

{

C[i].withdrawal();

Break; } }

if(!found)

{

System.out.println("Search Failed..Account Not Exist..");

}

break;

case 5:

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

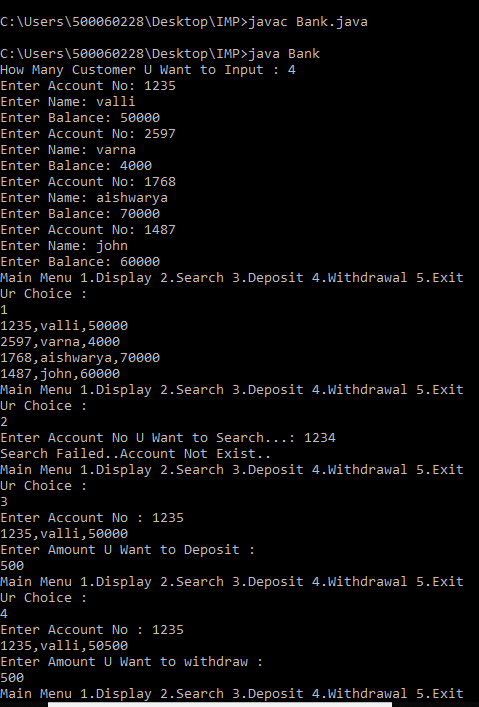
break;

}

}

while(ch!=5);

} }



**EXPEREIMENT-5**

**Title - EXCEPTIONS AND THREADS**

**1.**

**ALGORITHM-**

1. First import the thread package in the program.
2. Write the class and inherit the Thread using “extends” keyword.
3. Create the run() method and open the loop.
4. Implement the “for” loop inside the run method-
5. For(int i=1;i<=10; i++)
6. Inside the loop implement the try & catch exception.
7. Inside the exception call the sleep method (time value in it) with the Thread.

try {

Thread.sleep(1000);

}catch(InterruptedException e){}

1. After the exception, print the value of “ i ” in the for loop.
2. Close the brackets of the method.
3. Now in the main(), create an object for the class and call the start() method with the object created, to call the run method.

import java.lang.Thread;

public class Threads1 extends Thread{

public void run(){

for(int i=1;i<=10;i++)

{

try {

Thread.sleep(1000);

}catch(InterruptedException e){}

System.out.println("Threads :" +i);

}

}

public static void main(String args[])

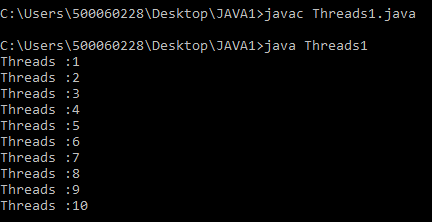
{

Threads1 th=new Threads1();

th.start();

}

}

****

**2.**

import java.lang.Thread;

class MyThread extends Thread

{

static Thread mt;

public void run()

{

for(int i=1;i<=5;i++)

{

try{

mt.join(2000);

Thread.sleep(1000);

}catch(InterruptedException e){}

System.out.println("ChildThread");

}

}

}

class Threads5

{

public static void main(String args[])throws InterruptedException

{

MyThread th=new MyThread();

th.start();

MyThread.mt=Thread.currentThread();

for(int i=1;i<=5;i++)

{

try{

Thread.sleep(1000);

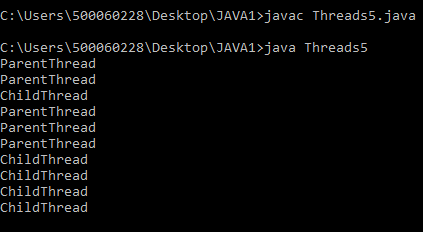
}catch(InterruptedException e){}

System.out.println("ParentThread");

}

}

}

****

**3.**

import java.lang.Thread;

class Hello extends Thread

{

public void run()

{

for(int i=1;i<=5;i++)

{

try{

Thread.sleep(1000);

}catch(InterruptedException e){}

System.out.println("HELLO");

}}}

class Hi extends Thread

{

public void run()

{

for(int i=1;i<=5;i++)

{

try{

Thread.sleep(1000);

}catch(InterruptedException e){}

System.out.println("HI");

}

}

}

public class Threads2

{ public static void main(String args[])

{

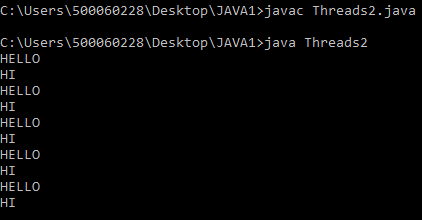
Hello h=new Hello();

h.start();

Hi h1=new Hi();

h1.start();

} }



**EXPEREIMENT-6**

**Title - AWT Frames**

**1.**

import java.awt.\*;

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

class Tab extends Frame implements WindowListener{

Label l1;

TextField tf;

Button btn;

public void windowActivated(WindowEvent e) {}

public void windowClosed(WindowEvent e) {}

public void windowClosing(WindowEvent e) {

dispose();

}

public void windowDeactivated(WindowEvent e) {}

public void windowDeiconified(WindowEvent e) {}

public void windowIconified(WindowEvent e) {}

public void windowOpened(WindowEvent arg0) {}

public Tab(){

setLayout(new FlowLayout());

l1=new Label("Enter Name");

tf=new TextField(15);

btn=new Button("submit");

add(l1);

add(tf);

add(btn);

setSize(300,300);

setVisible(true);

addWindowListener(this);

}

public static void main(String args[])

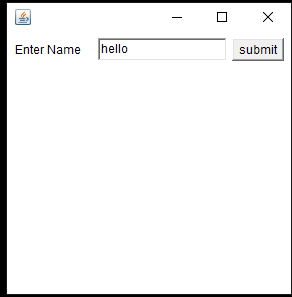
{

Tab t=new Tab();

}

}

Screenshot (52)



**2.**

import java.awt.\*;

import java.awt.event.WindowEvent;

import java.awt.event.WindowListener;

class Tab1 extends Frame implements WindowListener{

Label l1;

TextField tf;

Button b1,b2,b3,b4,b5;

public void windowActivated(WindowEvent e) {}

public void windowClosed(WindowEvent e) {}

public void windowClosing(WindowEvent e) {

dispose();

}

public void windowDeactivated(WindowEvent e) {}

public void windowDeiconified(WindowEvent e) {}

public void windowIconified(WindowEvent e) {}

public void windowOpened(WindowEvent arg0) {}

public Tab1(){

setLayout(new FlowLayout());

l1=new Label("Enter Name");

tf=new TextField(15);

b1=new Button("North");

b2=new Button("South");

b3=new Button("West");

b4=new Button("East");

b5=new Button("submit");

add(l1);

add(tf);

add(b1,"North");

add(b2,"South");

add(b3,"West");

add(b4,"East");

add(b5);

setSize(300,300);

setVisible(true);

addWindowListener(this);

}

public static void main(String args[])

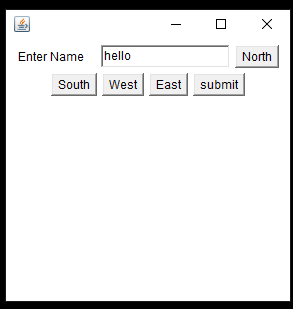
{

Tab1 t=new Tab1();

}

}

Screenshot (55)



**3.**

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

public class Program extends Frame implements ActionListener{

Button b1,b2;

public Program()

{

setLayout(new FlowLayout());

b1=new Button("click");

b2=new Button("hello");

add(b1);

add(b2);

setSize(300,300);

setVisible(true);

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b1)

{

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

}

if(ae.getSource()==b2)

{

JOptionPane.showMessageDialog(null,"Better Luck next Time");

}

}

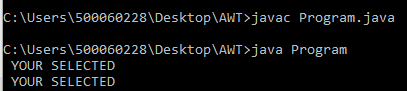
public static void main(String args[])

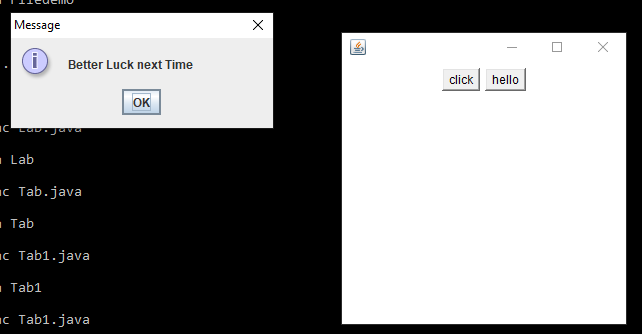
{

new Program();

}

}

****

****

**EXPEREIMENT-7**

**Title - File IO**

**1.**

import java.io.\*;

class Filedemo{

public static void main(String args[])throws IOException {

File f=new File("data.txt");

System.out.println(f.exists());

f.createNewFile();

System.out.println(f.exists());

//For Folder

File f1=new File("data");

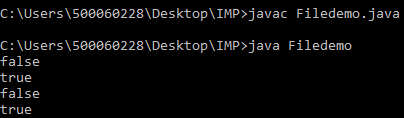
System.out.println(f1.exists());

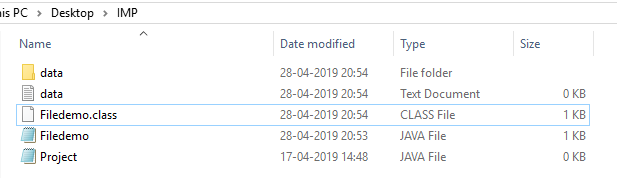
f1.mkdir();

System.out.println(f1.exists());

}

}

****

****

**2.**

import java.io.\*;

import java.lang.\*;

class File1{

public static void main(String args[]){

File f=new File("data");

try{

if(f.isDirectory())

{

String[] str=f.list();

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

{

Thread.sleep(500);

System.out.println(str[i]);

}

}

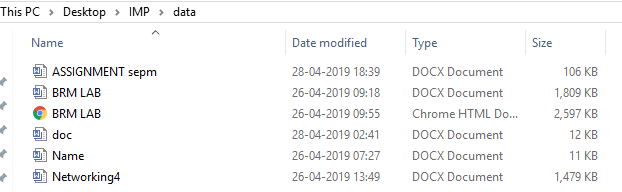
else

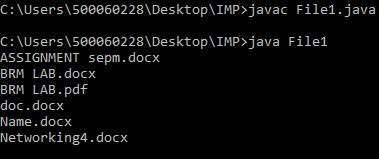
System.out.println("Isn't Directory");

}catch(Exception e){System.out.println(e.getMessage());}

}

}





**3.**

import java.io.\*;

class File2{

public static void main(String args[]) throws Exception{

FileOutputStream f=new FileOutputStream("data.txt",true);

DataInputStream dis=new DataInputStream(System.in);

char ch;

while((ch=(char)dis.read())!='\*')

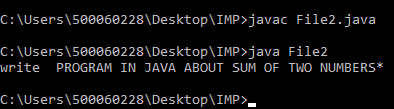
{

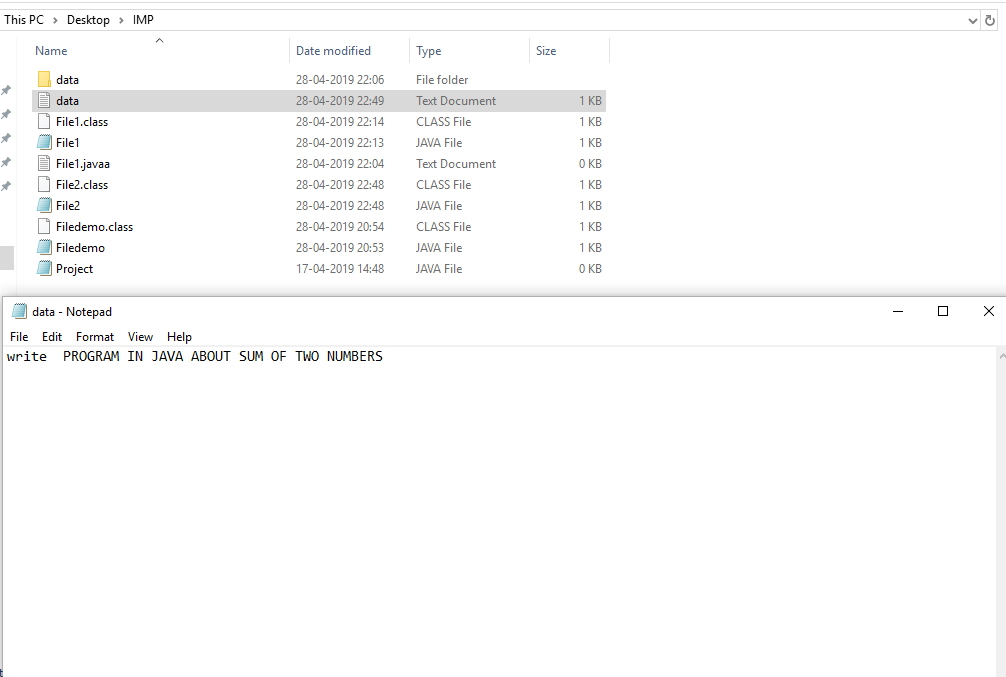
f.write(ch);

}

}

}

****

****