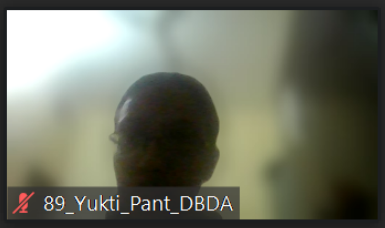
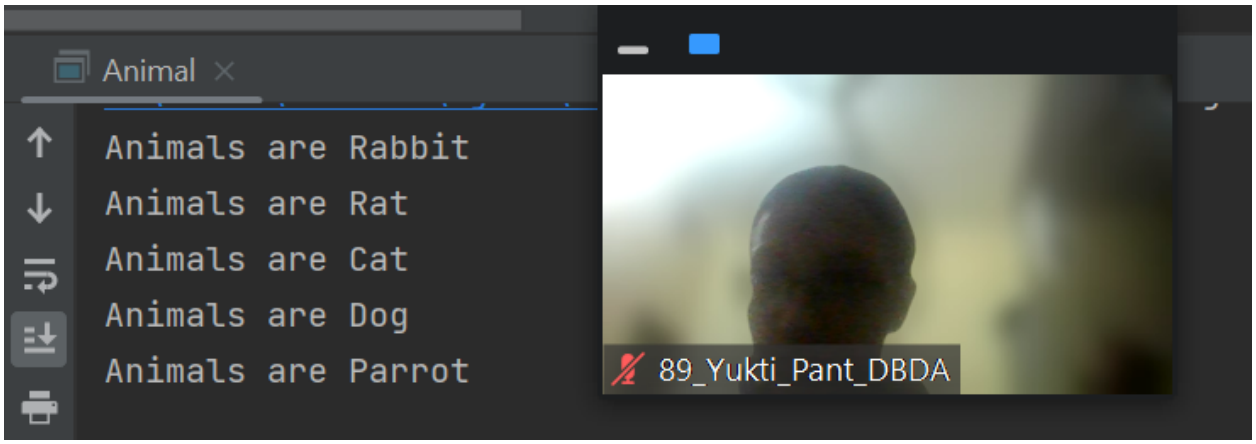
1.

```java
import java.util.*;
class Animal{
    public static void main(String[] args) {
        List<String>list_string =new ArrayList<String>();
        list_string.add("Rabbit");
        list_string.add("Rat");
        list_string.add("Cat");
        list_string.add("Dog");
        list_string.add("Parrot");
        list_string.forEach((n)->System.out.println("Animals are "+n));
    }
}
```






```java
import java.util.*;
class Animal{
    public static void main(String[] args) {
        List<String>list_string =new ArrayList<String>();
        list_string.add("Rabbit");
        list_string.add("Rat");
        list_string.add("Cat");
        list_string.add("Dog");
        list_string.add("Parrot");
        list_string.forEach((n)->System.out.println("Animals are "+n));
    }
}
```

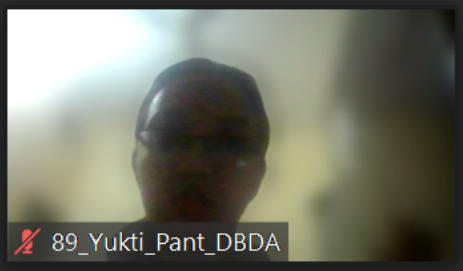
2.

```java
import java.util.*;
class BankAccount {
    private int accno;
    private double balance;
    Scanner sc = new Scanner(System.in);

    public void show() {
```

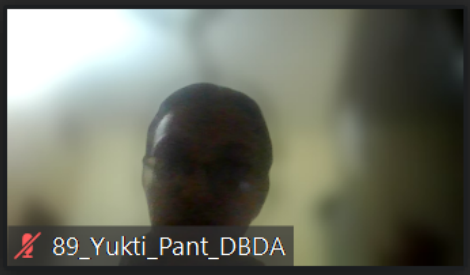
```
        System.out.println("Account no. is " + accno);
        System.out.println("Balance is " + balance);
    }

    public void deposit() {
        long amt;
        System.out.println("Enter depositing amt");
        amt = sc.nextLong();
        balance = balance + amt;
    }

    public void withdraw() {
        long amt;
        System.out.println("Enter amt you want to withdraw");
        amt = sc.nextLong();
        try {
            if (balance >= amt) {
                balance = balance - amt;
                System.out.println("Balance after withdrawl is " + balance);
            } else {
                System.out.println("Your balance is insufficient");
            }
        } catch (Exception e) {
            System.out.println(e);
        }

    }
    static class Banking {
        public static void main(String[] args) {
            Scanner sc = new Scanner(System.in);
            System.out.println("bank details");
            System.out.println("1.Show the account");
            System.out.println("2.Deposit your amt");
            System.out.println("3.Withdraw the amt");
            System.out.println("What is your choice ?");
            int ch;
            ch=sc.nextInt();
            switch (ch){
                case 1:
                    BankAccount ob1 = new BankAccount();
                    System.out.println("Bank balance is " + ob1.balance + "Bank
account no. is" + ob1.accno);
                    break;
                case 2:
                    System.out.println("Account no. ? ");
                    String accno=sc.next();
                    System.out.println("enter amount ");
                    BankAccount ob2 = new BankAccount();
                    System.out.println("now balance is " + ob2.balance);
```

```
                    break;
                case 3:
                    System.out.println("Account no. ?");
                    accno=sc.next();
                    System.out.println("Enter amount ");
                    BankAccount ob3 = new BankAccount();
                    System.out.println("now balance is "+ob3.balance);
                    break;
            }
        }
    }
}
```

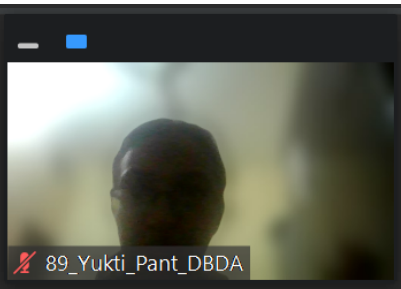
```
import java.util.*;
6 usages
class BankAccount {
    2 usages
    private int accno;
    10 usages
    private double balance;
    2 usages
    Scanner sc = new Scanner(System.in);

    public void show() {
        System.out.println("Account no. is " + accno);
        System.out.println("Balance is " + balance);
    }

    public void deposit() {
        long amt;
        System.out.println("Enter depositing amt");
        amt = sc.nextLong();
        balance = balance + amt;
    }
```



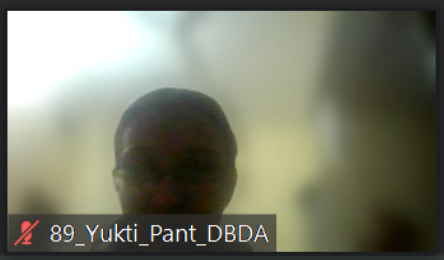
```java
    public void withdraw() {
        long amt;
        System.out.println("Enter amt you want to withdraw");
        amt = sc.nextLong();
        try {
            if (balance >= amt) {
                balance = balance - amt;
                System.out.println("Balance after withdrawl is " + balance
            } else {
                System.out.println("Your balance is insufficient");
            }
        } catch (Exception e) {
            System.out.println(e);
        }

    }
    static class Banking {
        public static void main(String[] args) {
            Scanner sc = new Scanner(System.in);
            System.out.println("bank details");
```



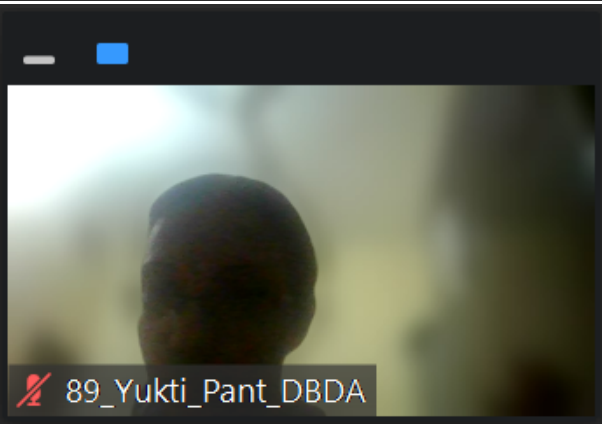
```java
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("bank details");
    System.out.println("1.Show the account");
    System.out.println("2.Deposit your amt");
    System.out.println("3.Withdraw the amt");
    System.out.println("What is your choice ?");
    int ch;
    ch=sc.nextInt();
    switch (ch){
        case 1:
            BankAccount ob1 = new BankAccount();
            System.out.println("Bank balance is " + ob1.balance + "Bank account no. is"
                    + ob1.accno);
            break;
        case 2:
            System.out.println("Account no. ? ");
            String accno=sc.next();
            System.out.println("enter amount ");
            BankAccount ob2 = new BankAccount();
            System.out.println("now balance is " + ob2.balance);
            break;
        case 3:
            System.out.println("Account no. ?");
```



```
                case 2:
                    System.out.println("Account no. ? ");
                    String accno=sc.next();
                    System.out.println("enter amount ");
                    BankAccount ob2 = new BankAccount();
                    System.out.println("now balance is " + ob2.balance);
                    break;
                case 3:
                    System.out.println("Account no. ?");
                    accno=sc.next();
                    System.out.println("Enter amount ");
                    BankAccount ob3 = new BankAccount();
                    System.out.println("now balance is "+ob3.balance);
                    break;
            }
        }
    }
}
```




```
bank details
1.Show the account
2.Deposit your amt
3.Withdraw the amt
What is your choice ?
2
Account no. ?
2
enter amount
now balance is 0.0

Process finished with exit code 0
```

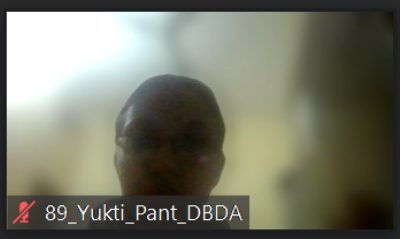


3.

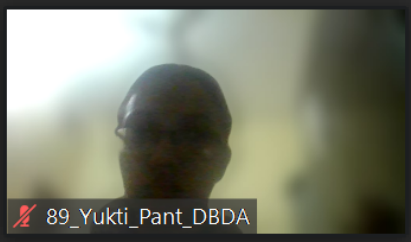
```
class Shape {
    void draw() {
        System.out.println("Lets draw shapes");
    }
    void erase(){
        System.out.println("Now erasing shapes");
    }
}
class Circle extends Shape{
    @Override
    void draw() {
        System.out.println("Lets draw a circle");
    }

    @Override
    void erase() {
        System.out.println("Now erasing circle");
    }
}
class Triangle extends Shape{
    @Override
    void draw() {
        System.out.println("Lets draw a triangle");
    }

    @Override
    void erase() {
        System.out.println("Now erasing triangle");
    }
}
class Square extends Shape{
    @Override
    void draw() {
        System.out.print("Lets draw a square");
    }

    @Override
    void erase() {
        System.out.println("Now erasing square");
    }
}
public class Solutions{
    public static void main(String[] args) {
        Shape c=new Circle();
```

```
        Shape t=new Triangle();
        Shape s=new Shape();
        c.draw();
        c.erase();
        t.draw();
        t.erase();
        s.draw();
        s.erase();
    }
}
```

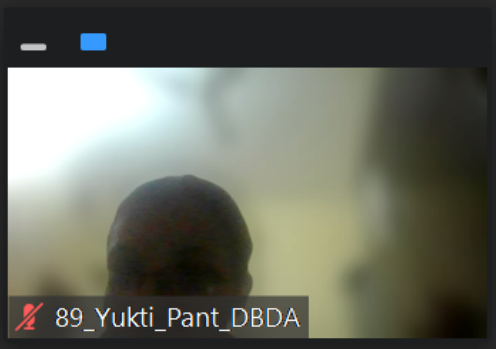
```
7 usages  3 inheritors
class Shape {
    3 usages  3 overrides
    void draw() {
        System.out.println("Lets draw shapes");
    }
    3 usages  3 overrides
    void erase(){
        System.out.println("Now erasing shapes");
    }
}
1 usage
class Circle extends Shape{
    3 usages
    @Override
    void draw() {
        System.out.println("Lets draw a circle");
    }

    3 usages
    @Override
    void erase() {
        System.out.println("Now erasing circle");
    }
```



```
}
1 usage
class Triangle extends Shape{
    3 usages
    @Override
    void draw() {
        System.out.println("Lets draw a triangle");
    }

    3 usages
    @Override
    void erase() {
        System.out.println("Now erasing triangle");
    }
}
class Square extends Shape{
    3 usages
    @Override
    void draw() {
        System.out.print("Lets draw a square");
    }

    3 usages
    @Override
```




```
C:\Users\ZS283MZ\.jdks\corretto-15.0.2\bin\java.exe "-javaagent:C:\Prog
Lets draw a circle
Now erasing circle
Lets draw a triangle
Now erasing triangle
Lets draw shapes
Now erasing shapes

Process finished with exit code 0
```



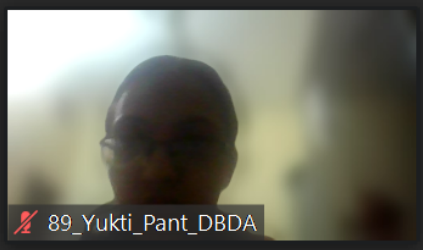
```
    @Override
    void erase() {
        System.out.println("Now erasing square");
    }
}
public class Solutions{
    public static void main(String[] args) {
        Shape c=new Circle();
        Shape t=new Triangle();
        Shape s=new Shape();
        c.draw();
        c.erase();
        t.draw();
        t.erase();
        s.draw();
        s.erase();
    }
}
```

4.

```
class GrandParent {
   public String grandFatherName;
   public String grandMotherName;

   GrandParent(String F, String M) {
        grandFatherName = F;
        grandMotherName = M;
        System.out.println("Grandfather's name is " + grandFatherName +
                " Grandmother's name is " + grandMotherName);
   }
}
class Parent extends GrandParent{
        String FatherName;
        String MotherName;

   Parent(String F, String M) {
        super(F, M);
   }
        System.out.println("Father's name is",+FatherName+" Mother's name is
"+MotherName);
        }
        public class Child extends Parent {
            Child(String F, String M) {
                super(F, M);
            }
```

```
        public static void main(String[] args) {
            Child(String FatherName, String MotherName, String
grandFatherName, String grandMotherName) {
                Child ob = new Child("Satish", "Kamakshi", "Vishnu",
"Saraswati");
                System.out.println(ob.FatherName +
";"ob.MotherName";"ob.grandFatherName";"ob.grandMotherName);

            }
        }
    }
```

```
1 usage  2 inheritors
class GrandParent {
    4 usages
    public String grandFatherName;
    4 usages
    public String grandMotherName;

    1 usage
    GrandParent(String F, String M) {
        grandFatherName = F;
        grandMotherName = M;
        System.out.println("Grandfather's name is " + grandFatherName +
                " Grandmother's name is " + grandMotherName);
    }
}
1 usage  1 inheritor
class Parent extends GrandParent{
        3 usages
        String FatherName;
        3 usages
        String MotherName;
```



```
Parent(String F, String M) {
    super(F, M);
}
    System.out.println("Father's name is",+FatherName+" Mother's name is "+MotherName);
    }
    2 usages
    public class Child extends Parent {
        1 usage
        Child(String F, String M) {
            super(F, M);
        }

        public static void main(String[] args) {
            Child(String FatherName, String MotherName, String grandFatherName, String grandMotherName)
                Child ob = new Child( F: "Satish",  M: "Kamakshi", "Vishnu", "Saraswati");
                System.out.println( x: ob.FatherName + ";"ob.MotherName";"ob.grandFatherName";"ob.grandMo

            }
        }
    }
```