

ASSIGNMENT - 2

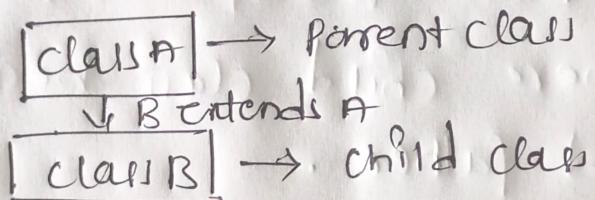
K.Rupesh Reddy
192321097
CSA0985

Q what is Inheritance and describe the types of inheritance

A Inheritance: The method to create a hierarchy between classes by inheriting from other classes.
There are 5 types of Inheritance:

1. Single Inheritance

In this we have one parent class and one child class. Both are interlinked and child is accessed by the parent class.



Program:

```
class A {  
    public void dispA() {  
        System.out.println("Base called class A");  
    }  
}
```

```
class B extends A {
```

```
    public void dispB() {  
        System.out.println("child class B called");  
    }  
}
```

class main {

public static void main (String[] args)

{ B obj=new B();

obj.fun();

obj.funB(); }

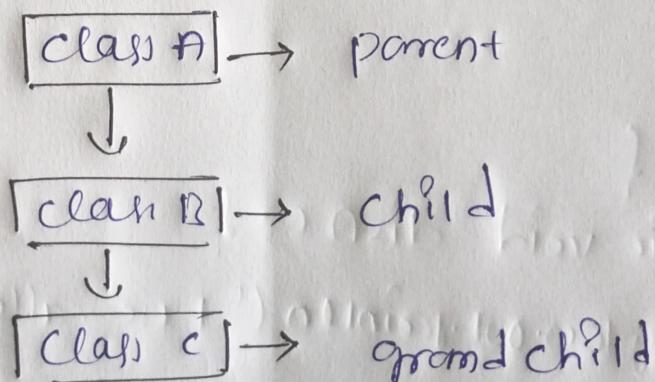
Output:

Base class is called

Child class is called.

2. Multi Level Inheritance

In this we have grand parent, parent and child class where child class become grand child for grand father.



Program:

class A {

public void fun()

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

class B extends A {

 public void disB() {

 System.out.println("child");

class C extends B {

 public void disC() {

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

class main {

 public static void main (String[] args) {

 C obj = new C();

 obj.disA();

 obj.disB();

 obj.disC();

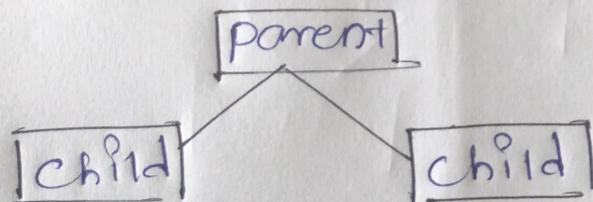
Output:

parent

child

grandchild

3. Hierarchical Inheritance: In this inheritance single parent class have many children.



Program:

class A {

 public void disp() {

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

class B extends A {

 public void disp() {

 System.out.println("child1"); } }

class C extends A {

 public void disp() {

 System.out.println("child2"); } }

class Main {

 public static void main (String [] args) {

 B objB = new B();

 C objC = new C();

 objB.disp();

 objB.disp();

 objC.disp();

Output:

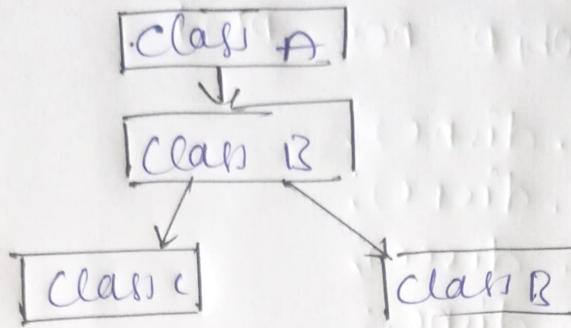
Parent

Child1

Child2

4. Hybrid Inheritance

In this inheritance is done when combination of 2 Inheritance as follow:-



Program:-

Class A {

```
public void disp() {  
    System.out.println("parent"); } }
```

Class B extends A {

```
public void disp() {  
    System.out.println("child"); } }
```

Class C extends B {

```
public void disp() {  
    System.out.println("grandchild1"); } }
```

Class D extends B {

```
public void disp() {  
    System.out.println("grandchild2"); } }
```

• Class Main {

```
    public static void main (String[] args) {
        C objC = new C();
        D objD = new D();
        objC.disA();
        objC.disB();
        objC.disC();
        objB.disA();
        objB.disB();
        objB.disC(); } }
```

Output:

Parent

Child

Grandchild 1

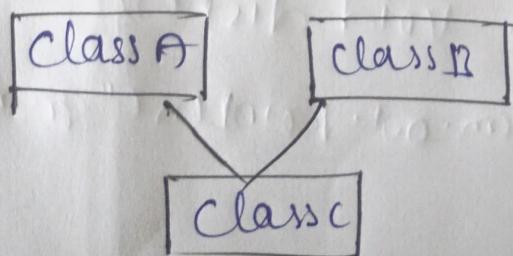
Parent

Child

Grandchild 2

5. Multiple Inheritance:

→ In this inheritance 2 parent class combines and forms a single child class where in Java it is not possible for overcoming we use Interface.



class A {

int a;

A() {

a = 5;

}

void dis1() {

System.out.println(a);

}

}

interface B {

int b = 10;

void dis2() {

System.out.println(b);

}

class C Extends A implements B {

int c = 15;

void dis3() {

System.out.println(c);

public class main {

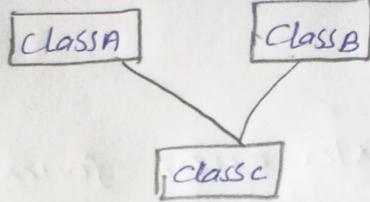
public static void main (String args[]) {

obj = new C();

obj.dis1(); obj.dis2();

obj.dis3();

22



Output :-

5

10

15

Exception handling:

4. Arithmetic exception

```
class Main{
```

```
    public static void main (String [] args) {
```

```
        try {
```

```
            int a=10/0;
```

```
            System.out.println(a);
```

```
Catch (ArithmeticException e)
```

```
{ System.out.println(e.getMessage()); }
```

```
}
```

Output: / by zero

2. Array index out of bounds

```
class Main{
```

```
    public static void main (String [] args) {
```

```
        try {
```

```
            int a[]={1,2,3};
```

```
            System.out.println(a[10]);
```

```
Catch (ArrayIndexOutOfBoundsException e)
{
    System.out.println(e.getMessage());}
}
```

3. Nested try;

```
class Main {
    public static void main (String [] args) {
        try {
            int a[] = {1, 2, 3};
```

```
        try {
            int a= 10/0; }
```

```
Catch (ArithException e)
```

```
{    System.out.println(e.getMessage());}
    System.out.println(a[10]);}
```

```
Catch (ArrayIndexOutOfBoundsException e)
```

```
{    System.out.println(e.getMessage());}
}
```

Output: / by zero

Index 10 is out of bounds 3