

# ASSIGNMENT

## JAVA

— V. Aadharsh Vishal

B.E CSE

192311009

07/2024

## Assignment 2 Inheritance

192311009

V. Aadharsh Vishal

Single inheritance.

29/07/24

O/P: child class  
Parent class.

class A {

public void disp() {

System.out.println("Parent class A");

}

}

class B extends A {

public void disp1() {

System.out.println("child class B");

}

}

public class Main {

public static void main (String [] args) {

b obj = new b();

obj.disp1();

obj.disp();

}

}

Multilevel inheritance:-

class a {

public void disp() {

System.out.println("Parent class A");

}

class <sup>a</sup> b extends a {

public void disp1() {



```
System.out.println("child class A");
```

```
}
```

```
}
```

class b extends a {

```
public void disp1() {
```

```
System.out.println("child class B");
```

```
}
```

```
}
```

class c extends b {

```
public void disp2() {
```

```
System.out.println("child class c");
```

```
}
```

```
}
```

public class Main {

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

```
c obj = new c();
```

```
obj.disp1();
```

```
obj.disp1();
```

```
obj.disp2();
```

```
}
```

```
}
```

Hierarchical:-

class a {

```
a() {
```

```
System.out.println("Parent class A");
```

```
}
```

```
}
```

class b extends a {

```
b() {
```

```
System.out.println("child class B");
```

```
}
```

```
}
```

O/P: Parent class A  
child class B  
child class c.

class c extends a {

c() {

System.out.println("child class c");

}

}

public class Main {

public static void main(String[] args) {

c obj = new c();

b obj1 = new b();

}

Hybrid inheritance.

class a {

public void dispA() {

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

}

}

class b extends a {

public void dispB() {

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

}

}

class c extends b {

public void dispC() {

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

}

}

class d extends b {

public void dispD() {

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

}

}

class Main {

OPR: Parent class A

child class c

~~child~~

Parent class A

child class B.

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

```
    C obj = new C ();
```

```
    obj. disp A ();
```

```
    obj. disp B ();
```

```
    obj. disp C ();
```

```
    D obj1 = new D ();
```

```
    obj1. disp A ();
```

```
    obj1. disp B ();
```

```
    obj1. disp D ();
```

```
}
```

3.

O/P: class A

class B

class C

class A

class B

class D.

Multiple inheritance.

```
interface A {
```

```
    int a = 10;
```

```
    void disp A ();
```

```
}
```

```
interface B.
```

```
    int b = 5;
```

```
    void disp B ();
```

```
}
```

```
class C implements A, B {
```

```
    int c = 15;
```

```
    public void disp A () {
```

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

```
}
```

```
    public void disp B () {
```

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

```
}
```

```
    public void disp C () {
```

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

```
}
```

```
    void process () {
```

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

```
}
```

```
class Main {
```

```
    public static void main
```

```
        (String [] args)
```

```
    {
```

```
        C obj = new C ();
```

```
        obj. disp A ();
```

```
        obj. disp B ();
```

```
        obj. disp C ();
```

```
    }
```

O/P = a = 10

b = 5

c = 15

a+b+c = 30



## Exception handling.

### Arithmetic Exception

```
public class Main {
```

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

```
        int a=5, b=0;
```

```
        try {
```

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

```
        }
```

```
        catch (ArithmeticException e) {
```

```
            System.out.println ("Divide by 0");
```

```
        }
```

```
    }
```

```
}
```

O/P: Arithmetic expression

~~by 0~~ / by zero

<sup>Index</sup>  
Array Out of Bound Exception.

O/P - Array Index Out of Bounds Exception

```
public 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 ("Array Index out of Bounds Exception:"  
                                + e.getMessage());
```

```
        }
```

```
    }
```

```
}
```

finally:-

O/P:- Arithmetic Exception  
This is the finally block

class main {

public static void main (String [] args) {

try {

int a = 50/0;

System.out.println("There is no errors in try block");

}

catch (ArithmeticException e) {

System.out.println("Arithmetic Exception =) " + e.getMessage());

}

finally {

System.out.println("This is the finally block");

}

}

}

IOException -

O/P:- test.txt (No such file or directory)

import java.io.\*;

class Main {

public static void main (String [] args) {

try {

File newFile = new File ("test.txt");

FileInputStream = new FileInputStream (newFile);

}

catch (IOException e) {

System.out.println(e.getMessage());

}

}

}



class NotFound Exception:

O/P:- class Not found Exception.

public class Main {

public static void main (String args[]) {

try {

class.forName ("A");

} catch (Exception e) {

e.printStackTrace();

}

}

}