



SNS COLLEGE OF ENGINEERING

Coimbatore-107
An Autonomous Institution

COURSE NAME : 23CSB201 & Object Oriented Programming

II YEAR/ III SEMESTER

UNIT – III EXCEPTION HANDLING AND MULTITHREADING

Topic: Multiple catch Clauses

Dr.P.Poonkodi

Assistant Professor(SG)

Department of Computer Science and Technology



Introduction

- we needed to handle more than one specific exception
- have more than one catch block containing the same code



Example

class Main

{

public static void main(String[] args)

{

try

{

int array[] = new int[10];

array[10] = 30 / 0;

}

catch (ArithmeticException e)

{

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

}

catch (ArrayIndexOutOfBoundsException e)

{

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

}

}

}



Multi catch block -Example

```
class Main
{
    public static void main(String[] args)
    {
        try
        {
            int array[] = new int[10];
            array[10] = 30 / 0;
        }
        catch (ArithmeticException | ArrayIndexOutOfBoundsException e)
        {
            System.out.println(e.getMessage());
        }
    }
}
```



Catching base Exception

```
class Main
```

```
{
```

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

```
    {
```

```
        try
```

```
        {
```

```
            int array[] = new int[10];
```

```
            array[10] = 30 / 0;
```

```
        }
```

```
        catch (Exception e)
```

```
        {
```

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

```
        }
```

```
    }
```

```
}
```



SNS COLLEGE OF ENGINEERING

Coimbatore-107
An Autonomous Institution

COURSE NAME : 23CSB201 & Object Oriented Programming

II YEAR/ III SEMESTER

UNIT – III EXCEPTION HANDLING AND MULTITHREADING

Topic: Nested try Statements

Dr.P.Poonkodi

Assistant Professor(SG)

Department of Computer Science and Technology



Nested Try Statement

- Sometimes a situation may arise where a part of a block may cause one error and the entire block itself may cause another error
- In such cases, exception handlers have to be nested



Syntax

//main try block

try

{

statement 1;

statement 2;

//try catch block within another try block

try

{

statement 3;

statement 4;

//try catch block within nested try block

try

{

statement 5;

statement 6;

}



Syntax

```
catch(Exception e2)
{
    //exception message
}

}
catch(Exception e1)
{
    //exception message
}

}
//catch block of parent (outer) try block
catch(Exception e3)
{
    //exception message
}
```



Example

```
public class NestedTryBlock{
public static void main(String args[]){
//outer try block
try{
    //inner try block 1
    try
    {
        System.out.println("going to divide by 0");
        int b =39/0;
    }
    //catch block of inner try block 1
    catch(ArithmeticException e)
    {
        System.out.println(e);
    }
}
```



Example

```
//inner try block 2
try
{
    int a[]=new int[5];
    //assigning the value out of array bounds
    a[5]=4;
}
//catch block of inner try block 2
catch(ArrayIndexOutOfBoundsException e)
{
    System.out.println(e);
}
System.out.println("other statement");
}
```



Example

```
//catch block of outer try block
```

```
catch(Exception e)
```

```
{
```

```
    System.out.println("handled the exception (outer catch)");
```

```
}
```

```
System.out.println("normal flow..");
```

```
}
```

```
}
```

```
C:\Users\Anurati\Desktop\abcDemo>javac NestedTryBlock.java
```

```
C:\Users\Anurati\Desktop\abcDemo>java NestedTryBlock
```

```
going to divide by 0
```

```
java.lang.ArithmeticException: / by zero
```

```
java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 5
```

```
other statement
```

```
normal flow..
```



References

- Java : the complete Reference (Eleventh Edition), Herbert Schildt, 2018.

