

Course Name: A Level (2nd Sem)

Subject: JAVA

Topic :finally block in java

Date: 15-04-20

Java finally block:

A **finally block** contains all the crucial or important statements that must be executed whether exception occurs or not. The statements present in this block will always execute regardless of whether exception occurs in try block or not such as closing a connection, stream etc.

```
class Finally1 {
    public static void main(String args[]) {
        try{
            int a=20/0;
            System.out.println(a);
        }
        catch(ArithmeticException e){
            System.out.println("Number should not be divided by zero");
        }
        finally{
            System.out.println("This is finally block");
        }
        System.out.println("Out of try-catch-finally block");
    } }
}
```

Examples using finally block:

Example1: Working of finally block when no exception occurs in try block.

```
class Finally1 {
    public static void main(String args[]){
        try{
            System.out.println("First statement of try block");
            int a=20/2;
            System.out.println(a);
        }
        catch(ArrayIndexOutOfBoundsException e){
            System.out.println("ArrayIndexOutOfBoundsException");
        }
        finally{
            System.out.println("finally block");
        }
        System.out.println("Out of try-catch-finally block");
    } }
}
```

Example2: Working of finally block when an exception occurs in try block but is not handled in the catch block:

```
class Finally2 {
    public static void main(String args[]) {
        try {
            System.out.println("First statement of try block");
            int a=20/0;
            System.out.println(a);
        }
        catch(ArrayIndexOutOfBoundsException e){
            System.out.println("ArrayIndexOutOfBoundsException");
        }
        finally{
            System.out.println("finally block");
        }
        System.out.println("Out of try-catch-finally block");
    }
}
```

Example3: When exception occurs in try block and handled properly in catch block:

```
class Finally3 {
    public static void main(String args[]) {
        try
        {
            System.out.println("First statement in try block");
            int a=20/0;
            System.out.println(a);
        }
        catch(ArithmeticException e)
        {
            System.out.println("ArithmeticException");
        }
        finally{
            System.out.println("finally block");
        }
        System.out.println("Out of try-catch-finally block");
    }
}
```

Important notes regarding finally block in java

1. A finally block must be associated with a try block, we cannot use finally without a try block. We should place those statements in this block that must be executed always.
2. Finally block is optional, however if you place a finally block then it will always run after the execution of try block.
3. In normal case when there is no exception in try block then the finally block is executed after try block.
4. The statements present in the finally block execute even if the try block contains control transfer statements like return, break or continue.

Exercise:

1. Why finally block is important in java? Explain its benefits with suitable example.
2. Can we use finally block without try-catch block. Elaborate
3. What will happen if we write our code without catch block(i.e with try and finally only). Write a suitable example demonstrating this.