

DATE: / /

PRACTICAL :- 13

PAGE No: _____

Aim: Create, debug and run Java program based on exception handling

Theory:

The Exception Handling in Java is one of the powerful mechanism to handle the runtime errors so that normal flow of application can be maintained.

In Java, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

Types of Java Exceptions.

There are mainly two types of exceptions: checked and unchecked. Here, an error is considered as the unchecked exception. According to Oracle, there are three types of exceptions

1. Checked Exception
2. Unchecked Exception
3. Error

1] checked Exception :-

The classes which directly inherit Throwable class except RuntimeException and Error are known as checked exception e.g. IOException, SQLException etc. checked exception are checked at compile-time.

2] Unchecked Exception.

The classes which inherit RuntimeException are known as unchecked exception e.g. ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException etc. Unchecked exception are not checked at compile-time, but they are checked at runtime.

3] Error.

Error is irrecoverable e.g. OutOfMemoryError, VirtualMachineError, AssertionError etc.

Java Exception keywords.

There are 5 keywords which are used in handling exception in Java.

Keyword	Description
---------	-------------

d
try

The "try" keyword is used to specify a block where we should place exception code.

The try block must be followed by either catch or finally.

It means, we can't use try block alone.

Catch

The "catch" block is used to handle the exception.

It must be preceded by try block which means we can't use catch block alone.

It can be followed by finally block letter.

Finally

The "Finally" block is used to execute the important code of the program it is executed whether an exception is handled or not.

Throw

The "throw" keyword is used to throw an exception

throws
s

The "throws" keyword is used to declare to declare exception. It doesn't - It doesn't throw an exception.

It specifies that there are may occur an exception in the method.

It is always with method signature.

Conclusion: Hence, we successfully create debug and run Java program based on exception handling.

Program :-

```

Class practical13 {
    public static void main (String arg []) {
        System.out.println (" Inside main ");
        int a = 10/0;
        System.out.println (" a = " + a);
        System.out.println (" program Exiting ---- ");
    }
}

```

```
C:\Users\Public\Java>javac practical13.java
```

```
C:\Users\Public\Java>java practical13
```

```
Inside main
```

```
Exception in thread "main" java.lang.ArithmeticException: / by zero  
    at practical13.main(practical13.java:4)
```

```
C:\Users\Public\Java>_
```