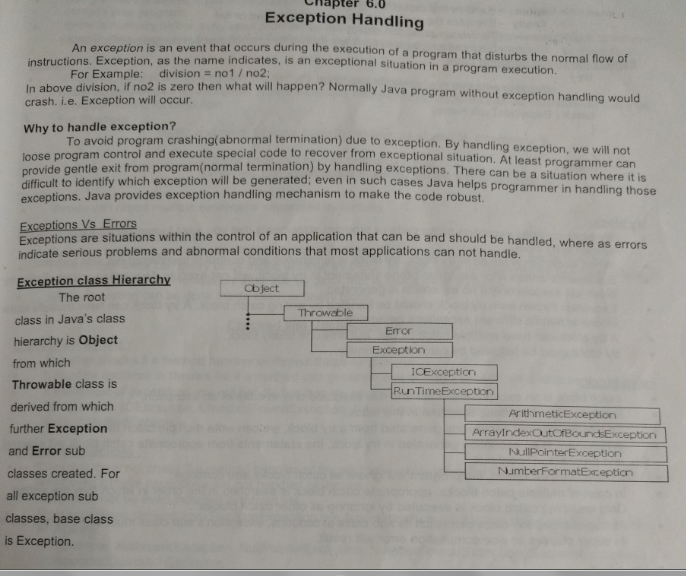
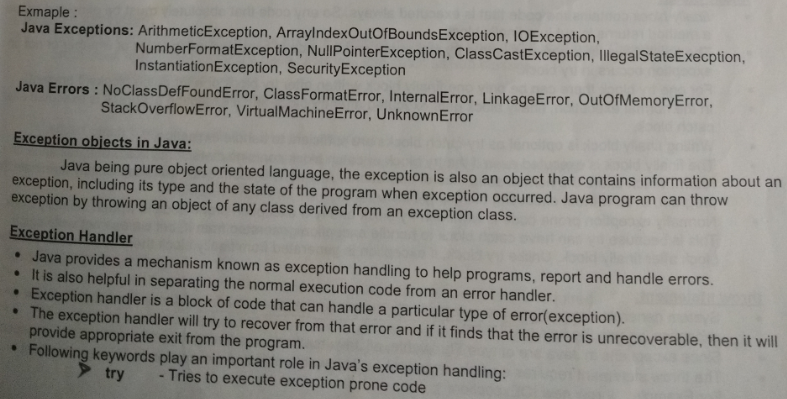
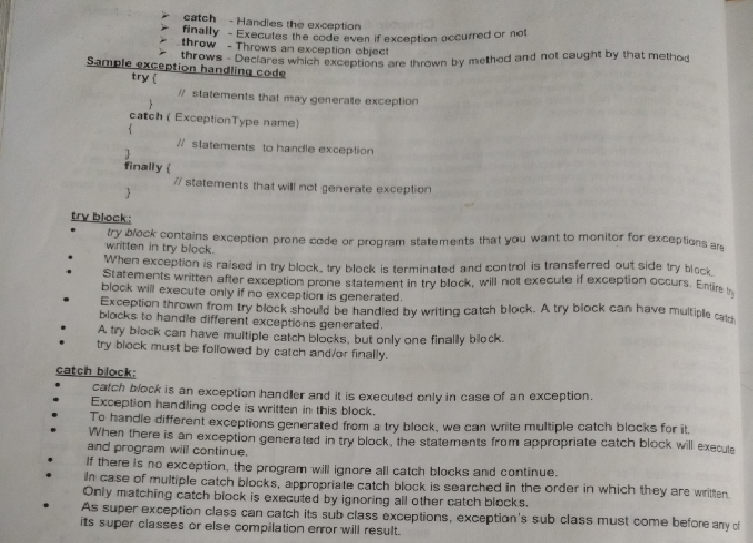
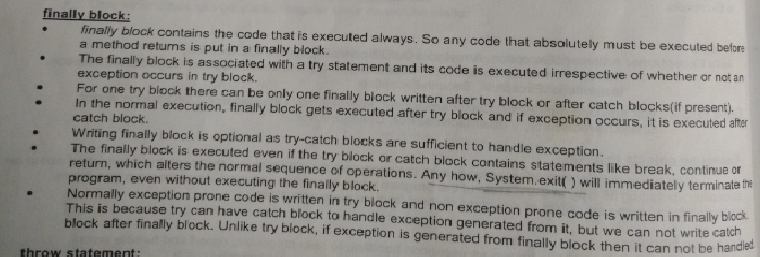
The **exception handling in java** is one of the powerful *mechanisms to handle the* ***runtime*** *errors* so that normal flow of the application can be maintained. All exception and errors types are sub classes of class **Throwable.**





try must have either catch OR finally. One out of these 2 are compulsory



Example in eclipse.

Throw and Throws :

Throw : The throw keyword in Java is used to explicitly throw an exception from a method or any block of code.

e.g. : class ThrowExcep

{

    static void fun()

    {

        try

        {

            throw **new** NullPointerException("demo");

        }

        catch(NullPointerException e)

        {

            System.out.println("Caught inside fun().");

            throw e; // rethrowing the exception

        }

    }

}

Throws :

throws is a keyword in Java which is used in the signature of method to indicate that this method might throw one of the listed type exceptions.

class tst

{

    public static void main(String[] args)throws InterruptedException

    {

        Thread.sleep(10000);

        System.out.println("Hello Geeks");

    }

}

throws is a keyword in Java which is used in the signature of method to indicate that this method might throw one of the listed type exceptions. The **caller** to these methods has to handle the exception using a try-catch block.

**Checked and Un-Checked exceptions :**

**Unchecked exceptions** are not checked at compile time. It means if your program is throwing an unchecked exception and even if you didn’t handle/declare that exception, the program won’t give a compilation error.

**Checked exceptions** are checked at compile-time. It means if a method is throwing a checked exception then it should handle the exception using [try-catch block](https://beginnersbook.com/2013/04/try-catch-in-java/) or it should declare the exception using [throws keyword](https://beginnersbook.com/2013/04/java-throws/), otherwise the program will give a compilation error.

 **If the superclass method does not declare an exception**

* If the superclass method does not declare an exception, subclass overridden method cannot declare the checked exception but it can declare unchecked exception.

 **If the superclass method declares an exception**

* If the superclass method declares an exception, subclass overridden method can declare same, subclass exception or no exception but cannot declare parent exception.

What is checked and unchecked exception