

Another program on try with resources :

```
-----  
package com.ravi.try_with_resources;  
  
import java.util.InputMismatchException;  
import java.util.Scanner;  
  
public class TryWithResourcesDemo2  
{  
    public static void main(String[] args)  
    {  
        Scanner sc = new Scanner(System.in);  
  
        try(sc)  
        {  
            System.out.println("Enter your Roll Number :");  
            int roll = sc.nextInt();  
            System.out.println("Your Roll Number is :" + roll);  
        }  
        catch(InputMismatchException e)  
        {  
            System.err.println("Input is not in a valid format");  
        }  
    }  
}
```

***Difference between Checked Exception and Unchecked Exception :

Strange behavior of Java compiler :

- 1) Why java compiler generates exception.
- 2) Test.class file is already available but still compiler is saying ClassNotFoundException
- 3) When we have 10/0; compiler does not have any issue

Checked Exception :

Checked Exception :

A checked exception is a common exception that must be declared or handled by the application code where it is thrown. Here compiler takes very much care and wanted the clarity regarding the exception by saying that, by using this code you may face some problem at runtime and you did not report me how would you handle this situation at runtime are called Checked exception, so provide either try-catch or declare the method as throws.

Except RuntimeException, all the checked exceptions are directly sub class of java.lang.Exception OR java.lang.Throwable.

Eg:

FileNotFoundException, IOException, InterruptedException, ClassNotFoundException, SQLException, CloneNotSupportedException, EOFException and so on

Unchecked Exception :-

An unchecked exception is rare and any exception that does not need to be declared or handled by the application code where it is thrown, here compiler does not take any care are called unchecked exception.

Unchecked exceptions are directly entertain by JVM because they are rarely occurred in java.

Error, sub classes of Error, RuntimeException and its sub classes are Unchecked Exception.

Eg:

ArithmaticException, ArrayIndexOutOfBoundsException, NullPointerException, NumberFormatException, ClassCastException, ArrayStoreException and so on.

Some Bullet points regarding Checked and Unchecked :

Checked Exception :

- 1) Common Exception
- 2) Compiler takes care (Will not compile the code)
- 3) Handling is compulsory (try-catch OR throws)
- 4) Directly the sub class of java.lang.Exception OR java.lang.Throwable (Excluding RuntimeException)

Unchecked Exception :

- 1) Rare Exception
- 2) Compiler will not take any care
- 3) Handling is not Compulsory
- 4) Sub class of RuntimeException OR Error (Including RuntimeException)

*Why compiler takes very much care regarding the checked Exception ?

As we know Checked Exceptions are very common exception so in case of checked exception "handling is compulsory" because checked Exception depends upon other resources as shown below.

IOException (we are depending upon System Keyboard OR Files)

FileNotFoundException (We are depending upon the file)

InterruptedException (Thread related problem)

ClassNotFoundException (class related problem)

SQLException (SQL related or database related problem)

CloneNotSupportedException (Object is the resource)

EOFException (We are depending upon the file)