**Exception**

An Exception is a runtime error situation that abnormally terminates execution of program. Exceptions occur due to the statements that can be processed by the compiler. There are basically 3 types of errors that are committed by the errors.

1. Syntax Errors
   The error that occurs due to the statements not meeting the standards or rules prescribed by the respective language.
   **Example:** Terminating a statement without semicolon.
2. Logical Errors
   These errors occur due to the bad logic implemented by the developer. It cannot be detected by the compiler.
   **Example:**

```
int StartLoop=11;
int EndLoop=1;
int answer=0;
for(int i=StartLoop;i<EndLoop;i++)
{
        answer=answer+1;
}
MessageBox.Show("answer="+answer.ToString());
```

3. Runtime Errors
   The errors that abnormally terminate the execution of a program.
   **Example:** Dividing a number by 0.

   All the Object Oriented Programming languages like Java, C# followed the structured way of handling the exception using try… catch block.

   **Syntax**

```
try
{
        //All the statements that may probably cause error
}
catch(Exception <objectvarialble>
{
        //Handle the error
}
finally
{
        //statements are executed irrespective of errors.
}
```

   **Try Block**
   This section can be used to place all the statements that may possibly cause errors.

   **Catch Block**
   Handle the errors under this section. A single try block can contain any number of catch blocks.
   **Finally Block**
   Place all the statements that are to be executed irrespective of the error situations.