EXCEPTION HANDLING

Made by :Mahmoud wael

WHAT IS EXCEPTION

C# Exceptions

When executing C# code, different errors can occur: coding errors made by the programmer, errors due to wrong input, or other unforeseeable things.

When an error occurs, C# will normally stop and generate an error message. The technical term for this is: C# will throw an **exception** (throw an error).

TYPES OF ERRORS

1. <u>Syntax errors</u>:

These are the type of errors that occur when code violates the rules of the programming language such as missing semicolons, brackets, or wrong indentation of the code.

2. <u>Logical errors</u>:

These are the type of errors that occurs when incorrect logic is implemented in the code and the code produces unexpected output.

TYPES OF ERRORS

3. Runtime errors:

These are the errors caused by unexpected condition encountered while executing the code that prevents the code to compile. These can be null pointer references, array out-of-bound errors, etc.

4. Time Limit exceeded error:

Time Limit Exceeded error is caused when a code takes too long to execute and execution time exceeds the given time in any coding contest. TLE comes because the online judge has some restrictions that the code for the given problem must be executed within the given time limit.

C# TRY AND CATCH

- The try statement allows you to define a block of code to be tested for errors while it is being executed.

- The catch statement allows you to define a block of code to be executed, if an error occurs in the try block.

```
try
{
   //  Block of code to try
}
catch (Exception e)
{
   //  Block of code to handle errors
}
```

FINALLY

- The finally statement lets you execute code, after try...catch, regardless of the result:

```
try
{
    int[] myNumbers = {1, 2, 3};
    Console.WriteLine(myNumbers[10]);
}

catch (Exception e)
{
    Console.WriteLine("Something went wrong.");
}

finally
{
    Console.WriteLine("The 'try catch' is finished.");
}
```

THE THROW KEYWORD

- The throw statement allows you to create a custom error.

- The throw statement is used together with an exception class. There are many exception classes available in C#: ArithmeticException, FileNotFoundException, IndexOutOfRangeException, TimeOutException, etc:

THE THROW KEYWORD

```csharp
static void checkAge(int age)
{
  if (age < 18)
  {
    throw new ArithmeticException("Access denied - You must be at least 18 years old.");
  }
  else
  {
    Console.WriteLine("Access granted - You are old enough!");
  }
}

static void Main(string[] args)
{
  checkAge(15);
}
```

END