**Handling Exception by try-catch:-**

We can maintain risky code in the try block and corresponding handling code inside catch block.

* A try block encloses code that may give rise one or more exception that we want to handle.
* A catch block encloses code that will be executed when an exception is thrown in the try block.
* The finally block is always executed before the method ends regardless of whether any exception are thrown in the try block.
* The try and catch blocks are bounded together we cannot separate them by putting the statement between these two block.
* If a try block can throw several different types of exception we can put several catch blocks after the try block to handle them.
* We cannot have just a try block itself each try block must always be followed by at least one block that either a catch or finally block.
* If we have a try that is part of a loop then the catch block that follows must be part of the same loop.

**Syntax:-**

*try*

*{*

*//risky code*

*}*

*catch (XXX ex)*

*{*

*//handling code*

*}*

*finally*

*{*

*//always executed*

*}*

**Example1:-**

**class** TT

{

**public** **static** **void** main (String[] args)

{

**try**

{

System.***out***.println (10/0);

}

**catch** (Exception ex)

{

System.***out***.println (ex.getMessage ());

}

}

}

**OutPut**

/ by zero

* We must catch the most derived type exception first and the most basic type exception the last otherwise our code will not compile.

**Example2:-**

while (condition)

{

try

{

…

}

catch (Exception e)

{

…  
}

}

**Example3:-**

try

{

…

}

catch(FileNotFoundException fnfe)

{

…

}

catch(IOException ioex)

{

…

}

catch(Exception e)

{

…

}