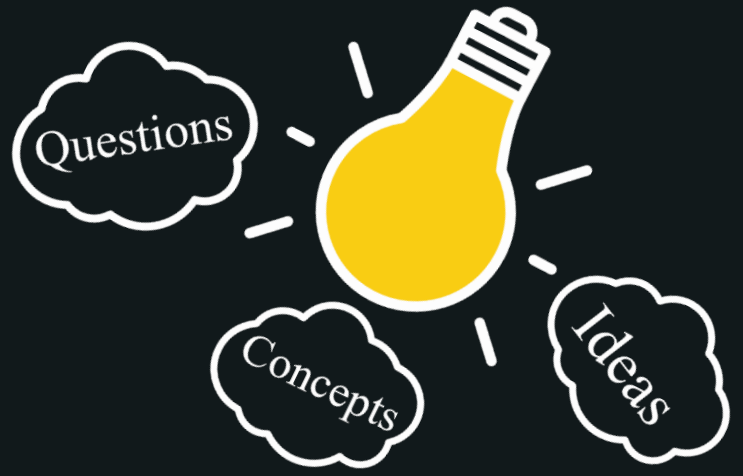

Exception Handling

C++ Programming

VEDINESH

Exception

Problems

Runtime errors.

Bad Impression
Bad Feedback

If we don't handle exception, program will crash (bad impression).

Not Business Friendly

Normal flow of the application can be maintained even after runtime errors.

Robust

What is Exception Handling?

VEDINESH

C++ provides a build-in error handling mechanism, that is called exception handling.

Try	Throw	Catch
monitor errors	throw errors	catch errors

Syntax

```
try
{
        . . . .
        throw;
        . . . .
}
```

```
catch( type 1 argument )
{
        . . . .
}
```

```
catch( type N argument )
{
        . . . .
}
```

In try we put the stmt, that we want to monitor

Throw is always used inside try

Stmt's after throw wouldn't be executed

catch should be written immediate after try

More than one catch can exist for try

But catch should be continuous (one after the another)

Example

```
int main
{
        int num1, num2, total;
        cout << “Enter 2 numbers”;
        cin >> num1 >> num2;

        try
        {
                if ( num2 == 0 )
                        throw num2;
                else
                        total =     num1 / num2;
        }

        catch( int  x)
        {
                cout << “Exception : Div by” << x;
        }
        return 0;
}
```

Program

VEDINESH

Write a program to perform division operation (use exception) to avoid divide by 0 problem.