



Multiple exceptions

Multiple catch blocks are used when we have to catch a specific type of exception out of many possible types of exceptions i.e., an exception of type char or int or short or long etc. Let's see the use of multiple catch blocks with an example.

```
#include <iostream>
#include <string>

int main()
{
    try
    {
        throw 123;
        // the following will not execute as
        // the control has been transferred to a catch clause
        throw std::string{ "Some string error" };
    }
    catch (int e)
    {
        std::cout << "Integer exception raised! The value is " << e <<
        '\n';
    }
    catch (const std::string& e)
    {
        // catch and handle the exception
        std::cout << "String exception raised!." << '\n';
        std::cout << "The exception has a value of: " << e << '\n';
    }
}
```



Here we throw multiple exceptions in the try block.

The first is of type int, and the second is of std::string type.

The moment the first exception is thrown, the control of the program is transferred to a catch clause.

This means that the remainder of the code inside the try block will not be executed.

A more realistic scenario would be:

```
#include <iostream>
#include <string>

int main()
{
    try
    {
        bool someflag = true;
        bool someotherflag = true;
        std::cout << "We can have multiple throw exceptions." << '\n';
        if (someflag)
        {
            std::cout << "Throwing an int exception." << '\n';
            throw 123;
        }
        if(someotherflag)
        {
            std::cout << "Throwing a string exception." << '\n';
            throw std::string{ "Some string error" };
        }
    }
    catch (int e)
```



```
{  
    // catch and handle the exception  
    std::cout << "Integer exception raised!" << '\n';  
    std::cout << "The exception has a value of: " << e << '\n';  
}  
  
catch (const std::string& e)  
{  
    // catch and handle the exception  
    std::cout << "String exception raised!." << '\n';  
    std::cout << "The exception has a value of: " << e << '\n';  
}  
}
```