**Handling multiple Catch blocks:**

**Example 1:**

#include<iostream.h>

#include<conio.h>

void test(int x) {

try {

if (x > 0)

throw x;

else

throw 'x';

} catch (int x) {

cout << "Catch a integer and that integer is:" << x;

} catch (char x) {

cout << "Catch a character and that character is:" << x;

}

}

void main() {

clrscr();

cout << "Testing multiple catches\n:";

test(10);

test(0);

getch();

}

**Example 2:**

#include <iostream>

**using** **namespace** std;

**int** main()

{

**int** choice;

**try**

{

cout<<"Enter any choice: ";

cin>>choice;

**if**(choice == 0)

cout<<"Hello Friend!"<<endl;

**else** **if**(choice == 1)

**throw** (100); //throw integer value

**else** **if**(choice == 2)

**throw** ('x'); //throw character value

**else** **if**(choice == 3)

**throw** (1.23f); //throw float value

**else**

cout<<"Good Bye !!!"<<endl;

}

**catch**(**int** a)

{

cout<<"Integer Exception Block, value is: "<<a<<endl;

}

**catch**(**char** b)

{

cout<<"Character Exception Block, value is: "<<b<<endl;

}

**catch**(**float** c)

{

cout<<"Float Exception Block, value is: "<<c<<endl;

}

**return** 0;

}

**Re-throwing an Exception**

**Example 1:**

#include <iostream>  
using namespace std;  
void MyHandler()  
{  
   try  
   {  
       throw “hello”;  
   }  
   catch (const char\*)  
   {  
   cout <<”Caught exception inside MyHandler\n”;  
   throw; //rethrow char\* out of function  
   }  
}  
int main()  
{  
   cout<< “Main start”;  
   try  
   {  
       MyHandler();  
   }  
   catch(const char\*)  
   {  
      cout <<”Caught exception inside Main\n”;  
   }  
       cout << “Main end”;  
       return 0;  
}

**Example 2:**

#include <iostream>

using namespace std;

void Xhandler()

{

try {

throw "hello"; // throw a char \*

}

catch(char \*) { // catch a char \*

cout << "Caught char \* inside Xhandler\n";

throw ; // rethrow char \* out of function

}

}

int main()

{

cout << "start\n";

try{

Xhandler();

}

catch(char \*) {

cout << "Caught char \* inside main\n";

}

cout << "end";

return 0;

}