# Exceptions handling

```cpp
#include <iostream>
using namespace std;
int main(int argc, char** argv) {
    int x = -10;
    cout << "before try\n";
    try{
        cout << "inside try \n";
        if (x < 0)
            throw x;
        cout << "this code never be executed\n";
    }
    catch(int x){
        cout << "exception catch!\n";
    }
    cout << "cont'd... after catch\n";
    return 0;
}
```

```
before try
inside try
exception catch!
cont'd... after catch
```

# Exceptions handling

```cpp
1    #include <iostream>
2    using namespace std;
3    int main(int argc, char** argv) {
4    //There is a special catch block called 'catch all' catch(...)
5    //that can be used to catch all types of exceptions.
6        try{
7                throw 10;
8        }
9        catch(char ch){
10           //char exception didn't exist
11           cout << "char exception catch!\n";
12       }
13       catch(...){
14           //be executed
15           cout<< "default exceptions\n";
16       }
17       return 0;
18   }
```

```
E:\try_catch\try2\try_catch
default exceptions
```

# Exceptions handling

```cpp
#include <iostream>
using namespace std;
class MyException : public exception{
    const char* what() const throw(){
        return "** my exception happened **\n";
    }
} MyExcept;
int main(int argc, char** argv) {
    try{
        throw MyExcept;
    }
    catch(exception& e){
        cout << e.what()<<'\n';
    }
    return 0;
}
```

```cpp
#include <exception>
#include <iomanip>
using namespace std;
float divide_by_zero(float x, float y){
    if (y == 0)
        throw y;
    else
        return x/y;
}
int main(int argc, char** argv) {
    try{
        divide_by_zero(1.0f, 0.0f);
    }
    catch(float e){
        cout.fill('*');
        cout<<setw(10)<<"catch "<<setw(5)<<e<<endl;
    }
    double prec  = 1.234567;
    for (int i = 1; i < 8; i++){
        cout<<setprecision(i)<<prec<<endl;
    }
    return 0;
}
```

```
****catch ****0
1
1.2
1.23
1.235
1.2346
1.23457
1.234567
```

# Exceptions handling

```cpp
double hMean(double a, double b);
int main(void)
{

    double x, y, z;
    cout << "enter two numbers: ";
    while (cin >> x >> y) {
        try { z = hMean(x, y);   }
        catch (const char* s) {
            cout << s << endl;
            cout << "enter a new pair of numbers: ";
            continue;
        }
        cout << "harmonic mean of " << x << " and " << y
             << " is " << z << endl;
        cout << "enter the next set of numbers <q to quit>: "; }
    cout << "bye!\n";
    return (0);
}
double hMean(double a, double b) {
    if (a == -b)
        throw "bad hMean() arguments: a = -b not allowed";
    return 2.0*a*b / (a + b);
}
```

```
enter two numbers: 2 3
harmonic mean of 2 and 3 is 2.4
enter the next set of numbers <q to quit>: 3 -3
bad hMean() arguments: a = -b not allowed
enter a new pair of numbers: 5 90
harmonic mean of 5 and 90 is 9.47368
enter the next set of numbers <q to quit>: q
bye!
```

# THANK YOU
# FOR
## YOUR
# ATTENTOIN
# ANY QUESTIONS?