//**cerr** does not require a buffer, so it is faster than the other ones and

//does not use the memory that **cout** uses, but because **cout** is buffered, it's more useful in //some cases. So:

// Use **cout** for the standard output.

// Use **cerr** to show errors.

// Use **clog** for logging.

#include <iostream>

using namespace std;

double answer(int a, int b)

{

if( b == 0 )

{

throw "Division by zero condition!";

}

return (a/b);

}

int main ()

{

system("cls");

b:

int x,y,r;

double z;

cout<<"\n\nENTER THE NUMBER DIVIDENT=";

cin>>x;

cout<<"\nENTER THE NUMBER DIVISOR=";

cin>>y;

try

{

r=answer(x,y);

cout<<"\nAnswer is="<<r;

}

catch(const char\* meg) //Because we are raising an exception of type const char\*,

//so while catching this exception,

//we have to use const char\* in catch block.

{

cout<<meg;

}

goto b;

getchar();

getchar();

return 0;

}///////////////////////////////////////////////////////////////////

//cerr does not require a buffer, so it is faster than the other ones and

//does not use the memory that cout uses, but because cout is buffered, it's more useful in //some cases. So:

// Use cout for the standard output.

// Use cerr to show errors.

// Use clog for logging.

#include <iostream>

#include <string>

using namespace std;

double answer(int a, int b)

{

if( b == 0 )

{

string a="Division by zero condition!";

throw a;

}

return (a/b);

}

int main ()

{

system("cls");

b:

int x,y,r;

double z;

cout<<"\n\nENTER THE NUMBER DIVIDENT=";

cin>>x;

cout<<"\nENTER THE NUMBER DIVISOR=";

cin>>y;

try

{

r=answer(x,y);

cout<<"\nAnswer is="<<r;

}

catch( string meg)

{

cout<<meg;

}

goto b;

getchar();

getchar();

return 0;

}