Rational rhapsody modeler

OOP has an imperative approach

//============================================================================

// Name : CW1.cpp

// Author : Ventsislav

// Version :

// Copyright : Your copyright notice

// Description : Hello World in C++, Ansi-style

//============================================================================

#include <iostream>

#include <stdexcept>

#include "Cup.hpp"

using namespace std;

enum Color {

WHITE, BLACK, RED, BLUE, GREEN

};

class Cupp {

public:

void fill(double quantity) {

}

double getQuantity() {

return quantity;

}

double getCapacity() {

return capacity;

}

private:

Color color;

double capacity;

double quantity;

};

void printCoord(int x, int y = 0, int z = 0) {

cout << x << ", " << y << ", " << z << endl;

}

int add(int x, int y) { //Overloading

return (x + y);

}

int add(int x, int y, int z) { //Overloading

return (x + y + z);

}

struct Complex { //structures

double re;

double im;

};

Complex operator+(const Complex& c1, const Complex& c2) { //structures

Complex result;

result.re = c1.re + c2.re;

result.im = c1.im + c2.im;

return result;

}

enum Error { //error handling

ERR\_NONE, ERR\_DIVIDE\_BY\_ZERO

};

int divide(int num1, int num2) { //error handling

if (num2 != 0) {

return num1 / num2;

}

throw std::runtime\_error("We have division by zero error! ");

}

int main() {

try {

cout << divide(3, 0);

} catch (const std::runtime\_error& err) {

//cout << "Exception caught: " << errMessage;

//runtime error processing

} catch (const std::logic\_error& err){

//handle another exception > logic error

}

return 0;

}

///////////////////////////////////////////////////////////////////////////////////////////////////

CW2 : Notes

Instance level relations

1. Composition >> One cannot exist without the other handle cannot exist without cup but cup can exist without handle
2. Aggregation >> Handle can exist without Cup

<

Handle\* extractHandle() {

Handle\* h = handle;

handle = nullptr;

return h;

}

>

1. Association >>

======================

Sdl2 Linux: packages::

build-essential

Libsdl2-dev

libsdl2-gfx-dev

Libsdl2-image-dev

Libsdl2-mixer-dev

Libsdl2-net-dev

Libsdl2-ttf-dev