Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №7**

По дисциплине «Проектирование программ в ИС»

**“Шаблонные классы”**

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Парфеевец И.А.

**Проверил:**

Монтик Н.С.

Брест 2022

ЦЕЛЬ РАБОТЫ

1. Изучение правил написания шаблонов функций.

2. Изучение правил написания шаблонов классов.

3. Реализация шаблонов функций.

4. Реализация шаблонов классов.

СПИСОК ЗАДАЧ

Индивидуальные задания на разработку классов.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить правила написания шаблонов функций и классов.

2. Для каждого индивидуального задания разработать шаблоны.

3. Реализовать консольные приложения для демонстрации работы шаблонов.

Код программы:

#include <iostream>

#include <string>

#include <cmath>

using std::string;

using std::cout;

using std::endl;

template <typename T, typename V,typename C>

class Transaction

{

public:

Transaction(T fromAcc, T toAcc, V code, C sum) :

fromAccount(fromAcc), toAccount(toAcc), code(code), sum(sum)

{ }

void getInfo()

{

std::cout << "From: " << fromAccount << "\nTo: " << toAccount

<< "\nSum: " << sum << "\nCode: " << code << endl;

}

V isvalid(T fromACC, T ToACC) {

if (fromACC.length() == ToACC.length()) {

cout << "Transaction is valid " << endl;

return 1;

}

else {

cout << "Transaction isn't valid" << endl;

return 0;

}

}

C sumc(Transaction& first ) { return first.sum + this->sum; }

C sumc(Transaction& first, Transaction& Second) { return this->sum+first.sum + Second.sum; }

private:

T fromAccount;

T toAccount;

V code;

C sum;

};

template <typename FV ,typename SV>

class calculations {

private:

FV a;

SV b;

public :

void showvalues() {

cout <<"a : " << a << endl;

cout << "b : " << b << endl << endl;

}

calculations(FV a, SV b) {

this->a = a;

this->b = b;

}

calculations() {

a = 0; b = 0;

}

FV Sumc() { return this->a + this->b; }

FV Subt() {

if (a > b) { return this-> a -(this-> b); }

else return this-> b - this-> a;

}

FV MP() { return this->a \*this-> b; }

};

template<typename EAS, typename Q,typename B>

class Employee {

private:

EAS EASs;

Q Qualification;

string name;

public :

Employee() { EASs = 0; name = "Name wasn't entered "; Qualification = "Qualification wasn't specified "; }

Employee(Employee& Sample) { name = Sample.name; Qualification = Sample.Qualification; EASs = Sample.EASs; }

Employee(EAS EASs, Q Qualification, string name) { this->EASs = EASs; this->Qualification = Qualification; this->name = name; }

void Getinfo() {

cout << "name :" << name << endl;

cout << "Qualification :" << Qualification<<endl;

cout << "EAS : " << EASs<<endl<<endl;

}

B isbetter(Employee& sample) {

if (this->EASs > sample.EASs) { cout << this->name << " is better than " << sample.name << endl; return true; }

else {

cout << this->name << " is better than " << sample.name << endl;

return false;

}

}

B issamename(Employee& sample) {

if (this->Employee.name == sample.name) { return true; }

else return false;

}

EAS calculation() { if (sizeof(EASs) == 4)return round(this->EASs / 10); else cout<<"Your score has to be a whole number"; }

};

int main(){

Transaction<string, int,int> t1("id1234", "id5678", 2804, 5000);

t1.getInfo();

cout << endl;

t1.isvalid("id1234","id5678");

Transaction<string, int, int> t2("id2680345", "id343040", 1034, 4054);

cout << endl<<t1.sumc(t2)<<endl;

cout << "\nsumc method with using double data type :\n";

Transaction<string, int, double> t3("id1236", "id5670", 2804, 5045.8);

Transaction<string, int, double>t4("id2680345", "id343040", 1034, 5679.5);

cout << endl << t3.sumc(t4) << endl;

cout << endl << endl << "||||||arithmetical class :|||||| \n";

calculations<double, int> cs(4.3, 2);

cs.showvalues();

cout << "subtraction :" << cs.Subt() << endl <<"Multiplication :" << cs.MP() << endl;

calculations <int, int> cr(5, 9);

cout << "\nObject 2 :\n";

cr.showvalues();

cout << "subtraction :" << cr.Subt() << endl << "Multiplication :" << cr.MP() << endl;

cout <<endl<< "||||||||||||||Employee class :|||||||||||||| \n";

Employee<double, char,bool> Andrusha(34.5, 'A', "Andre"); Andrusha.Getinfo();

Employee<double, char,bool> Dandy(32.1, 'D', "Dandy"); Dandy.Getinfo();

cout << "IS BETTER METHOD : " << Dandy.isbetter(Andrusha)<<endl;

cout << "EasCalulation METHOD : " << Dandy.calculation()<<endl<<endl;

cout << "||||||||||||||||||| now working with integer numbers:||||||||||| \n";

Employee<int, char, bool> AndreHudick(34, 'R', "Hudick"); AndreHudick.Getinfo();

Employee<int, char, bool> Nikita(97, 'C', "Nikita"); Nikita.Getinfo();

cout << "IS BETTER METHOD : " << AndreHudick.isbetter(Nikita) << endl;

cout << "EasCalulation METHOD : " << Nikita.calculation() << endl << endl;

return 713;

}

Результат работы программы :

From: id1234

To: id5678

Sum: 5000

Code: 2804

Transaction is valid

9054

sumc method with using double data type :

10725.3

||||||arithmetical class :||||||

a : 4.3

b : 2

subtraction :2.3

Multiplication :8.6

Object 2 :

a : 5

b : 9

subtraction :4

Multiplication :45

||||||||||||||Employee class :||||||||||||||

name :Andre

Qualification :A

EAS : 34.5

name :Dandy

Qualification :D

EAS : 32.1

Dandy is better than Andre

IS BETTER METHOD : 0

Your score has to be a whole numberEasCalulation METHOD : 0

||||||||||||||||||| now working with integer numbers:|||||||||||

name :Hudick

Qualification :R

EAS : 34

name :Nikita

Qualification :C

EAS : 97

Hudick is better than Nikita

IS BETTER METHOD : 0

EasCalulation METHOD : 9

Вывод:получил навыки работы с шаблонными классами на языке C++