10#include <iostream>

#include <vector>

using namespace std;

class Contract {

unsigned int nrContract{};

unsigned int anul{};

string beneficiar;

string furnizor;

float valoare{};

public:

Contract() = default;

~Contract() = default;

void setNrContract(unsigned int aux) {this->nrContract = aux;}

void setAn(unsigned int aux) {this->anul = aux;}

void setBeneficiar(string& aux) {this->beneficiar = aux;}

void setFurnizor(string& aux) {this->furnizor = aux;}

void setValoare(float aux) {this->valoare = aux;}

unsigned int getNrContract() const {return this->nrContract;}

unsigned int getAn() const {return this->anul;}

string getBeneficiar() {return this->beneficiar;}

string getFurnizor() {return this->furnizor;}

float getValoare() const {return this->valoare;}

};

class ContractInchiriere : protected Contract {

unsigned int perioada{};

// am presupus ca perioada este un numar natural

// adica, nu putem avea de pilda: 2,5 luni

public:

ContractInchiriere() = default;

~ContractInchiriere() = default;

void citireContractInchiriere() {

unsigned int nrContract\_;

unsigned int anul\_;

string beneficiar\_;

string furnizor\_;

float valoare\_;

cout << "Dati nr contractului:";

cin >> nrContract\_;

this->setNrContract(nrContract\_);

cout << "Dati anul:";

cin >> anul\_;

this->setAn(anul\_);

cout << "Dati beneficiarul:";

cin >> beneficiar\_;

this->setBeneficiar(beneficiar\_);

cout << "Dati furnizorul:";

cin >> furnizor\_;

this->setFurnizor(furnizor\_);

cout << "Dati valoarea:";

cin >> valoare\_;

this->setValoare(valoare\_);

cout << "Dati perioada (nr luni):";

cin >> this->perioada;

}

void afisareContractInchiriere() {

cout << "Nr contractului: " << this->getNrContract() << "\n";

cout << "Anul:" << this->getAn() << "\n";

cout << "Beneficiarul:" << this->getBeneficiar() << "\n";

cout << "Furnizorul:" << this->getFurnizor() << "\n";

cout << "Valoarea:" << this->getValoare() << "\n";

cout << "Perioada (nr luni):" << this->perioada << "\n";

}

float valoare() {

return this->getValoare() \* this->perioada;

}

};

class Dosar {

vector<ContractInchiriere\*> contracteInchiriere;

unsigned int nrContracteInchiriere{};

public:

Dosar() = default;

~Dosar() {

if(!this->contracteInchiriere.empty()) {

for(auto& i : this->contracteInchiriere)

delete i;

}

}

void citireContracteInchiriere() {

cout << "Dati numarul de contracte:";

cin >> this->nrContracteInchiriere;

for(int i = 0; i < this->nrContracteInchiriere; i++) {

auto\* aux = new ContractInchiriere;

aux->citireContractInchiriere();

this->contracteInchiriere.push\_back(aux);

}

}

void afisareContracteInchiriere() {

cout << "Contracte de inchiriere:\n";

for(int i = 0; i < this->nrContracteInchiriere; i++) {

cout << "Contractul " << i+1 << ". \n";

this->contracteInchiriere[i]->afisareContractInchiriere();

}

}

void valoareContractIndividual() {

for(int i = 0; i < this->nrContracteInchiriere; i++)

cout << "Valoarea contractului " << i+1 << " este: " << this->contracteInchiriere[i]->valoare() << "\n";

}

void valoareTotalaContracte() {

float sum = 0;

for(int i = 0; i < this->nrContracteInchiriere; i++)

sum += this->contracteInchiriere[i]->valoare();

cout << "Valoarea toatala a contractelor este: " << sum << "\n";

}

};

int main() {

auto \*dosar = new Dosar;

dosar->citireContracteInchiriere();

dosar->afisareContracteInchiriere();

dosar->valoareContractIndividual();

dosar->valoareTotalaContracte();

delete dosar;

return 0;

}