

Приклад лабораторної роботи №2

Автор : Рибалко Д.

Створимо клас Train

Інтерфейсна частина буде розташовуватися в файлі Train.h

```
////////Train.h////////
#ifndef TRAIN_H
#define TRAIN_H
#include "string.h"
class Train
{
    int platskart,kupe,vip,number;
    char punkt[20];
    float time;
public:
    Train();
    Train(int platskartN, int kupeN, int vipN, int numberN, char * punktN, float timeN);
    void setPlatskart(int platskart){this->platskart=platskart;}
    void setKupe(int kupe){this->kupe=kupe;}
    void setVip(int vip){this->vip=vip;}
    void setNumber (int number){this->number=number;}
    void setPunkt (char *punkt){strcpy(this->punkt,punkt);}
    void setTime (float time){this->time=time;}
    int getPlatskart(){return platskart;}
    int getKupe(){return kupe;}
    int getVip(){return vip;}
    int getNumber (){return number;}
    char*getPunkt (){return punkt;}
    float getTime (){return time;}

    void enterDannieKlav();
    void showDanie();
    int amountMest();
    virtual ~Train();
protected:
private:
};
#endif // TRAIN_H
```

Описова частина в файлі Train.cpp

```
////////Train.cpp////////
#include <iostream>
#include <stdio.h>
#include <windows.h>
#include "Train.h"
using namespace std;
Train::Train()
{
    strcpy(punkt,"");
    platskart=0,kupe=0,vip=0,number=0,time=0;
}
Train::~Train()
{
    //dtor
}
Train::Train(int platskartN, int kupeN, int vipN, int numberN,
char * punktN, float timeN)
{
    strcpy(punkt,punktN);
    platskart=platskartN;kupe=kupeN;vip=vipN;number=numberN;time=timeN;
```

```

}
void Train::enterDannieKlav()
{
    Train temp;
    getchar();
    cout << "Пункт назначения: ";
    gets(temp.punkt);
    setPunkt(temp.punkt);
    cout << "Время прибытия: ";
    cin >> temp.time;
    setTime(temp.time);
    cout << "Ввести номер поезда: ";
    cin >> temp.number;
    setNumber(temp.number);
    cout << "Кол.-во мест плацкарт: ";
    cin >> temp.platskart;
    setPlatskart(temp.platskart);
    cout << "Кол.-во мест купе: ";
    cin >> temp.kupe;
    setKupe(temp.kupe);
    cout << "Кол.-во мест люкс: ";
    cin >> temp.vip;
    setVip(temp.vip);
}

void Train::showDanie()
{
    cout << "Пункт назначения:   "<< this->punkt << endl;
    cout << "Время прибытия:      "<< this->time << endl;
    cout << "Номер поезда:           "<< this->number << endl;
    cout << "Кол.-во мест плацкарт: "<< this->platskart << endl;
    cout << "Кол.-во мест купе:      "<< this->kupe << endl;
    cout << "Кол.-во мест люкс:     "<< this->vip << endl;
}

int Train::amountMest()
{
    int amount=0;
    amount=getPlatskart()+getKupe()+getVip();
    return amount;
}

```

Нам знадобиться ще один клас, методи якого вмiють створювати масиви об'єктiв типу <Train>, назвемо його TrainArray

```

#ifndef TRAINARRAY_H
#define TRAINARRAY_H
#include "Train.h"
class TrainArray
{
private:
    Train *sostav;
    int kolvo;
public:
    TrainArray();
    void setKolvo(int n){kolvo=n;}
    int getKolvo(){return kolvo;}
    void creatArrayTrain(int n);
    virtual ~TrainArray();
    void enterArrayTrain();
    void showsArrayTrain();
    void fileWrite(char* txt);
    int fileOfSize(char *txt);
    void fileRead(char *txt);
}

```

```

        void zadanie1();
        void zadanie2();
        void zadanie3();
    protected:

};

#endif // TRAINARRAY_H

#include <fstream>
#include <stdlib.h>
#include <iostream>
#include "Train.h"
#include "TrainArray.h"

using namespace std;
TrainArray::TrainArray()
{
    //ctor
}
void TrainArray::creatArrayTrain(int n)
{
    sostav=new Train[n];
}
TrainArray::~TrainArray()
{
    //dtor
}
void TrainArray::enterArrayTrain()
{
    for(int i=0;i<getKolvo();i++)
    {
        sostav[i].enterDannieKlav();
    }
}

void TrainArray::showsArrayTrain(){
    system("CLS");
    for(int i=0;i<getKolvo();i++)
    {
        sostav[i].showDanie();
    }
}

void TrainArray::fileWrite(char* txt)
{
    ofstream t(txt,ios::binary | ios::app);
    for(int i=0;i<getKolvo();i++)
    {
        t.write((char*)&sostav[i], sizeof sostav[i]);
    }
    t.close();
}

int TrainArray::fileOfSize (char *txt) //определение количества поездов в файле
{
    ifstream t(txt,ios::binary);
    t.seekg(0,ios_base::end);
    int k = t.tellg()/sizeof (Train);
    t.close();
    return k;
}

void TrainArray::fileRead(char *txt) //чтение из файла
{

```

```

ifstream t(txt,ios::binary);
for(int i=0;i<kolvo;i++)
{
    t.read((char*)&sostav[i], sizeof sostav[i]);
}
t.close();
}

```

```

void TrainArray::zadanie1()
{
    getchar();
    cout << "Задание 1" << endl;
    cout << "Введите город: ";
    char town[20];
    gets(town);
    bool flag=0;
    for(int i=0;i<kolvo;i++)
    {
        if (!strcmpi(sostav[i].getPunkt(),town))
        {
            sostav[i].showDanie();
            flag=1;
        }
    }
    if(!flag) cout << "Не найдено" << endl;
}

```

```

void TrainArray::zadanie2()
{
    cout << "Задание 2" << endl;
    cout << "Введите город: ";
    char town[20];
    double time2;
    gets(town);
    bool flag=0;
    cout << "Введите время: ";
    cin >> time2;
    for(int i=0;i<kolvo;i++)
    {
        if (!strcmpi(sostav[i].getPunkt(),town) && sostav[i].getTime()==time2){
            sostav[i].showDanie();
            flag=1;}
    }
    if(!flag) cout << "Не найдено" << endl;
}

```

```

void TrainArray::zadanie3()
{
    getchar();
    cout << "Задание 3" << endl;
    cout << "Введите город: ";
    char town[20];
    int obshmesta;
    bool flag=0;
    gets(town);
    cout << "Введите кол.- во мест: ";
    cin >> obshmesta;
    for (int i=0;i<kolvo;i++)
    {
        if (!strcmpi(sostav[i].getPunkt(),town) && sostav->amountMest()== obshmesta)
        {
            sostav[i].showDanie();
            flag=1;}
    }
}

```

```
    if(!flag) cout << "Не найдено"<< endl;
}
```

І все це починає працювати, коли ми створимо в головній функції main об'єкт класу TrainArray

```
//////////main.cpp//////////
```

```
#include <iostream>
#include <stdio.h>
#include <windows.h>
#include <fstream>
#include "Train.h"
#include "TrainArray.h"
using namespace std;
```

```
int main()
```

```
{
    int n,kolvo;
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    char txt[20];
    TrainArray vagons;
    system("CLS");
    cout << "\t< Меню >"<< endl;
    cout << "Запись в файл - 1, чтени с файла - 2"<< endl;
    cout << "Введите номер: ";
    cin >> n;
    system("CLS");
    switch (n)
    {
    case 1:
    {
        getchar();
        cout << " --Запись в файл--" << endl;
        cout << "Название файла: ";
        gets(txt);
        cout<<"Кол.-во поездов: ";
        cin >> kolvo;

        vagons.setKolvo(kolvo);
        vagons.creatArrayTrain(kolvo);
        vagons.enterArrayTrain();
        vagons.showsArrayTrain();
        vagons.fileWrite(txt);

        break;
    }
    default :
```

```
{
  getchar();
  cout << " **Чтение с файла**" << endl;
  cout << "Название файла: ";
  gets(txt);
  kolvo=vagons.fileOfSize(txt);
  cout << "Количество поездов: " << kolvo << endl;

  vagons.setKolvo(kolvo);
  vagons.creatArrayTrain(kolvo);
  vagons.fileRead(txt);
  vagons.showsArrayTrain();

}
}
vagons.zadanie1();
vagons.zadanie2();
vagons.zadanie3();
return 0;
}
```