

4IT Erasmus+Project 2024-1-RO01-KA121-VET-000214264
Beneficiary Colegiul National Gheorghe Vranceanu, Bacau, Romania

Sending organization: Colegiul National ”Roman-Vodă” Roman, Romania

MOBILITY PORTFOLIO
Paphos, Cyprus

Mobility period:
PAPHOS, CYPRUS, 13.04.2025-27.04.2025 – practice period 14.04.2025-25.04.2025

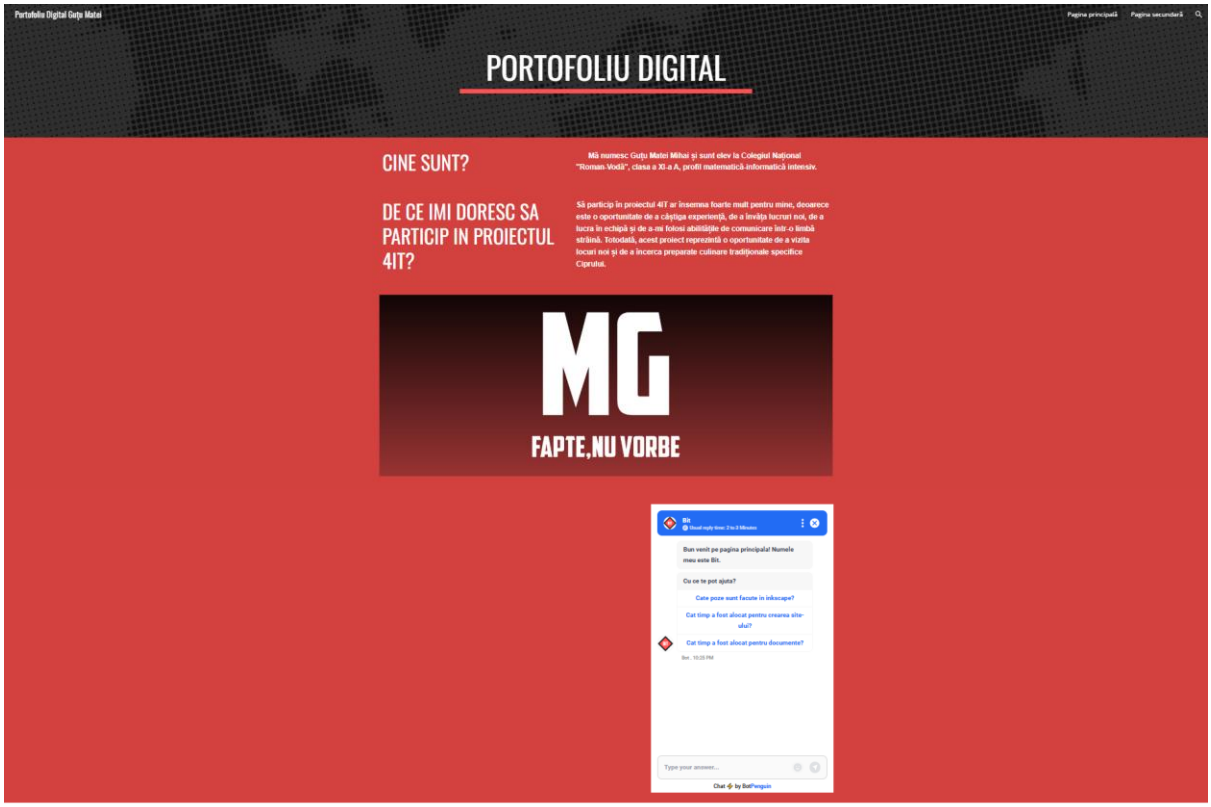
Receiving

Practice tutor: Lazaros Lazarou

Participant: Guțu Matei Mihai

DISCLAIMER: The European Commission’s support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Portofoliu Digital Guțu Matei - site selecție

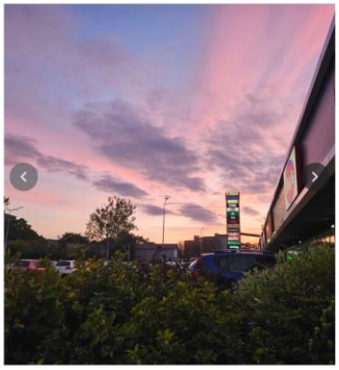




Activate Windows
Go to Settings to activate Windows.



Activate Windows



Activate Windows
Go to Settings to activate Windows.

Guțu Matei - prezentare pregătire

Antreprenori de succes care aplică principii ecologice

Elon Musk

Elon Musk este unul dintre cei mai cunoscuți și influenți antreprenori din lume.
În 2004 a investit în Tesla Motors, lansând in 2008 modelul Roadster, urmat de Model S, Model X, Model 3 si Cybertruck. Sub conducerea lui Musk, Tesla a devenit un pionier în domeniul energiei regenerabile și al tehnologiei bateriilor, cu produse precum Powerwall și Solar Roof

Boyan Slat

Un alt antreprenor de succes este Boyan Slat. Acesta făcea scufundări când a realizat că în apă era mai mult plastic decât pești. Acest lucru l-a motivat foarte tare să curețe oceanele, în 2012 abandonându-și studiile pentru a-și dezvolta ideea. În 2013 a creat “The Ocean Cleanup”, o organizație non-profit, reușind să strângă 2.2 milioane de dolari. Până în prezent, compania a reușit să scoată 21.000.000 kg de gunoi din oceane și râuri.

Antreprenori de succes care aplică principii ecologice

Elon Musk



Boyan Slat



Produse finale

```cpp
#include <iostream>
#include <string>
#include <stdlib.h>
using namespace std;
/*
-----ASSIGNMENT-----
program that uses classes and functions
functions
-add data
-show data
-search by ...(4 in total)
-exit
*/
class cars {
private :
    string brand,model,platenumber,owner;
public :
    cars() {
        brand="EMPTY";
        model="EMPTY";
        platenumber="EMPTY";
        owner="EMPTY";
    }
    void setbrand(string tempbrand) {
        brand=tempbrand;
    }
    void setmodel(string tempmodel) {
        model=tempmodel;
    }
    void setplatenumber(string tempplatenumber) {
        platenumber=tempplatenumber;
    }
    void setowner(string tempowner) {
        owner=tempowner;
    }

    string getbrand() {
        return brand;
    }
    string getmodel() {
        return model;
    }
    string getplatenumber() {
        return platenumber;
    }
```

```
        return platenumber;
    }
    string getowner() {
        return owner;
    }
}*garage,*temp;
int arraysize;

void adddata() {
    int n;
    do {
        string tempbrand,tempmodel,tempplatenumber1,tempplatenumber2,tempowner;
        cout<<"How many cars would you like to add or 0 to exit : ";
        n=arraysize;
        int tmpN;
        cin>>tmpN;
        if(tmpN<1)return;
        arraysize+=tmpN;
        temp=new cars[arraysize];
        for(int i=0; i<n; i++)temp[i]=garage[i];
        for(int i=n; i<arraysize; i++) {
            cout<<"Brand : ";
            cin>>tempbrand;
            cout<<'\n';
            cout<<"Model : ";
            cin>>tempmodel;
            cout<<'\n';
            cout<<"Plate number : ";
            cin>>tempplatenumber1>>tempplatenumber2;
            cout<<'\n';
            cout<<"Owner : ";
            cin>>tempowner;
            cout<<'\n';
            tempplatenumber1+=" "+tempplatenumber2;
            temp[i].setbrand(tempbrand);
            temp[i].setmodel(tempmodel);
            temp[i].setplatenumber(tempplatenumber1);
            temp[i].setowner(tempowner);
        }
        delete [] garage;
        garage=temp;
    } while(n<arraysize);
    system("clear");
}

```

```
88
89  void displaydata() {
90      for(int i=0; i<arraysize; i++) {
91          cout<<"----------CAR "<<i+1<<"----------"<<'\n';
92          cout<<"Brand : "<<garage[i].getbrand()<<'\n';
93          cout<<"Model : "<<garage[i].getmodel()<<'\n';
94          cout<<"Plate number : "<<garage[i].getplatenumber()<<'\n';
95          cout<<"Owner : "<<garage[i].getowner()<<'\n';
96          cout<<'\n';
97      }
98      cout<<"Press any key to continue ";
99      string a;
100     cin>>a;
101     system("clear");
102 }
103
104 void searchbybrand(string tempbrand) {
105     for(int i=0; i<arraysize; i++) {
106         if(garage[i].getbrand()==tempbrand) {
107             cout<<"Car found !"<<'\n';
108             cout<<"----------CAR "<<i+1<<"----------"<<'\n';
109             cout<<"Brand : "<<garage[i].getbrand()<<'\n';
110             cout<<"Model : "<<garage[i].getmodel()<<'\n';
111             cout<<"Plate number : "<<garage[i].getplatenumber()<<'\n';
112             cout<<"Owner : "<<garage[i].getowner()<<'\n';
113             cout<<'\n';
114             cout<<"Press any key to continue ";
115             string a;
116             cin>>a;
117             system("clear");
118             return;
119         }
120     }
121     cout<<"Car not found !!"<<'\n';
122     cout<<"Press any key to continue ";
123     string a;
124     cin>>a;
125     system("clear");
126 }
127
128 void searchbymodel(string tempmodel) {
129     for(int i=0; i<arraysize; i++) {
130         if(garage[i].getmodel()==tempmodel) {
131             cout<<"Car found !"<<'\n';
132             cout<<"----------CAR "<<i+1<<"----------"<<'\n';
```

```
            cout<<"----------CAR "<<i+1<<"----------"<<'\n';
            cout<<"Brand : "<<garage[i].getbrand()<<'\n';
            cout<<"Model : "<<garage[i].getmodel()<<'\n';
            cout<<"Plate number : "<<garage[i].getplatenumber()<<'\n';
            cout<<"Owner : "<<garage[i].getowner()<<'\n';
            cout<<'\n';
            cout<<"Press any key to continue ";
            string a;
            cin>>a;
            system("clear");
            return;
        }
    }
    cout<<"Car not found !!"<<'\n';
    cout<<"Press any key to continue ";
    string a;
    cin>>a;
    system("clear");
}

void searchbyplatenumber(string tempplatenumber) {
    for(int i=0; i<arraysize; i++) {
        if(garage[i].getplatenumber()==tempplatenumber) {
            cout<<"Car found !"<<'\n';
            cout<<"----------CAR "<<i+1<<"----------"<<'\n';
            cout<<"Brand : "<<garage[i].getbrand()<<'\n';
            cout<<"Model : "<<garage[i].getmodel()<<'\n';
            cout<<"Plate number : "<<garage[i].getplatenumber()<<'\n';
            cout<<"Owner : "<<garage[i].getowner()<<'\n';
            cout<<'\n';
            cout<<"Press any key to continue ";
            string a;
            cin>>a;
            system("clear");
            return;
        }
    }
    cout<<"Car not found !!"<<'\n';
    cout<<"Press any key to continue ";
    string a;
    cin>>a;
    system("clear");
}

void searchbyowner(string tempowner) {
```

```
176 void searchbyowner(string tempowner) {
177     for(int i=0; i<arraysize; i++) {
178         if(garage[i].getowner()==tempowner) {
179             cout<<"Car found !"<<'\n';
180             cout<<"----------CAR "<<i+1<<"----------"<<'\n';
181             cout<<"Brand : "<<garage[i].getbrand()<<'\n';
182             cout<<"Model : "<<garage[i].getmodel()<<'\n';
183             cout<<"Plate number : "<<garage[i].getplatenumber()<<'\n';
184             cout<<"Owner : "<<garage[i].getowner()<<'\n';
185             cout<<'\n';
186             cout<<"Press any key to continue ";
187             string a;
188             cin>>a;
189             system("clear");
190             return;
191         }
192     }
193     cout<<"Car not found !!"<<'\n';
194     cout<<"Press any key to continue ";
195     string a;
196     cin>>a;
197     system("clear");
198 }
199 
200 void menu() {
201     int ok;
202     string tempbrand,tempmodel,tempplatenumber1,tempplatenumber2,tempowner;
203     do {
204         cout<<"What would you like to do : "<<'\n';
205         cout<<"1.Add data"<<'\n';
206         cout<<"2.Display data"<<'\n';
207         cout<<"3.Search by brand"<<'\n';
208         cout<<"4.Search by model"<<'\n';
209         cout<<"5.Search by plate number"<<'\n';
210         cout<<"6.Search by owner"<<'\n';
211         cout<<"0.Exit"<<'\n';
212         cin>>ok;
213         system("clear");
214 
215         switch(ok) {
216         case 1: {
217             adddata();
218             break;
219         }
220         case 2: {
```

```
        }
        case 2: {
            displaydata();
            break;
        }
        case 3: {
            cout<<"Brand : ";
            cin>>tempbrand;
            searchbybrand(tempbrand);
            break;
        }
        case 4: {
            cout<<"Model : ";
            cin>>tempmodel;
            searchbymodel(tempmodel);
            break;
        }
        case 5: {
            cout<<"Plate number : ";
            cin>>tempplatenumber1>>tempplatenumber2;
            tempplatenumber1+=" "+tempplatenumber2;
            searchbyplatenumber(tempplatenumber1);
            break;
        }
        case 6: {
            cout<<"Owner : ";
            cin>>tempowner;
            searchbyowner(tempowner);
            break;
        }
        case 0: {
            return;
        }
        }
    } while(ok);
}
int main() {
    menu();

    return 0;
}
```

```cpp
#include <iostream>
using namespace std;


//CREATE THE CLASS
class car{
    private:
        string brand;
        string model;
    public:

        car(){
            brand="EMPTY";
            model="EMPTY";

        }
        void setBrand(string B){
            brand=B;
        }
        void setModel(string M){
            model=M;
        }
        string getBrand(){
            return brand;
        }
        string getModel(){
            return model;
        }
};

car *readData(car *, int & ,int &);

void displaydata(car *carsArray,int arrSize){
    for(int i=0;i<arrSize;i++){
        cout<<"---------- CAR "<<i+1<<" ----------"<<'\n';
        cout<<"Brand : "<<carsArray[i].getBrand()<<'\n';
        cout<<"Model : "<<carsArray[i].getModel()<<'\n';
        cout<<"\n\n";
    }
}


void searchbybrand(car *carsArray,int arrSize){
    string tempbrand;
    cout<<"Brand : ";
```

```
    string tempbrand;
    cout<<"Brand : ";
    cin>>tempbrand;
    for(int i=0;i<arrSize;i++)if(tempbrand==carsArray[i].getBrand()){
        cout<<"Car found !!"<<'\n';
        cout<<"Brand : "<<tempbrand<<'\n';
        cout<<"Model : "<<carsArray[i].getModel();
        cout<<"\n\n";
        return;
    }
    cout<<"Car not found... \n\n";
}
int main(){

    int ARRAYSIZE=0;
    int nextAvailPos=0;
    car *carsArray;
    carsArray=new car[ARRAYSIZE];
    int choice=-1;

    do{
        cout<<"Please Choose from the menu below: "<<endl;
        cout<<"1. Enter Data"<<endl;
        cout<<"2. Display Data"<<endl;
        cout<<"3. Search By Brand"<<endl;
        cout<<"0. Exit"<<endl;
        cin>>choice;

        switch(choice){
            case 1:
                carsArray=readData(carsArray,ARRAYSIZE, nextAvailPos);
                break;
            case 2: displaydata(carsArray,ARRAYSIZE);
                break;
            case 3:searchbybrand(carsArray,ARRAYSIZE);
                break;
            case 0:
                break;


        }


    }while (choice!=0);

```

```
88
89      return 0;
90  }
91  //function car *readData(car *, int)
92  //the function populates the array with information
93  //from the user
94  //the function accepts a pointer to a car object(the array)\
95  //and the size of the array
96  //the function return back a pointer to aa car object which is the array
97  //and is now populated with data.
98  car *readData(car *arr, int &arrSize,int &nextPos){
99      int numcars=0, //how many cars the user is going to input
100     newSize=0; //the new size of the array which is calculated by adding arraySize+numCars
101     string tstring; //temporary string to read keyboard input
102     car *tmp; //temporary car object to copy everything from the car array nad grow the array;
103
104     cout<<"How many cars are you entering: ";
105     cin>>numcars; //read from the keyboard how many cars the user needs to input
106     for (int i=0;i<numcars;i++){//repeat the reading process 'numCars' times
107
108         if( nextPos>=arrSize){ //first check if the array is full
109            newSize=arrSize+numcars;//if you got here the array is full calculate the new size
110            tmp=new car[newSize]; //allocate temporary space for the new bigger array
111            for (int j=0;j<arrSize;j++){//copy everything from the small array to the new bigger one
112                tmp[j]=arr[j];
113             }
114             arrSize=newSize; //save the newsize into arrSize (update the arrSize).
115
116             delete []arr; //release the memory from the old array back to the OS
117             arr=tmp; //set the pointer of the array to point to the new allocated space
118
119         }
120         cout<<"Enter BRAND for car["<<nextPos<<"]";
121         cin>>tstring;
122         arr[nextPos].setBrand(tstring);
123         cout<<"Enter MODEL for car["<<nextPos<<"]";
124         cin>>tstring;
125         arr[nextPos].setModel(tstring);
126         nextPos++;
127
128     }
129     return arr;
130 }
```