

4IT Erasmus+Project, 2025-1-RO01-KA121-VET-000319082

Beneficiary:"Gheorghe Vranceanu" National College, Bacau

Sending organization: "Costache Negri" National College Targu Ocna

MOBILITY PORTFOLIO
Paphos, Cyprus

Mobility period: 01.04.2026-14.04.2026

Practice period 02.04.2026-13.04.2026

RECEIVING COMPANY: LAZAROU ICT PROFESSIONALS

Practice tutor: Lazaros Lazarou

Participant: Olaru Sara

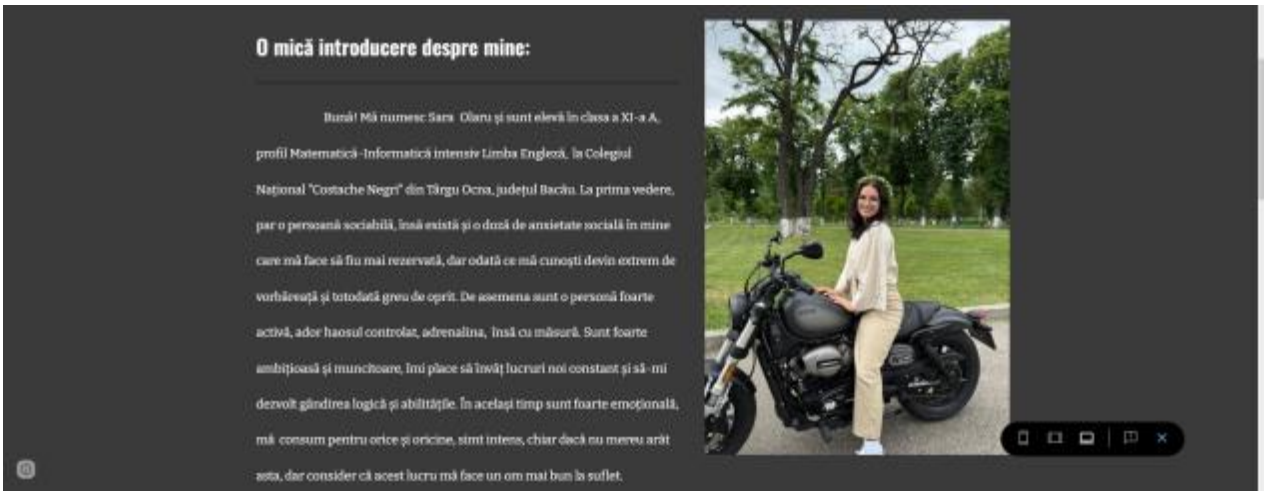
DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

2025-1-RO01-KA121-VET-000319082

SECTIUNEA I

➢ Pagina principală





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.

2025-1-RO01-KA121-VET-000319082





- Motto



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.

2025-1-RO01-KA121-VET-000319082

- Logo





- Un argument



- Un chatbot



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.

- Un avatar/gif asociat



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.

2025-1-RO01-KA121-VET-000319082

- **Pagina care reflectă personalitatea, pasiunile, aspirațiile mele**





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.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082

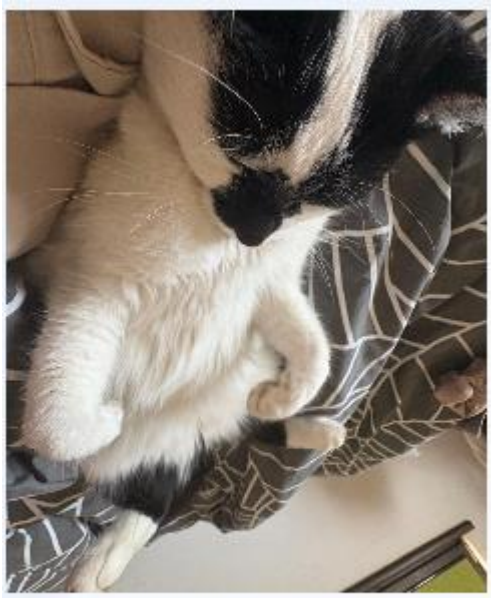




DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

➢ Galerie interactivă de imagini originale





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.

2025-1-RO01-KA121-VET-000319082

Bucătăria tradițională din Cipru

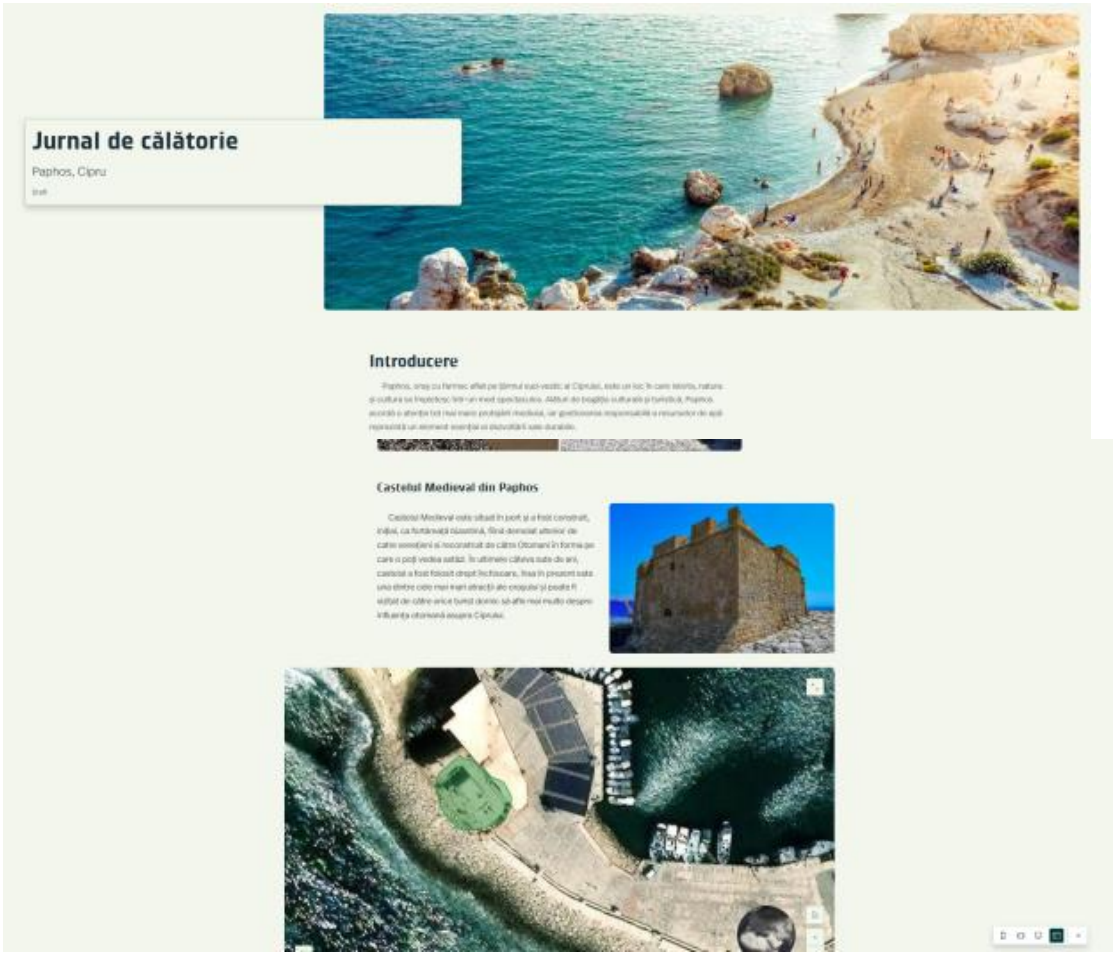




DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

2025-1-RO01-KA121-VET-000319082

➢ Proiect Pregătire culturală și Green Society



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.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082





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.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082

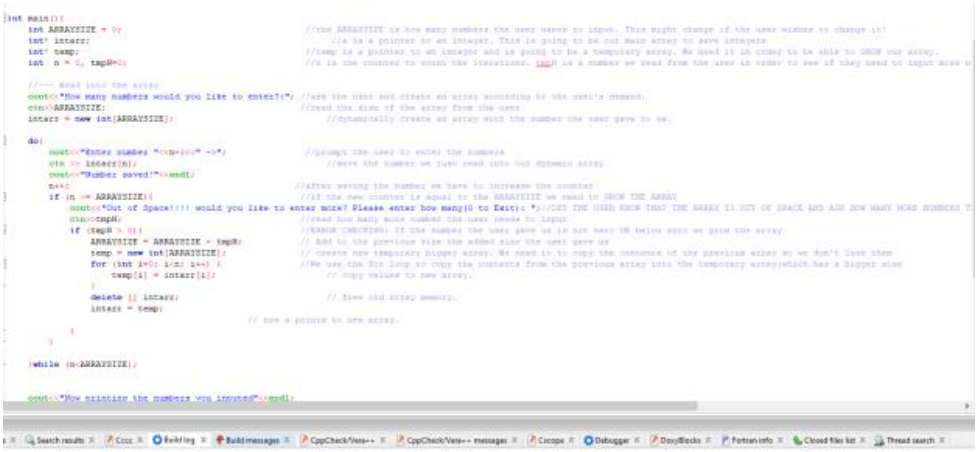


DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082

SECTIUNEA II

➢ Practică în C++





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.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082

```
struct cars
{
string name;
int phoneNumber;
string plateNumber;
string brand;
};
int main(){
    int ARRAYSIZE = 0;
    cars* stArray;

cars* stTemp;


    int  n = 0, tmpN=0;
    cout<<"How many cars would you like to enter?:";
    cin>>ARRAYSIZE;
    stArray = new cars [ARRAYSIZE];


    do{
        cout<<"Enter owner's name "<<n+1<<" ->";
        cin >>stArray[n].name;
          cout<<"Enter owner's phone number "<<n+1<<" ->";
        cin >>stArray[n].phoneNumber;
        cout<<"Enter plate number "<<n+1<<" ->";
        cin>>stArray[n].plateNumber;
        cout<<"Enter brand of the car "<<n+1<<" ->";
        cin>>stArray[n].brand;
        cout<<"Information saved!"<<endl;
        n++;
        if (n >= ARRAYSIZE){
            cout<<"Out of Space!!!! would you like to enter more? Please enter how many(0 to Exit): "<<endl;
            cin>>tmpN;
            if (tmpN > 0){
                ARRAYSIZE = ARRAYSIZE + tmpN;
                stTemp = new cars[ARRAYSIZE];

                for (int i=0; i<n; i++) {
                    stTemp[i].name=stArray[i].name;
                     stTemp[i].phoneNumber=stArray[i].phoneNumber;
                      stTemp[i].plateNumber=stArray[i].plateNumber;
```



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.

➢ Practică în Arduino

```
DigitalReadSerial

Reads a digital input on pin 2, prints the result to the Serial Monitor

This example code is in the public domain.

https://docs.arduino.cc/built-in-examples/basics/DigitalReadSerial/
*/

// digital pin 2 has a pushbutton attached to it. Give it a name:
int pushButton = 2;

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  // make the pushbutton's pin an input:
  pinMode(pushButton, INPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input pin:
  int buttonState = digitalRead(pushButton);
  // print out the state of the button:
  Serial.println(buttonState);
  delay(1);  // delay in between reads for stability
}
```

Debounce.ino

```
43
44  void setup() {
45    pinMode(buttonPin, INPUT);
46    pinMode(ledPin, OUTPUT);
47
48    // set initial LED state
49    digitalWrite(ledPin, ledState);
50  }
51
52  void loop() {
53    // read the state of the switch into a local variable:
54
55    int reading = digitalRead(buttonPin);
56
57
58    // check to see if you just pressed the button
59    // (i.e. the input went from LOW to HIGH), and you've waited long enough
60    // since the last press to ignore any noise:
61
62    // If the switch changed, due to noise or pressing:
63    if (reading != lastButtonState) {
64      // reset the debouncing timer
65      lastDebounceTime = millis();
66    }
67
68    if ((millis() - lastDebounceTime) > debounceDelay) {
69      // whatever the reading is at, it's been there for longer than the debounce
70      // delay, so take it as the actual current state:
```

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.

Co-funded by the Erasmus+ Programme of the European Union
Erasmus+
2025-1-RO01-KA121-VET-000319082



```
Fade.ino

  The analogWrite() function uses PWM, so if you want to change the pin you're
  using, be sure to use another PWM capable pin. On most Arduino, the PWM pins
  are identified with a "~" sign, like ~3, ~5, ~6, ~9, ~10 and ~11.

  This example code is in the public domain.

  https://docs.arduino.cc/built-in-examples/basics/Fade/
*/

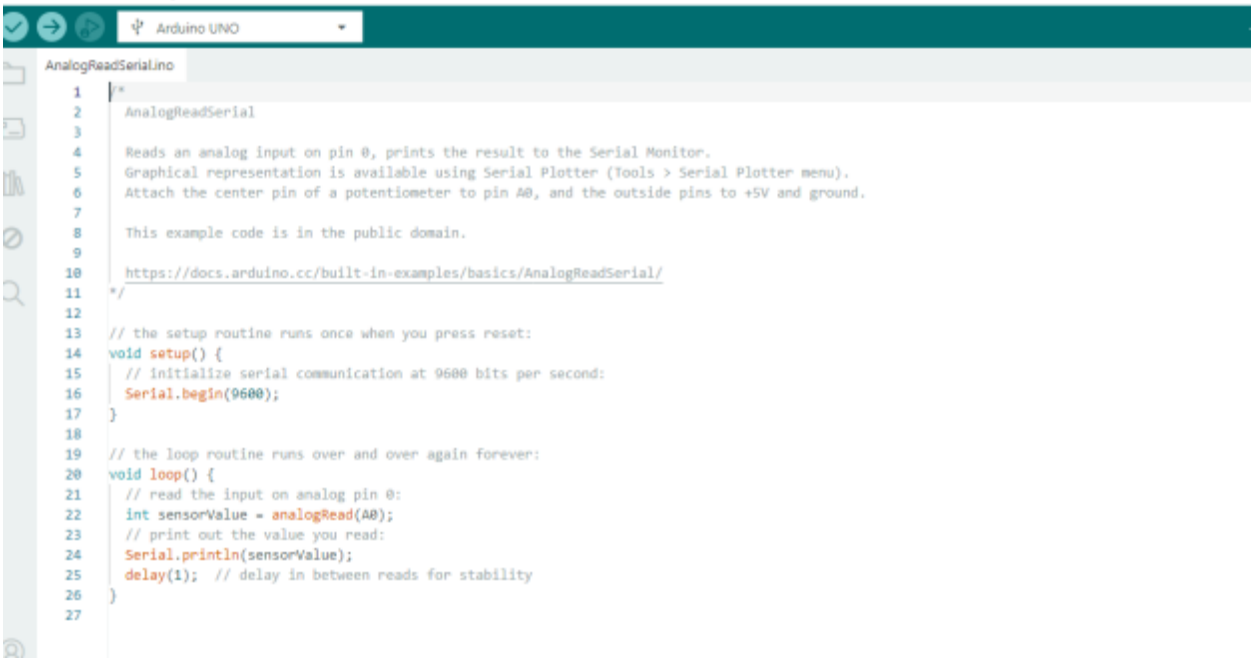
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```



Arduino UNO

AnalogReadSerial.ino

```
/*
  AnalogReadSerial

  Reads an analog input on pin 0, prints the result to the Serial Monitor.
  Graphical representation is available using Serial Plotter (Tools > Serial Plotter menu).
  Attach the center pin of a potentiometer to pin A0, and the outside pins to +5V and ground.

  This example code is in the public domain.

  https://docs.arduino.cc/built-in-examples/basics/AnalogReadSerial/
*/

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // print out the value you read:
  Serial.println(sensorValue);
  delay(1);  // delay in between reads for stability
}
```

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.