



Οδηγός Εκπαιδευτικού

Τεχν
ητή
Νοημο
σύνη

6

Δημιουργία εφαρμογής AI με το Arduino



Εισαγωγή στη δραστηριότητα

Σε αυτό το project θα φτιάξουμε μία εφαρμογή Arduino, που θα παίρνει μετρήσεις υγρασίας και θερμοκρασίας, και θα επικοινωνεί με το ChatGPT ώστε να λάβει και να προβάλει ένα σύντομο χαρακτηρισμό των καιρικών συνθηκών. Για το σκοπό αυτό, θα χρησιμοποιήσουμε την πλακέτα Arduino Uno R4 WiFi, η οποία διαθέτει δυνατότητα σύνδεσης στο Internet.

Απευθύνεται σε:

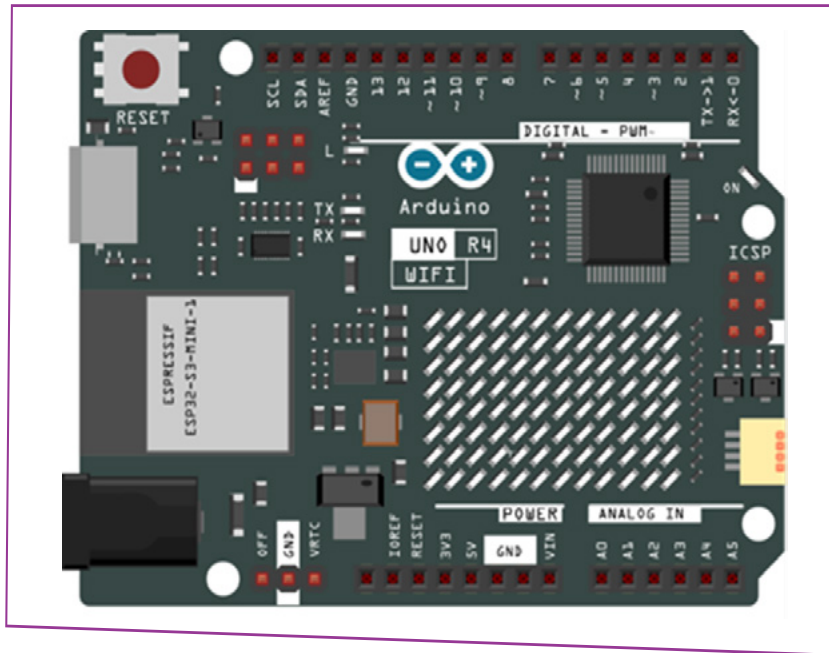
Εκπαιδευτικούς και κάθε ενδιαφερόμενο χρήστη με προηγούμενη εμπειρία στην τεχνητή νοημοσύνη και στο Arduino που θέλει να γνωρίσει πώς μπορεί να εντάξει την τεχνητή νοημοσύνη σε εφαρμογές Arduino

Στόχοι δραστηριότητας:

- Κατανόηση των βασικών αρχών της τεχνητής νοημοσύνης.
- Κατανόηση της σύνδεσης Arduino και AI.
- Δημιουργία AI εφαρμογής Arduino.

Εκτέλεση δραστηριότητας

Το Arduino Uno R4 WiFi

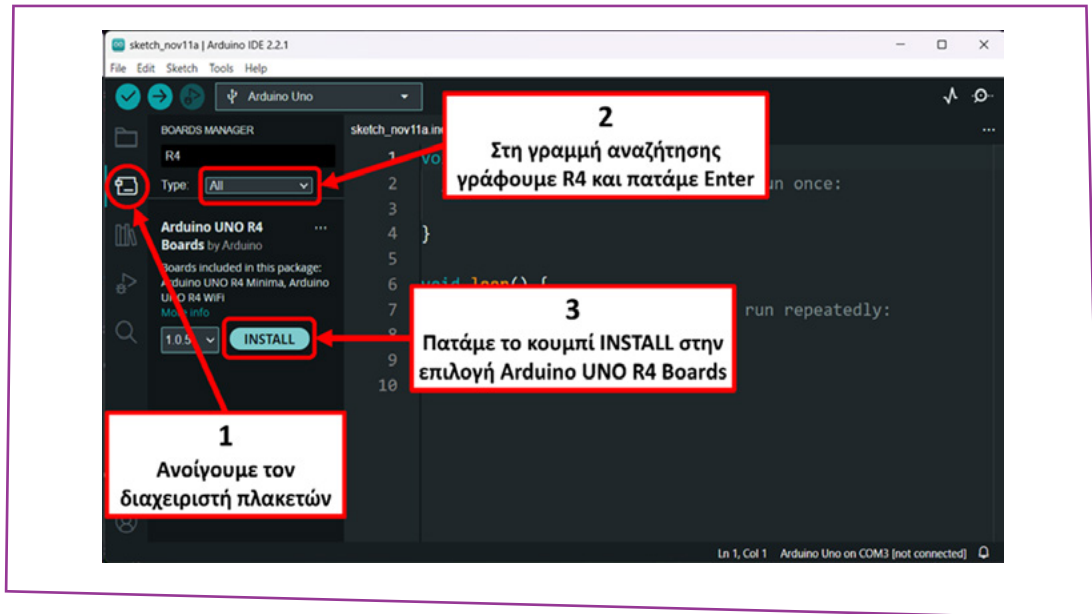


Εικόνα 1.

Το Arduino Uno R4 WiFi είναι μια νέα έκδοση πλακέτας. Έχει τις ίδιες διαστάσεις, την ίδια διάταξη ακροδεκτών και την ίδια τάση λειτουργίας (5V) με το Arduino Uno R3, αλλά διαθέτει αυξημένες δυνατότητες. Ο βασικός μικροελεγκτής της πλακέτας, ο RA4M1, είναι ισχυρότερος με περισσότερη μνήμη και αποθηκευτικό χώρο. Επιπλέον, η πλακέτα περιλαμβάνει έναν δεύτερο μικροελεγκτή ESP32-S3, ο οποίος παρέχει δυνατότητες ασύρματης συνδεσιμότητας Wi-Fi και BLE. Ένα ακόμα σημαντικό χαρακτηριστικό είναι το ενσωματωμένο LED Matrix, το οποίο αποτελείται από μια συστοιχία 12x8 LED.

Για να μπορέσουμε να προγραμματίσουμε την πλακέτα με το Arduino IDE θα πρέπει πρώτα να την προσθέσουμε μέσω του Boards Manager με τη διαδικασία που φαίνεται στην εικόνα.

Εκτέλεση δραστηριότητας

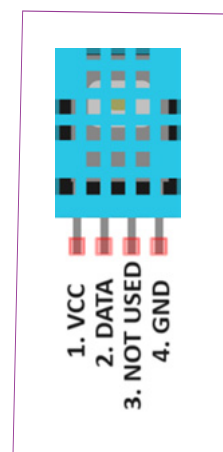


Εικόνα 2.

Ο Αισθητήρας DHT11

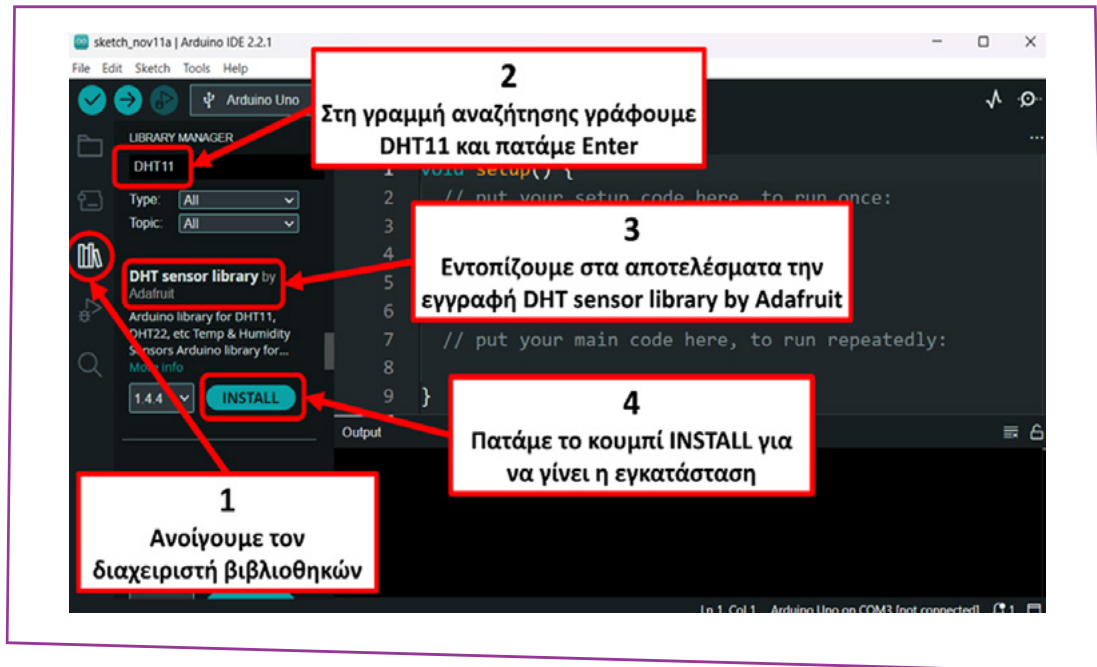
Ο DHT11 είναι ένας ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας. Διατίθεται σκέτος ή ως “module” τοποθετημένος πάνω σε ειδική πλακέτα. Όπως φαίνεται στην εικόνα, η βασική έκδοση διαθέτει τέσσερις ακροδέκτες. Συγκεκριμένα, οι ακροδέκτες 1 και 4 είναι οι VCC και GND και χρησιμοποιούνται για την τροφοδοσία του αισθητήρα. Ο ακροδέκτης 2 (DATA) είναι η έξοδος όπου ο αισθητήρας βγάζει τις μετρήσεις του. Ο ακροδέκτης 3 δεν χρησιμοποιείται. Ο αισθητήρας μετράει τη θερμοκρασία σε βαθμούς Κελσίου με ακρίβεια ενός βαθμού. Η υγρασία μετριέται ως ποσοστό επί τοις εκατό.

Ο DHT11 είναι αρκετά διαδεδομένος και έτσι υπάρχουν πολλές βιβλιοθήκες που υποστηρίζουν τη χρήση του. Εμείς θα χρησιμοποιήσουμε την DHT sensor library της εταιρείας Adafruit. Η προσθήκη της βιβλιοθήκης στο περιβάλλον του Arduino IDE γίνεται με το Library Manager.



Εικόνα 3.

Εκτέλεση δραστηριότητας



Εικόνα 4.

Κατά τη διάρκεια της εγκατάστασης, το Library Manager ζητάει την άδειά μας για να εγκαταστήσει και το Adafruit Unified Sensor, που είναι απαραίτητο για τη λειτουργία του DHT sensor library.

Εφαρμογή : Χαρακτηρισμός καιρικών συνθηκών με το ChatGPT

Το ChatGPT είναι ένα σύστημα τεχνητής νοημοσύνης της OpenAI, που χρησιμοποιεί αλγόριθμους επεξεργασίας φυσικής γλώσσας, για να μπορεί να επικοινωνεί με τους χρήστες του, με τρόπο που μοιάζει ανθρώπινος. Στην εφαρμογή μας θα δημιουργήσουμε έναν απλό μετεωρολογικό σταθμό με το Arduino R4 WiFi και τον αισθητήρα DHT11. Το Arduino θα μετράει τη θερμοκρασία και την υγρασία, και θα στέλνει τις μετρήσεις στο μοντέλο ChatGPT3.5-instruct, προκειμένου να λάβει έναν σύντομο λεκτικό χαρακτηρισμό των καιρικών συνθηκών. Στη συνέχεια θα προβάλλει στο ενσωματωμένο LED Matrix ένα μήνυμα, που θα περιλαμβάνει τις αρχικές μετρήσεις και τον λεκτικό χαρακτηρισμό.



Εκτέλεση δραστηριότητας

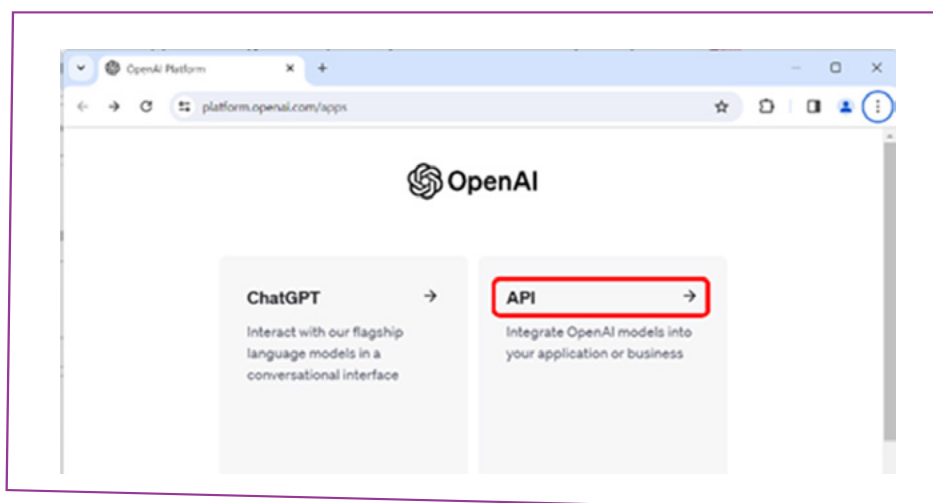
Δημιουργία API Key για την επικοινωνία με το ChatGPT

Η χρήση των υπηρεσιών του ChatGPT μέσα από το πρόγραμμα του Arduino, γίνεται με τη χρήση κατάλληλου API (Application Programming Interface). Η χρήση των API παρέχεται από την OpenAI ως συνδρομητική υπηρεσία και χρεώνεται με βάση τον όγκο των δεδομένων που ανταλλάσσονται. Ο όγκος των δεδομένων μετρείται σε tokens (ενδεικτικά μια αγγλική λέξη αντιστοιχεί κατά μέσο όρο σε 1.3 token ενώ για το μοντέλο gpt-3.5-turbo-instruct η χρέωση είναι 0.0015\$ ανά 1000 εισερχόμενα tokens και 0.0020\$ ανά 1000 εξερχόμενα). Με βάση την τρέχουσα τιμολογιακή πολιτική της OpenAI, σε κάθε νέος χρήστης της υπηρεσίας παρέχεται δωρεάν πίστωση 5 δολαρίων με διάρκεια 3 μηνών. Λεπτομερείς πληροφορίες για τη χρέωση της χρήσης των API μπορείτε να βρείτε στη διεύθυνση <https://openai.com/pricing>.

Για να μπορεί η εφαρμογή να επικοινωνεί με το μοντέλο ChatGPT, θα πρέπει μαζί με κάθε αίτημα να στέλνει και ένα API key, με βάση το οποίο θα γίνεται η πιστοποίηση και η χρέωση. Η έκδοση του API key γίνεται μέσα από την ιστοσελίδα της OpenAI:

1 Ανοίγουμε τη σελίδα <https://openai.com> και συνδεόμαστε πατώντας το link **Log in**. Η σύνδεση μπορεί να γίνει είτε με εξειδικευμένο λογαριασμό OpenAI, είτε χρησιμοποιώντας έναν λογαριασμό Google, Microsoft ή Apple.

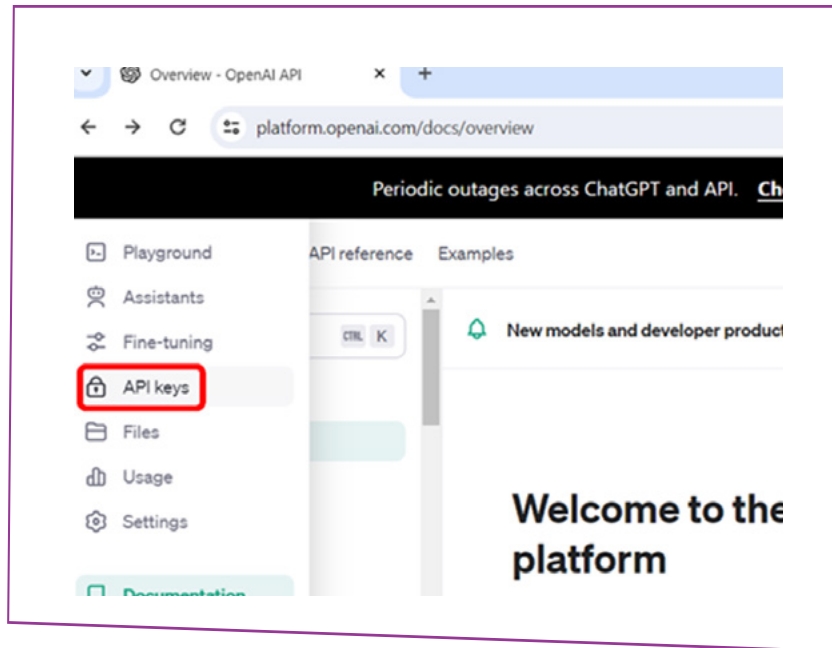
2 Στη νέα σελίδα επιλέγουμε API.



Εικόνα 5.

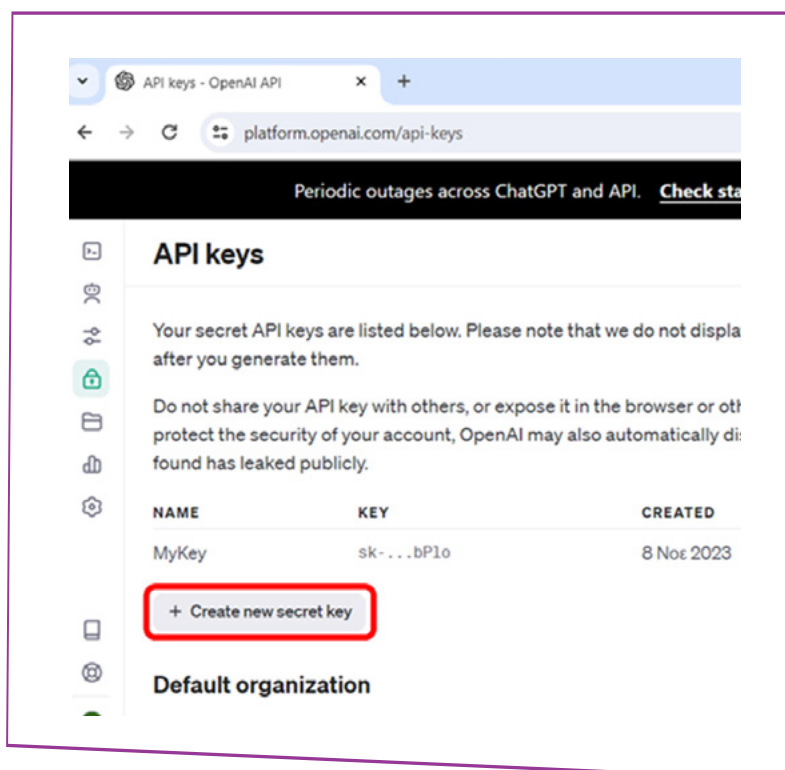
Εκτέλεση δραστηριότητας

3 Στο μενού στην αριστερή πλευρά πατάμε API keys.



Εικόνα 6.

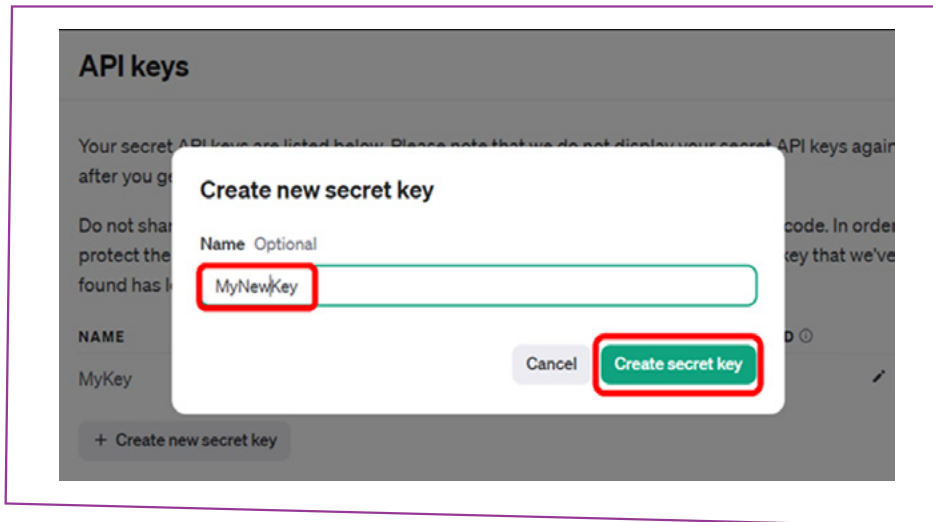
4 Επιλέγουμε Create new secret key.



Εικόνα 7.

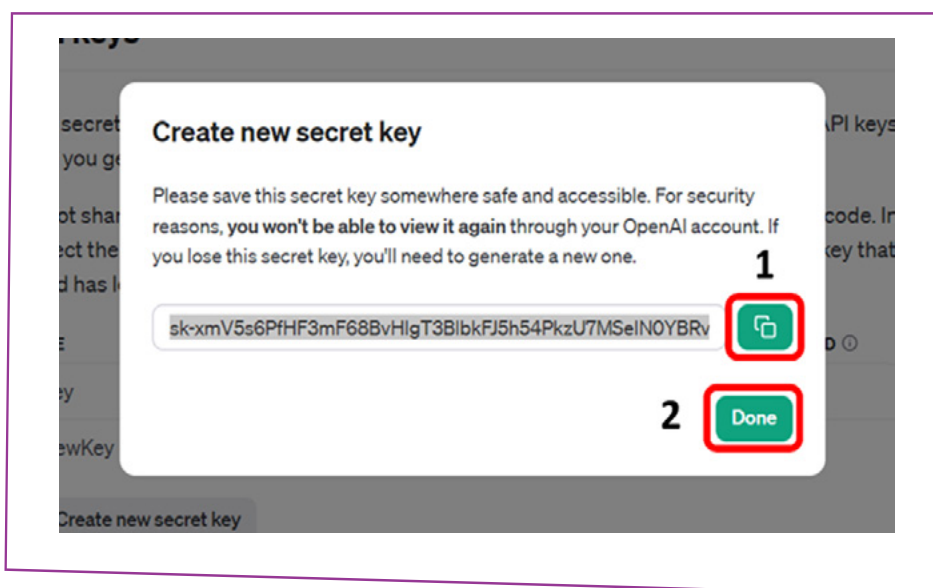
Εκτέλεση δραστηριότητας

- 5 Προαιρετικά ορίζουμε ένα όνομα για το κλειδί και πατάμε το κουμπί Create secret key.



Εικόνα 8.

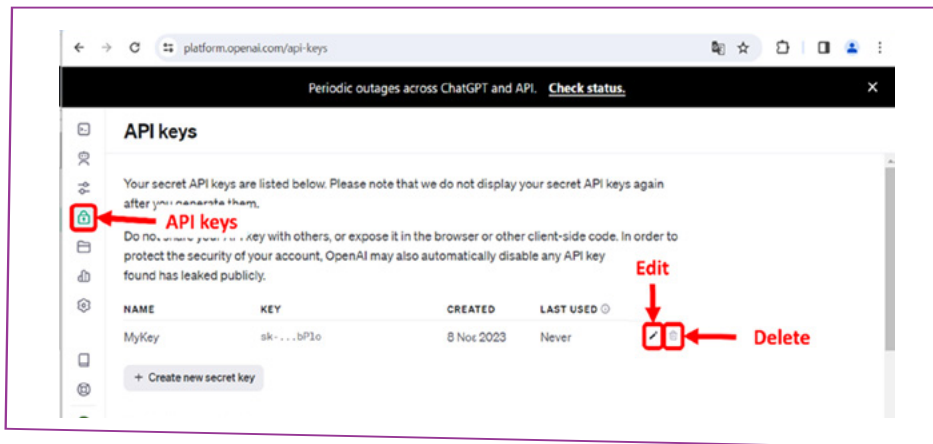
- 6 Αντιγράφουμε το κλειδί πατώντας το σχετικό κουμπί, το κάνουμε επικόλληση σε ένα κενό έγγραφο και το αποθηκεύουμε. Μετά πατάμε Done.



Εικόνα 9.

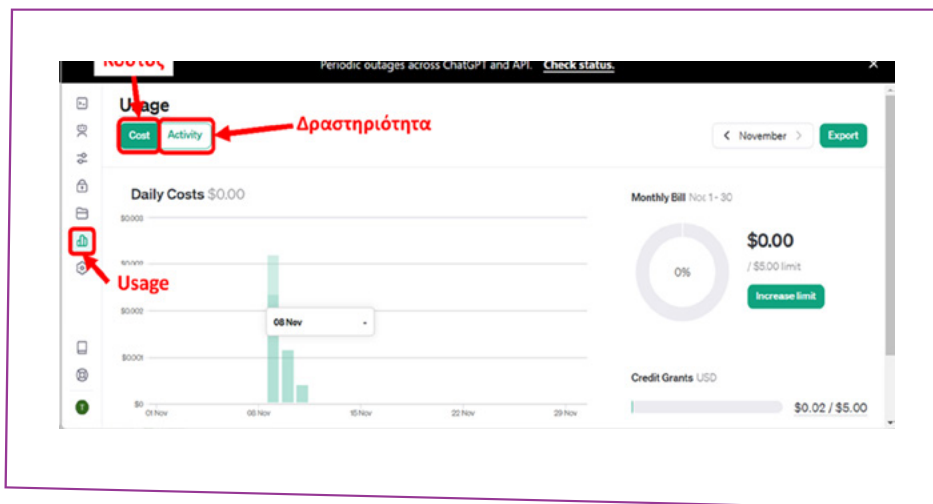
Εκτέλεση δραστηριότητας

- 7 Επιλέγοντας από το αριστερό μενού API keys μπορούμε να διαχειριζόμαστε τα κλειδιά που έχουμε δημιουργήσει.



Εικόνα 10.

- 8 Πατώντας στο Usage βρίσκουμε χρήσιμες πληροφορίες για τη δραστηριότητα και τη χρέωση των APIs.



Εικόνα 11.

Υλικά

Θα χρησιμοποιήσουμε:

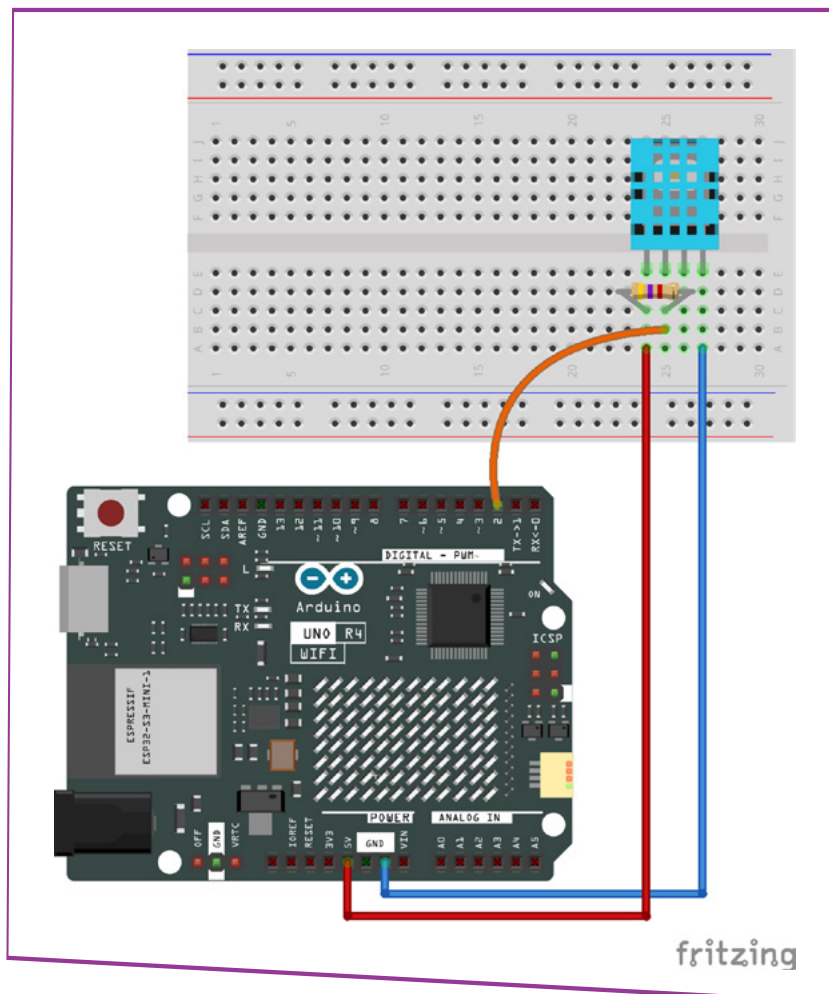
- Μια πλακέτα Arduino Uno R4 WiFi
- Έναν αισθητήρα DHT11
- Ένα Breadboard
- Μια αντίσταση 4.7 kΩ
- Μερικά καλώδια jumper

Εκτέλεση δραστηριότητας

Κύκλωμα

Τοποθετούμε τον DHT11 πάνω στο breadboard και στη συνέχεια, με τα καλώδια και την αντίσταση, πραγματοποιούμε τις συνδέσεις που φαίνονται στον ακόλουθο πίνακα και στο κύκλωμα:

Pin DHT11	Pin Arduino
VCC	5V
DATA	D2
DATA	5V μέσω της αντίστασης 4.7 kΩ
GND	GND



Εικόνα 12.



Εκτέλεση δραστηριότητας

Επισημαίνεται ότι ο ακροδέκτης DATA του αισθητήρα, εκτός από τον ακροδέκτη D2 του Arduino, θα πρέπει να συνδεθεί μέσω της αντίστασης 4.7 kΩ και με το pin 5V της πλακέτας (ή το VCC του DHT11 το οποίο συνδέεται με το 5V).

Ανάπτυξη προγράμματος σε Arduino IDE

Στο πρόγραμμα, η λειτουργία της λήψης μετρήσεων, της επικοινωνίας με το ChatGPT και της εμφάνισης των αποτελεσμάτων, εκτελούνται μέσα από το setup με την κλήση σχετικών συναρτήσεων. Με τον τρόπο αυτό, όταν συνδέετε την πλακέτα στο ρεύμα, η διαδικασία θα εκτελείται μόνο μία φορά και αν θέλετε να την επαναλάβετε θα πρέπει να πατήσετε το κουμπί reset του Arduino.

Για να λειτουργήσει η εφαρμογή, θα πρέπει να εγκατασταθούν στο Arduino IDE οι παρακάτω βιβλιοθήκες:

- DHT sensor library
- ArduinoHttpClient
- Arduino_JSON
- ArduinoGraphics

Η εγκατάσταση των βιβλιοθηκών πραγματοποιείται με τη χρήση του Library Manager, σύμφωνα με τη διαδικασία που παρουσιάστηκε προηγουμένως.

Στον κώδικα της εφαρμογής που παρατίθεται στη συνέχεια, θα πρέπει να γίνουν οι παρακάτω αλλαγές:

- 1 Στην εντολή
`const char* ssid = "Your_WiFi_SSID";`
να αντικαταστήσετε το `Your_WiFi_SSID` με το SSID του ασύρματου δικτύου σας.
- 2 Στην εντολή
`const char* password = "Your_WiFi_Password";`
να αντικαταστήσετε το `Your_WiFi_Password` με το Password του ασύρματου δικτύου.
- 3 Στην εντολή
`String OpenAI_key = "Your_API_key";`
να αντικαταστήσετε το `Your_API_key` με το API key που δημιουργήσατε στο OpenAI.



Εκτέλεση δραστηριότητας

Για τη διευκόλυνσή σας οι σχετικές εντολές εμφανίζονται **τονισμένες**:

```
#include "WiFiS3.h"
#include <ArduinoHttpClient.h>
#include <Arduino_JSON.h>
#include "ArduinoGraphics.h"
#include <Arduino_LED_Matrix.h>
#include <Wire.h>
#include "DHT.h"

#define DHTPIN 2 //Pin σύνδεσης αισθητήρα από κύκλωμα
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

const char* ssid = "Your_WiFi_SSID"; //SSID ασύρματου δικτύου
const char* password = "Your_WiFi_Password"; //Password ασύρματου δικτύου
int status = WL_IDLE_STATUS;
WiFiSSLClient wifi;

HttpClient client = HttpClient(wifi, "api.openai.com", 443); //AI server
String OpenAIModel = "gpt-3.5-turbo-instruct"; //Ορισμός μοντέλου AI
String OpenAI_key = "Your_API_key"; //Το κλειδί API που δημιουργήσατε

ArduinoLEDMatrix matrix;

void setup() {
  Serial.begin(115200);
  matrix.begin(); //Αρχικοποίηση LED Matrix
  dht.begin(); //Αρχικοποίηση αισθητήρα
  connectToWiFi(); //Σύνδεση στο δίκτυο
  delay(1000);

  int humidity = dht.readHumidity(); //Λήψη μετρήσεων
  int temperature = dht.readTemperature();
  String weather = "Temp " + String(temperature) +
    "C Hum " + String(humidity) + "% "; //Μήνυμα με μετρήσεις
  String query = "Given a temperature of " + String(temperature) +
    " degrees Celsius and humidity of " + String(humidity) +
    "%, characterize the weather in a few words:"; //Ερώτημα για ChatGPT
  weather = weather + Characterization(query); //Προσθήκη απόκρισης στο μήνυμα
  displayOnMatrix(weather); //Προβολή μηνύματος
}

void loop() {
}
```



Εκτέλεση δραστηριότητας

```
void connectToWiFi() { //Σύνδεση στο ασύρματο δίκτυο
  while (status != WL_CONNECTED) { //Όσο δεν έχουμε συνδεθεί
    unsigned long start = millis();
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(ssid);
    status = WiFi.begin(ssid, password); //Προσπάθεια σύνδεσης
    while ((millis() - start <= 10000) &&
      status != WL_CONNECTED); //Αναμονή μέχρι την επόμενη προσπάθεια
  }
  Serial.println("Connected");
}

String Characterization(String query) { //Επικοινωνία με το ChatGPT
  String jsonPayload = "{\n\"model\":\n\"\" + OpenAImodel + "\",\n\"prompt\":\n\"\" +
    query + "\",\n\"max_tokens\":10}"; //Προετοιμασία αιτήματος
  client.beginRequest(); //Αποστολή αιτήματος HTTP με Headers
  client.post("/v1/completions");
  client.setHeader("Content-Type", "application/json");
  client.setHeader("Authorization", "Bearer " + OpenAI_key);
  client.setHeader("Content-Length", jsonPayload.length());
  client.beginBody();
  client.print(jsonPayload);
  client.endRequest();
  int httpResponseCode = client.responseStatusCode(); //Λήψη απόκρισης
  String response = client.responseBody();
  if (httpResponseCode == 200) { //Έλεγχος για σφάλμα
    JSONVar jsonData = JSON.parse(response); //Ανάκτηση χαρακτηρισμού
    String message = jsonData["choices"][0]["text"];
    message.trim();
    message = message + " ";
    return message; //Επιστροφή χαρακτηρισμού
  } else {
    return "Failed to get characterization "; //Επιστροφή μηνύματος σφάλματος
  }
}

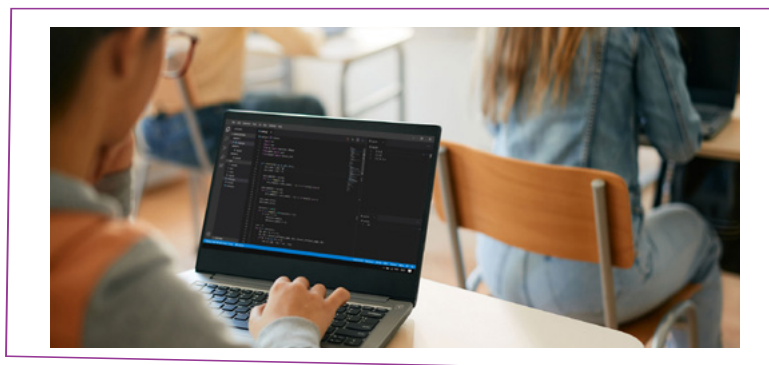
void displayOnMatrix(String characterization) { //Εμφάνιση κειμένου στο LED Matrix
  for (int i = 0; i < 3; i++) { //Το μήνυμα εμφανίζεται 3 φορές
    matrix.beginDraw();
    matrix.stroke(0xFFFFFFFF);
    matrix.textScrollSpeed(70); //Ταχύτητα κύλισης
    matrix.textFont(Font_4x6); //Μέγεθος γραμματοσειράς
    matrix.beginText(10, 1, 0xFFFFFFFF);
    matrix.println(characterization); //Εμφάνιση κειμένου
    matrix.endText(SCROLL_LEFT);
    matrix.endDraw();
  }
}
```

Εκτέλεση δραστηριότητας

Η λειτουργία του κώδικα είναι η εξής:

- Αφού αρχικοποιηθεί ο DHT11 και το ενσωματωμένο LED matrix του Arduino Uno R4, καλείται η συνάρτηση connectToWiFi(), η οποία προσπαθεί κάθε 10 δευτερόλεπτα να κάνει τη σύνδεση με το ασύρματο δίκτυο.
- Μόλις γίνει η σύνδεση, λαμβάνονται οι μετρήσεις της υγρασίας και της θερμοκρασίας και διαμορφώνεται ένα αρχικό μήνυμα εξόδου (Temp xx C Hum yy%).
- Στη συνέχεια διαμορφώνεται το ερώτημα που θα υποβληθεί στο ChatGPT (Given a temperature of xx degrees Celsius and humidity of yy%, characterize the weather in a few words:)
- Ακολούθως καλείται η συνάρτηση Characterization, η οποία υλοποιεί την επικοινωνία με το ChatGPT. Στην Characterization αρχικά διαμορφώνεται το συνολικό αίτημα που θα υποβληθεί στο ChatGPT και περιλαμβάνει πληροφορίες όπως το μοντέλο στο οποίο απευθύνεται το ερώτημα, το ίδιο το ερώτημα και το μέγιστο πλήθος tokens που θα πρέπει να περιλαμβάνει η απόκριση. Η υποβολή του αιτήματος και η λήψη της απόκρισης γίνεται με το πρωτόκολλο HTTP, χρησιμοποιώντας μεθόδους της βιβλιοθήκης ArduinoHttpClient. Αντίστοιχα, η αποκωδικοποίηση της απόκρισης και η ανάκτηση του χαρακτηρισμού των καιρικών συνθηκών γίνεται με τις μεθόδους της βιβλιοθήκης Arduino_JSON.h.
- Τέλος, ο χαρακτηρισμός που έκανε το ChatGPT προστίθεται στο περιεχόμενο του μηνύματος εξόδου, το οποίο στέλνεται για εμφάνιση στο LED Matrix, με την κλήση της συνάρτησης displayOnMatrix(). Η displayOnMatrix χρησιμοποιεί τις μεθόδους της βιβλιοθήκης Arduino_LED_Matrix.

Εφόσον δεν προκύψει κάποιο πρόβλημα με τη λειτουργία του αισθητήρα ή με τη σύνδεση στο διαδίκτυο, η εκτέλεση του κώδικα έχει σαν αποτέλεσμα να εμφανιστεί στο ενσωματωμένο LED matrix ένα κυλιόμενο μήνυμα που περιλαμβάνει τις μετρήσεις και τον χαρακτηρισμό που έκανε το ChatGPT. Το μήνυμα εμφανίζεται 3 φορές.



Εικόνα 13.

Πηγές εικόνων

Εικόνα 1, 3 και 12: Απεικόνιση εξαρτημάτων και κυκλωμάτων που δημιουργήθηκε με το λογισμικό σχεδίασης Fritzing. Η επίσημη ιστοσελίδα του λογισμικού βρίσκεται στη διεύθυνση <https://fritzing.org/>.

Εικόνα 2 και 4: Επεξεργασμένο στιγμιότυπο οθόνης από το ολοκληρωμένο περιβάλλον ανάπτυξης [Arduino IDE](#).

Εικόνα 5-10: Επεξεργασμένο στιγμιότυπο οθόνης από την ιστοσελίδα platform.openai.com/api-keys/.

Εικόνα 11: Επεξεργασμένο στιγμιότυπο οθόνης από την ιστοσελίδα platform.openai.com/usage/.

Εικόνα 13: <https://stock.adobe.com/uk/images/close-up-of-high-school-student-learns-coding-during-computer-class-in-classroom/474764728>

Σημείωση 1

Το υλικό πνευματικής ιδιοκτησίας τρίτων (εικόνες με ή χωρίς προσαρμογή, προσαρμοσμένοι κώδικες, προσαρμοσμένο κείμενο κ.ά.) σημειώνεται ρητά και διανέμεται με την αντίστοιχη άδεια που ορίζεται από τους όρους χρήσης αυτού. Η χρήση στον παρόντα οδηγό γίνεται για δωρεάν εκπαιδευτικούς μη εμπορικούς σκοπούς.

Σημείωση 2

Οι λέξεις Arduino, Arduino UNO R4 WiFi καθώς και λογότυπα ή άλλα διακριτικά γνωρίσματα που αναφέρονται στον παρόντα οδηγό ή απεικονίζονται στις εικόνες που αυτός εμπεριέχει είναι κατοχυρωμένα εμπορικά σήματα και διακριτικά γνωρίσματα που συνιστούν εμπορική ιδιοκτησία της Arduino S.r.l./Arduino AG. Το Ίδρυμα Vodafone, η Vodafone Παναφον Α.Ε.Ε.Τ ή η SciCo δεν σχετίζονται με την εν λόγω εταιρεία.