

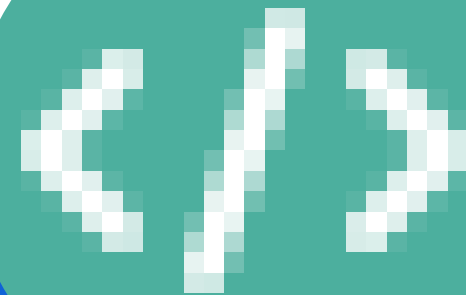
Μάθημα 4 :

Παρακολούθηση Περιβάλλοντος

Block-based Προγραμματισμός

Στόχοι Μαθήματος

Μάθετε τα βασικά του block-based
προγραμματισμού



SCRATCH

Μάθετε πώς να χρησιμοποιείτε το Scratch και το
Mind+ για να προγραμματίσετε μια αποστολή!

Δοκιμάστε τον εαυτό σας προγραμματίζοντας μια
αποστολή με drone!



Εισαγωγή στον Προγραμματισμό που βασίζεται σε μπλοκ

Τι είναι ο προγραμματισμός
που βασίζεται σε μπλοκ;

Ορισμός: Μέθοδος οπτικής κωδικοποίησης για
προγραμματισμό χωρίς σύνταξη, συνδέοντας εντολές με τη
μορφή μπλοκ

Κύριος σκοπός: Να είναι μια φιλική προς αρχάριους
προσέγγιση στον προγραμματισμό, εστιάζοντας στις
βασικές έννοιες της δομής, της λογικής και της επίλυσης
προβλημάτων.



Γιατί να χρησιμοποιήσω προγραμματισμό βασισμένο σε μπλοκ;

1 Απλοποιεί την κωδικοποίηση

- Επιτρέπει στους αρχάριους να γράφουν ολοκληρωμένα προγράμματα χωρίς να ανησυχούν για τις λεπτομέρειες της σύνταξης και άλλες πολυπλοκότητες του προγραμματισμού που βασίζεται σε κείμενο.

2 Οπτική μάθηση

- Επιτρέπει στους προγραμματιστές οποιασδήποτε ηλικίας να κατανοήσουν οπτικά τη δομή και τις συνδέσεις του προγραμματισμού μέσω της αναπαράστασης μπλοκ.

3 Ενθαρρύνει τον πειραματισμό και τη δημιουργικότητα

- Με ένα τόσο εύχρηστο εργαλείο, οι άπειροι προγραμματιστές είναι γρήγορα ελεύθεροι να δοκιμάσουν νέες ιδέες και να δοκιμάσουν τις δεξιότητές τους στην επίλυση προβλημάτων.

Εργαλεία για προγραμματισμό βασισμένο σε μπλοκ

1 Scratch

- Μια δημοφιλής γλώσσα προγραμματισμού βασισμένη σε μπλοκ με το δικό της περιβάλλον ανάπτυξης. Δωρεάν στη χρήση, φιλική προς αρχάριους, με πολλές προκατασκευασμένες εφαρμογές που μπορείτε να βρείτε στην επίσημη ιστοσελίδα!

2 Mind+

- Το Mind+ είναι ένα δωρεάν λογισμικό προγραμματισμού βασισμένο στο Scratch 3.0 που υποστηρίζει διάφορα υλικά ανοιχτού κώδικα όπως Arduino, micro:bit, mPython-esp32 και όλα τα είδη μικροελεγκτών.
- Το Mind + υποστηρίζει δεκάδες βιβλιοθήκες μονάδων επέκτασης που χρησιμοποιούνται συνήθως, γεγονός που το καθιστά κατάλληλο για προγραμματισμό με drone.

Τύποι μπλοκ

1

MOTION

It allows you to position and move the sprite within the work area.

2

LOOKS

It allows you to change the appearance and let the sprite speak.

3

SOUND

It allows you to introduce sound effects.

4

EVENTS

It allows to start the program according to the different events and manages the messages.

5

CONTROL

It allows you to make loops and insert conditional blocks.

6

SENSING

It allows you to manage interactions.

7

OPERATORS

It allows you to perform mathematical and comparison operations.

8

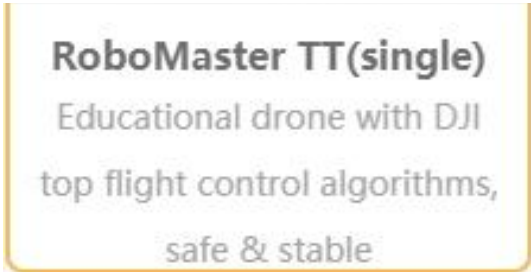
VARIABLES

It allows you to create and manage variables.

Προγραμματισμός βασισμένος σε μπλοκ και drones

Ο προγραμματισμός που βασίζεται σε μπλοκ μπορεί να χρησιμοποιηθεί για τον προγραμματισμό συσκευών drone, κατευθύνοντάς τις να ακολουθούν διαφορετικές διαδρομές πτήσης, να εκτελούν απογειώσεις και προσγειώσεις, να χρησιμοποιούν αισθητήρες, να τραβούν φωτογραφίες και πολλά άλλα.

Για να προγραμματίσετε το drone Robomaster TT στο Mind+, πρέπει πρώτα να εγκαταστήσετε την σωστή

Επιλέξτε  έπειτα  και μετά 

Στο Mind+, τα μπλοκ που χρησιμοποιούνται για τον προγραμματισμό των drones είναι

Fuchsia

και μπορεί να βρεθεί στην ενότητα  μετά την εγκατάσταση της επέκτασης.

Απλό παράδειγμα προγραμματισμού drone με βάση μπλοκ

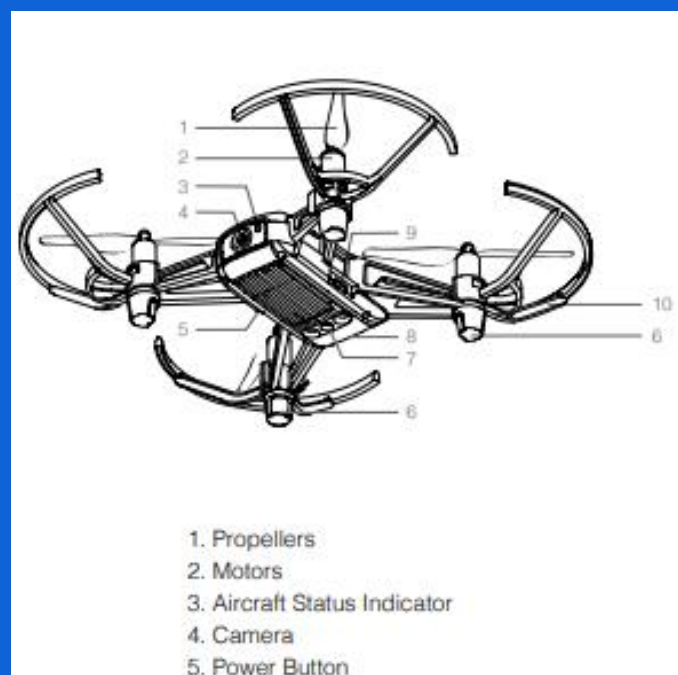
Τα μπλοκ συνδέονται μεταξύ τους σαν ένα παζλ.

Ο προγραμματιστής μπορεί να σύρει και να αποθέσει μπλοκ για να δημιουργήσει το πρόγραμμά του.

Αυτό το απλό παράδειγμα δείχνει πόσο εύκολα μπορούν να χρησιμοποιηθούν μπλοκ για να προγραμματιστεί ένα drone ώστε να: απογειωθεί, να πετάξει αριστερά για 50 εκατοστά, να πετάξει δεξιά για 50 εκατοστά, να το επαναλάβει 10 φορές και μετά να προσγειωθεί.



Αποστολή περίπτωσης χρήσης 2: Σύνδεση του Drone



Βήμα 1

Πατήστε το κουμπί
λειτουργίας του
drone.

Συνδεθείτε με το
drone μέσω Wifi,
στο δίκτυο TELLO
ή RMTT



Βήμα 2

Βήμα 3

Μετάβαση στο
μενού



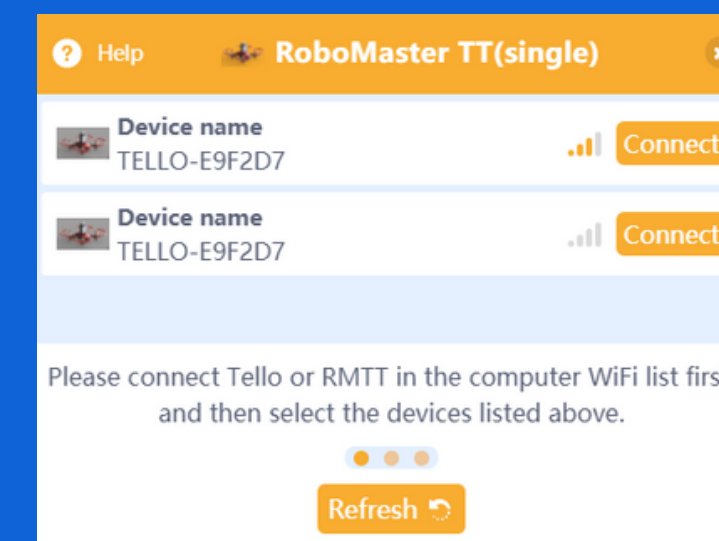
στο Mind+ και
πατήστε το



▶ RoboMaster TT(single)



Συνδεθείτε στη
συσκευή μέσω του
Mind+



Βήμα 4

Αποστολή περίπτωσης χρήσης 2: Προγραμματισμός Drone και αισθητήρες

Σκοπός:

Στόχος της αποστολής είναι να:

- Διδάξτε προγραμματισμό drone με block – based γλώσσα προγραμματισμού
- Διδάξτε τη χρήση αισθητήρων με drones
- Διδάξτε τη συλλογή και ανάλυση δεδομένων αισθητήρων

Σε αυτό το μάθημα προγραμματισμού

θα κάνετε τα εξής:

- Προγραμματίστε ένα drone Robomaster TT χρησιμοποιώντας το Mind+
- Συλλέξτε δεδομένα από τους εσωτερικούς αισθητήρες του Drone
- Οργανώστε τα δεδομένα σε λίστες για να βοηθήσετε στην ανάλυση δεδομένων

Αποστολή περίπτωσης χρήσης 2: Βασικά στοιχεία πτήσης με drone

Ξεκινώντας

Χρησιμοποιώντας απλές εντολές που βρίσκονται στα



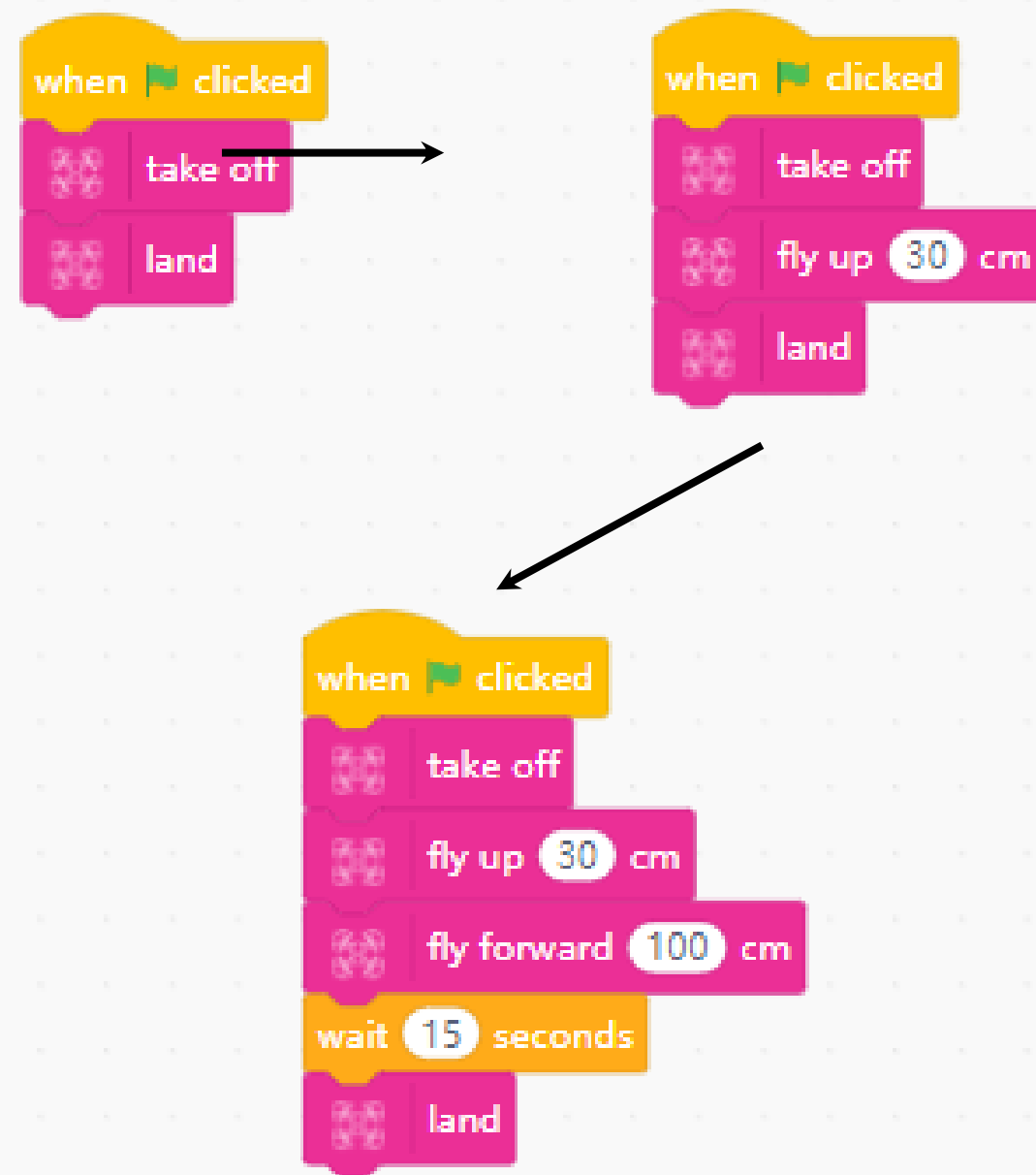
και



Σταδιακά φτιάχνετε ένα απλό πρόγραμμα για το drone για να:

- Απογειώνεται και μετά να πετάει 30 εκατοστά ψηλά
- Πετάει προς τα εμπρός 100 εκατοστά και να περιμένει 15 δευτερόλεπτα
- Προσγειώνεται

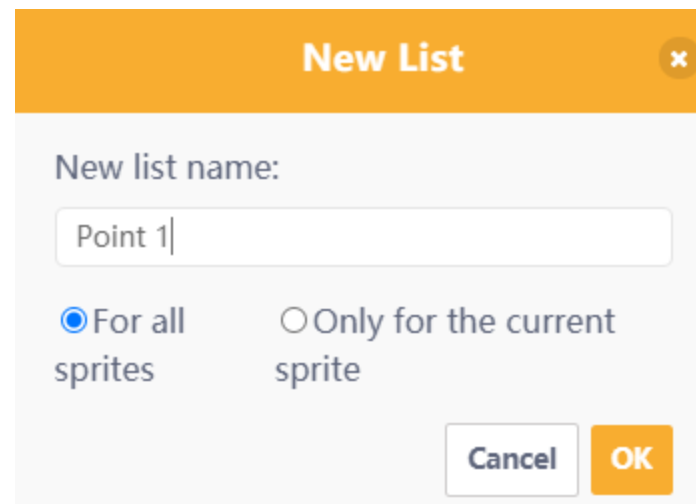
(1)



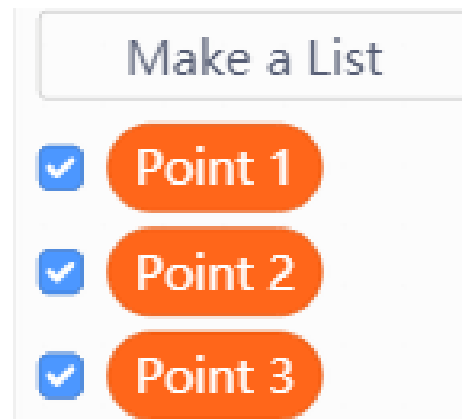
Αποστολή περίπτωσης χρήσης 2: Δημιουργία Λιστών

Δημιουργία λιστών

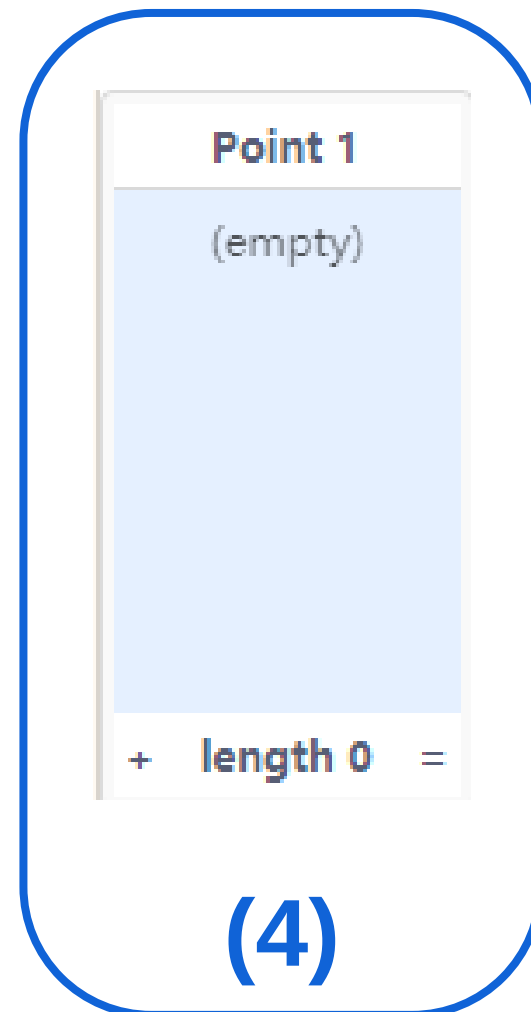
(2)



(3)



(4)



Πριν χρησιμοποιήσουμε λίστες για την αποθήκευση δεδομένων, πρέπει να τις δημιουργήσουμε. Στην ενότητα



βρείτε την εντολή

Make a List

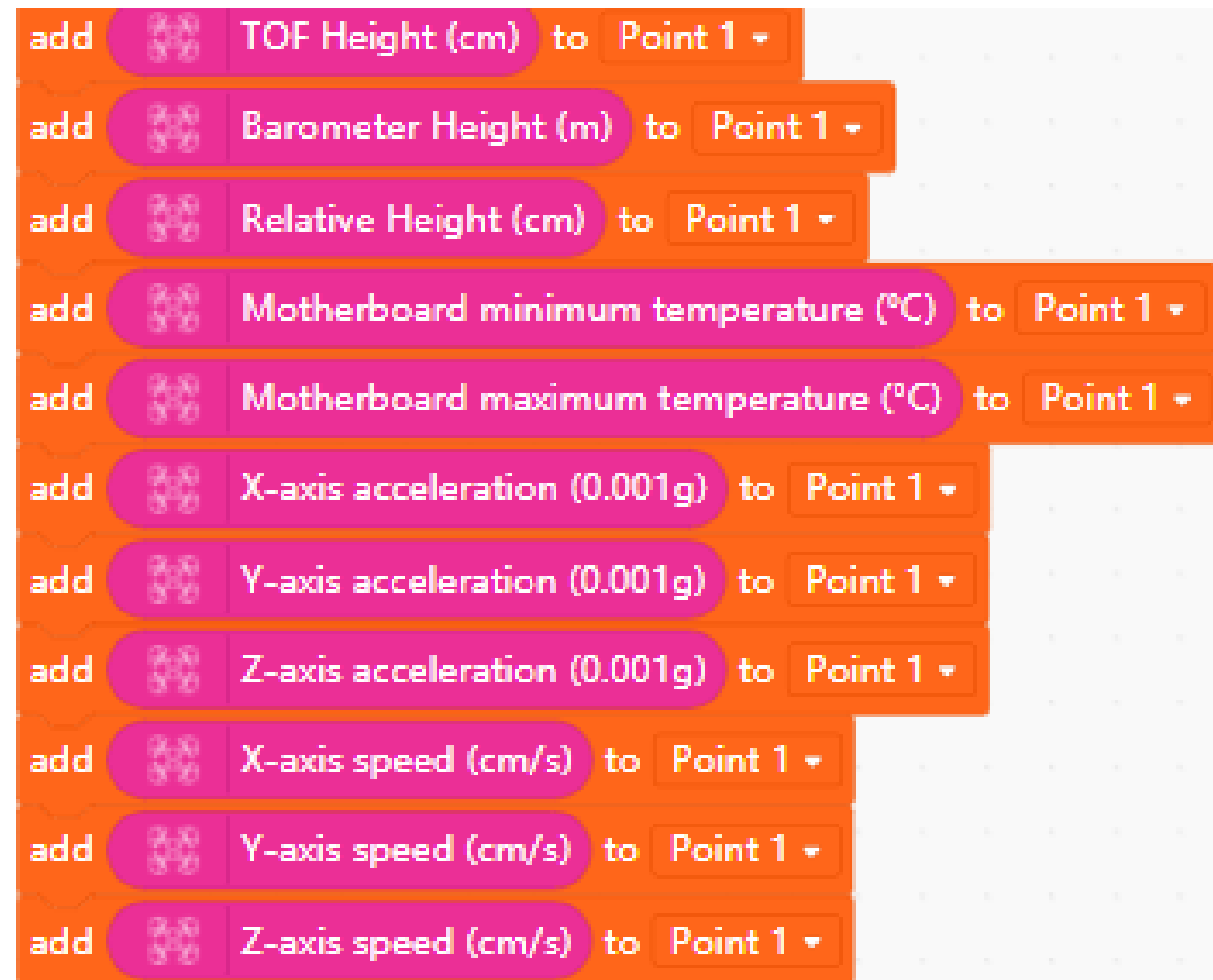
που προκαλεί το αναδυόμενο παράθυρο (2).

Χρησιμοποιούμε αυτήν την εντολή για να δημιουργήσουμε τρεις λίστες: Point 1, Point 2, Point 3, που φαίνονται στο (3).

Αυτές οι λίστες εμφανίζονται κενές στην ενότητα sprite, όπως φαίνεται στο (4).

Αποστολή περίπτωσης χρήσης 2: Συμπλήρωση λιστών με δεδομένα

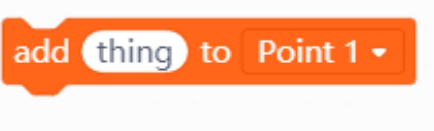
(5)



The screenshot shows a vertical stack of ten 'add' blocks, each with a sensor icon and a dropdown menu set to 'Point 1'. The blocks are as follows:

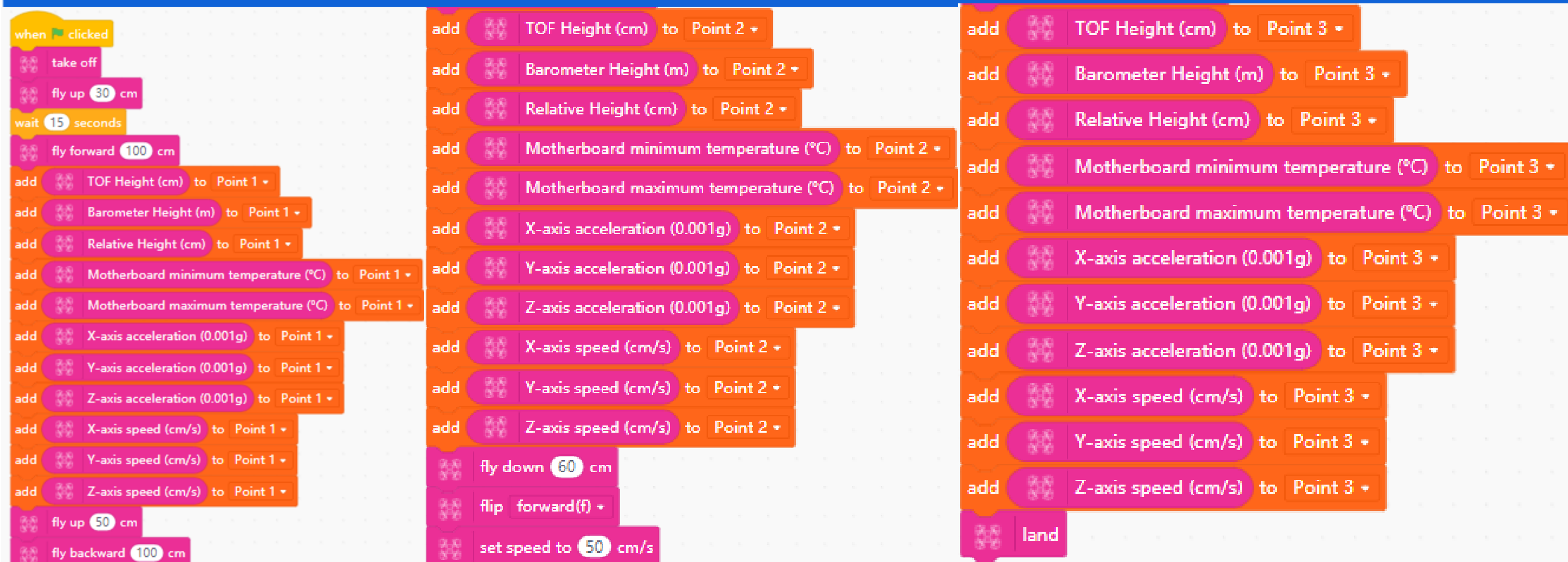
- add TOF Height (cm) to Point 1
- add Barometer Height (m) to Point 1
- add Relative Height (cm) to Point 1
- add Motherboard minimum temperature (°C) to Point 1
- add Motherboard maximum temperature (°C) to Point 1
- add X-axis acceleration (0.001g) to Point 1
- add Y-axis acceleration (0.001g) to Point 1
- add Z-axis acceleration (0.001g) to Point 1
- add X-axis speed (cm/s) to Point 1
- add Y-axis speed (cm/s) to Point 1
- add Z-axis speed (cm/s) to Point 1

Δεδομένα σε λίστες

Αφού δημιουργήσετε τις λίστες, στην ενότητα  Variables βρίσκετε την εντολή  την οποία μπορείτε να χρησιμοποιήσετε για να βάλετε διαφορετικές μεταβλητές στις λίστες.

Στην ενότητα  Function βρίσκετε μεταβλητές όπως  την οποία χρησιμοποιείτε στο (5) για να βάλετε τα δεδομένα που συλλέγονται από τους εσωτερικούς αισθητήρες του drone στη λίστα Point 1.

Αποστολή περίπτωσης χρήσης 2: Τελικό πρόγραμμα



Συνδυασμός

Επαναλαμβάνετε τη διαδικασία στο (5) για τις λίστες Point 2 και Point 3. Στη συνέχεια, συνδυάζετε αυτές τις εντολές με το απλό μας πρόγραμμα στο (1).

Ανάμεσα στις μετρήσεις, προσθέτετε διαφορετικές ενέργειες drone για να διασφαλίσετε ότι υπάρχει ποικιλία

Αποστολή περίπτωσης χρήσης 2: Τελικό Αποτέλεσμα

Τέλος του Προγράμματος

Αφού εκτελέσετε το πρόγραμμα, έχετε 3 διαφορετικές λίστες με μετρήσεις που προέρχονται από τους αισθητήρες του drone.

Θα χρησιμοποιήσετε αυτά τα δεδομένα στα επόμενα στάδια της αποστολής!

Η διαδικασία μπορεί να αναπαραχθεί χρησιμοποιώντας εξωτερικούς αισθητήρες αντί για εσωτερικούς, για τη συλλογή δεδομένων που σχετίζονται με το περιβάλλον!

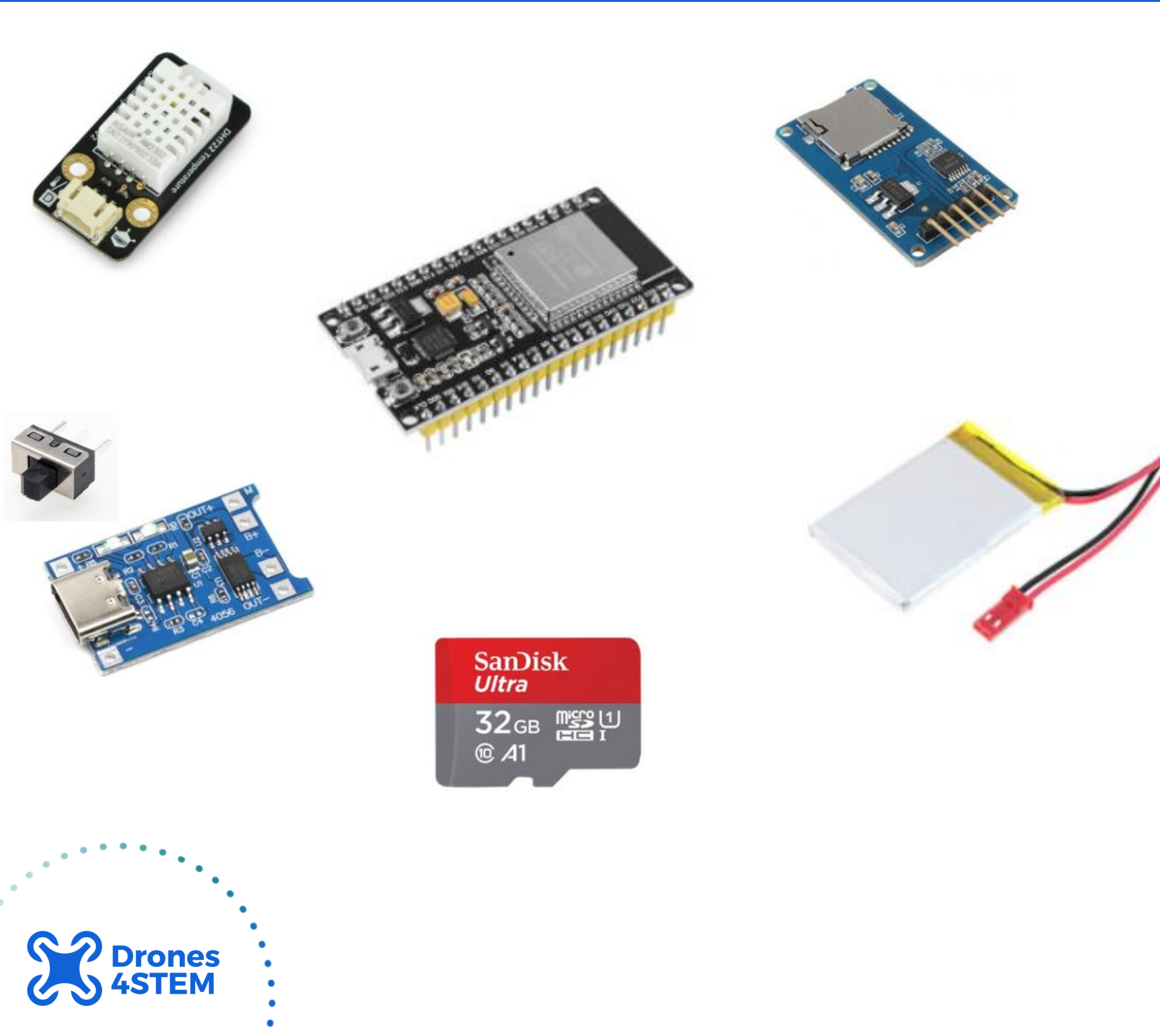
Point 1		Point 2		Point 3	
1	107	1	220	1	154
2	37.84	2	38.22	2	38.02
3	80	3	130	3	120
4	71	4	71	4	69
5	74	5	73	5	72
6	-32.00	6	-32.00	6	-34.00
+ length 11 =		+ length 11 =		+ length 11 =	

Αποστολή περίπτωσης χρήσης 2: Bonus

Χρήση εξωτερικών αισθητήρων

Για να χρησιμοποιήσετε εξωτερικούς αισθητήρες, θα προσθέσετε μια επιπλέον ενότητα στην Περίπτωση Χρήσης. Αυτό απαιτεί:

- Ένας αισθητήρας θερμοκρασίας και υγρασίας DHT11
- Μια πλακέτα ανάπτυξης ESP32
- Μια πλακέτα Breakout για κάρτα Micro SD Mini
- Μπαταρία (Li-Po - 3.7v 800mAh)
- Ένας φορτιστής μπαταρίας (1A USB-C - TP4056)
- Ένας διακόπτης (Mini SPDT - 1A/250VAC)
- Μια κάρτα microSD (SDSQUA4-032G-GN6MA)



Αποστολή περίπτωσης χρήσης 2: Bonus



Προγραμματισμός

Για τη χρήση εξωτερικού αισθητήρα, θα μεταβούμε στην ενότητα Κώδικας του Mind+ για να γράψουμε κώδικα Python.

Το αρχείο main.py της περίπτωσης χρήσης είναι αυτό που θα χρησιμοποιήσουμε.

Για να δημιουργήσετε ένα νέο αρχείο Python, κάντε κλικ στο εικονίδιο  δίπλα στο "Αρχείο στο έργο" και, στη συνέχεια, γράψτε ή επικολλήστε τον κώδικά σας.

Το αρχείο main.py θα προγραμματίσει τη συλλογή δεδομένων από τον αισθητήρα.

Αποστολή περίπτωσης χρήσης 2: Bonus

```
import machine
import dht
import time
import os
from machine import SPI, Pin
import sdcard

# DHT22 Sensor Setup
DHT_PIN = 4
dht_sensor = dht.DHT22(machine.Pin(DHT_PIN))

# SD Card Configuration
CS_PIN = 5 # Confirm this matches your wiring
spi = SPI(1,
          baudrate=400000, # Reduced baudrate for stability
          sck=Pin(13),
          mosi=Pin(15),
          miso=Pin(14))

try:
    sd = sdcard.SDCard(spi, Pin(CS_PIN))
    os.mount(sd, '/sd')
    print("SD card mounted successfully")
    print("Root directory:", os.listdir('/'))
    print("SD card contents:", os.listdir('/sd'))
except Exception as e:
    print("SD card initialization failed:", e)
    machine.reset()

# File Configuration
FILE_PATH = '/sd/data.csv'

# Test file operations
try:
    with open(FILE_PATH, 'a') as f:
        f.write('Test entry\n')
        print("File write test successful")
except Exception as e:
    print("File operation failed:", e)

def log_data(temp, hum):
    try:
        timestamp = "{:02d}::{:02d}::{:02d}".format(*time.localtime())
        data = f"Temp {temp:.1f},Hum {hum:.1f}\n"
        with open(FILE_PATH, 'a') as f:
            f.write(data)
        print(f"Data logged: {data.strip()}")
    except Exception as e:
        print("Logging failed:", e)

while True:
    try:
        dht_sensor.measure()
        temp = dht_sensor.temperature()
        hum = dht_sensor.humidity()
        log_data(temp, hum)
    except OSError as e:
        print("Sensor read error:", e)

    time.sleep(10)
```

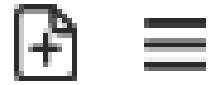
Ο κώδικας

Ο κώδικας pythοn στο main.py φροντίζει για τα ακόλουθα πράγματα:

- Ρύθμιση του αισθητήρα DHT11
- Ρύθμιση παραμέτρων της κάρτας SD
- Δημιουργία αρχείου .csv για τα δεδομένα
- Καταγραφή των μετρήσεων του αισθητήρα στο αρχείο .csv

Αποστολή περίπτωσης χρήσης 2: Bonus

File in the project



 main.py

 sdcard.py

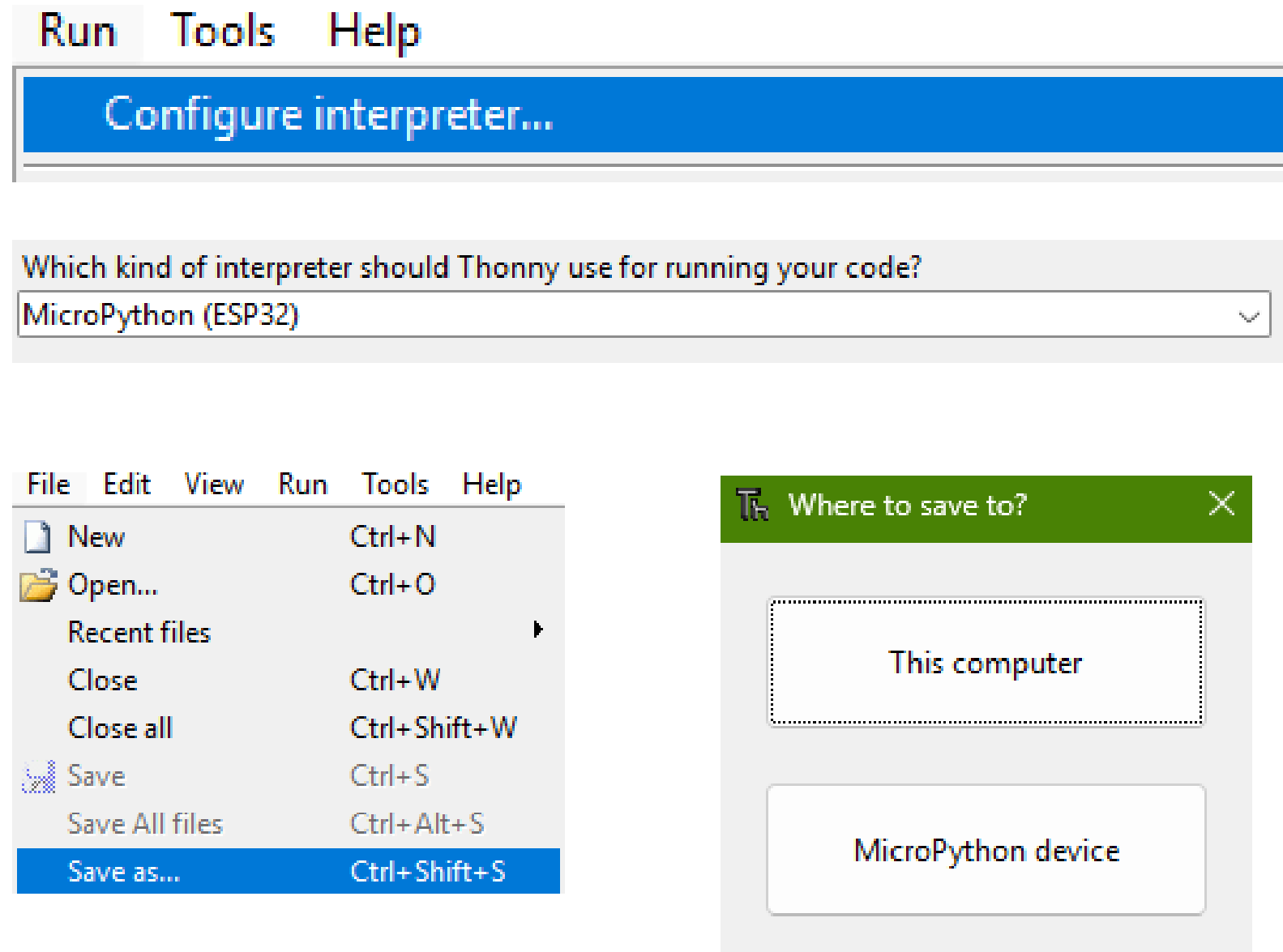
Βιβλιοθήκη SD

Για να λειτουργήσει ο κώδικας στο main.py, απαιτείται μια πρόσθετη βιβλιοθήκη.

Η βιβλιοθήκη sdcard.py μπορεί να βρεθεί σε αυτόν τον σύνδεσμο: <https://github.com/micropython/micropython-lib/blob/master/micropython/drivers/storage/sdcard/sdcard.py>

Αντιγράψτε τον κώδικα και επικολλήστε τον σε ένα νέο αρχείο python με το όνομα sdcard.py

Αποστολή περίπτωσης χρήσης 2: Bonus



Φόρτωση κώδικα στον πίνακα ανάπτυξης

Ένας τρόπος για να αποθηκεύσουμε τα αρχεία `main.py` και `sdcard.py` στην πλακέτα ανάπτυξης είναι μέσω του IDE Thonny, ακολουθώντας τα ακόλουθα βήματα:

- Βήμα 1: Ανοίξτε το αρχείο `main.py` με το Thonny.
- Στο Thonny, κάντε κλικ στην επιλογή **Εκτέλεση**, έπειτα στην επιλογή **Ρύθμιση παραμέτρων διερμηνέα** και επιλέξτε `MicroPython ESP32`. Πατήστε **OK**.
- Στη συνέχεια, κάντε κλικ στο **Αρχείο**, «**Αποθήκευση ως**» και επιλέξτε την επιλογή `MicroPython`.
- Επαναλάβετε για το αρχείο `sdcard.py`.

Τώρα, ο κώδικας βρίσκεται στην πλακέτα ανάπτυξης και θα εκτελείται αυτόματα κάθε φορά που ενεργοποιείτε τον διακόπτη!

Ερωτήσεις

- Πώς ο προγραμματισμός που βασίζεται σε μπλοκ καθιστά τον προγραμματισμό πιο προσβάσιμο;
- Γιατί το drone έχει εσωτερικούς αισθητήρες;
- Πώς μπορεί να χρησιμοποιηθεί ο προγραμματισμός drone βάσει μπλοκ στην παρακολούθηση του περιβάλλοντος;

Ποιες είναι οι ερωτήσεις σας;



Πρόσθετοι Πόροι

Περαιτέρω ανάγνωση

<https://mindplus.dfrobot.com/RMTT>

Βίντεο

Παρουσιάζουμε το Mind+: Ένα λογισμικό προγραμματισμού Sctrach 3.0



Σας ευχαριστώ για την
προσοχή σας!



Co-funded by
the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

