

Φύλλο Εργασίας 9

Αισθητήρες και μετρήσεις...



Τι θα μάθουμε;

Θα δούμε πώς μπορούμε να φτιάξουμε ένα απλό πρόγραμμα με την Python στο AstroPi. Θα μάθουμε πώς να χρησιμοποιούμε το AstroPi για να κάνουμε μετρήσεις και να επεξεργαζόμαστε τις τιμές των μετρήσεων αυτών με την Python.

Για τις δοκιμές μας θα χρησιμοποιήσουμε την ιστοσελίδα <https://trinket.io/sense-hat>



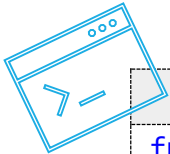
Τι πρέπει να θυμόμαστε;

Όλα όσα έχουμε μάθει ως τώρα για τον προγραμματισμό με την Python, μπορούν να χρησιμοποιηθούν και στο AstroPi. Για τις παρακάτω δραστηριότητες πρέπει να θυμόμαστε πολύ καλά όλα όσα έχουμε μάθει για τον τρόπο που εργαζόμαστε με τις μεταβλητές.



Τι θα μπορούμε να κάνουμε μετά;

Θα μπορούμε να φτιάχνουμε προγράμματα που θα χρησιμοποιούν τους αισθητήρες του AstroPi για να κάνουν μετρήσεις, θα μπορούμε να κάνουμε υπολογισμούς με τις μετρήσεις αυτές και να δείχνουμε τα αποτελέσματα στην οθόνη ή/και στην οθόνη του AstroPi.



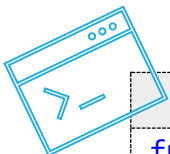
Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat() sense.show_message("Hello!")</pre>	

1

1. Τρέξτε τον παραπάνω κώδικα και παρατηρήστε την οθόνη του AstroPi. Ποια εντολή νομίζετε ότι «εμφανίζει το μήνυμα»; Ποια εντολή της Python κάνει κάτι παρόμοιο;



*Tip: Οι δύο πρώτες γραμμές του κώδικα χρησιμοποιούνται για να «συνδέσουν» το πρόγραμμά μας με τους αισθητήρες του AstroPi. Θα δούμε περισσότερα για το **from** και το **import** αργότερα.*



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat() sense.show_message("Hello!") print("Hello Python!") sense.show_message("Goodbye!")</pre>	Hello Python!

2

2. Τρέξτε τον παραπάνω κώδικα! Πού εμφανίζεται το κάθε μήνυμα; Σε αυτό το πρόγραμμα, πώς νομίζετε ότι πρέπει να ξεκινάνε οι εντολές που έχουν σχέση με το AstroPi;

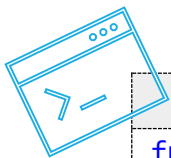


Οι παρακάτω εντολές δείχνουν πώς μπορούμε να πάρουμε μετρήσεις από τους αισθητήρες του AstroPi, ώστε να τις χρησιμοποιήσουμε στο πρόγραμμά μας στην Python:

<code>.get_humidity()</code>	Επιστρέφει την τιμή της «υγρασίας»
<code>.get_temperature()</code>	Επιστρέφει την τιμή της «θερμοκρασίας»
<code>.get_pressure()</code>	Επιστρέφει την τιμή της «ατμοσφαιρικής πίεσης»
<code>.get_compass()</code>	Επιστρέφει τη διεύθυνση του Βορρά (σε μοίρες)
<code>.get_compass_raw()</code>	Επιστρέφει την ένταση του «μαγνητικού πεδίου» (σε τρεις τιμές)



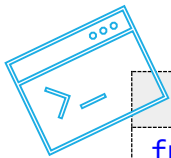
Tip: Όπως κάνουμε και με το **input()**, πρέπει να αποθηκεύουμε την τιμή της μέτρησης που παίρνουμε σε μια μεταβλητή. Στεφτείτε τις παραπάνω εντολές ως ένα «ιδιαίτερο» **input()**.



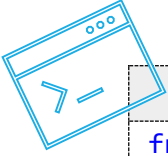
Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat() temp = sense.get_temperature() print("Θερμοκρασία:" , temp , "βαθμοί")</pre>	Θερμοκρασία: 19.833333333333333 βαθμοί

3

3. Να αλλάξετε **μόνοι σας** το παραπάνω πρόγραμμα ώστε να μετράει το επίπεδο της υγρασίας, τη θερμοκρασία και την ατμοσφαιρική πίεση και να τα δείχνει σε ξεχωριστά μηνύματα στην οθόνη. Αφού τελειώσουν όλες οι μετρήσεις και τα μηνύματα στην οθόνη, να εμφανιστεί το μήνυμα «End» στην οθόνη του AstroPi.



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat()</pre>	



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat() import time temp1 = sense.get_temperature() time.sleep(5) temp2 = sense.get_temperature() print("Θερμοκρασία:" , temp1 , "βαθμοί") print("Θερμοκρασία:" , temp2 , "βαθμοί")</pre>	Θερμοκρασία: 19.823333333333333 βαθμοί Θερμοκρασία: 19.633386833333333 βαθμοί

4

4. **Τρέξτε μόνοι σας τον παραπάνω κώδικα!** Τι νομίζετε ότι κάνει η εντολή **time.sleep(5)**; Ποια άλλη γραμμή κώδικα νομίζετε ότι είναι απαραίτητη για την **time.sleep(5)**;

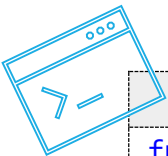


5

5. **Θα φτιάξουμε μαζί** ένα πρόγραμμα που θα μετράει κάθε 5 δευτερόλεπτα τη θερμοκρασία και θα τη δείχνει στην οθόνη. Το πρόγραμμά μας θα «τρέχει» συνεχώς. Επίσης, πριν την κάθε μέτρηση θα εμφανίζεται στην οθόνη του AstroPi το μήνυμα «...».



Tip: Όταν θέλουμε να κάνουμε ένα κομμάτι κώδικα να «τρέχει για πάντα» μπορούμε να χρησιμοποιήσουμε στο **while** μια συνθήκη που είναι πάντα αληθής, ώστε να μη χρειάζεται να την αλλάζουμε με μεταβλητές, π.χ. **while 1 == 1** : ή **while 1 < 2**: ή ακόμη και **while True**:



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat()</pre>	

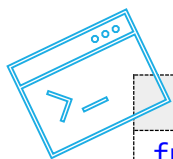


6. **Φτιάξτε με την ομάδα σας** ένα πρόγραμμα που θα μετράει τη θερμοκρασία ή την υγρασία του χώρου και θα κάνει όλα τα παρακάτω:

- Το πρόγραμμά σας θα κάνει μόνο 10 μετρήσεις.
- Πριν ξεκινήσει τις μετρήσεις, θα ρωτήσει τον χρήστη κάθε πόσα δευτερόλεπτα θα πρέπει να κάνει μέτρηση.
- Πριν ξεκινήσει τις μετρήσεις, θα ρωτήσει τον χρήστη τι θέλει να μετρήσει (τη θερμοκρασία ή την υγρασία) και θα μετράει μόνο αυτή την τιμή.
- Για κάθε μέτρηση που θα κάνει, θα δείχνει την τιμή της μέτρησης στην οθόνη και θα την αποθηκεύει σε μια λίστα.
- Πριν ξεκινήσουν οι μετρήσεις, θα εμφανιστεί στην οθόνη του AstroPi το μήνυμα «Start» και μόλις τελειώσουν οι μετρήσεις το μήνυμα «End».
- Μετά τις μετρήσεις, το πρόγραμμα θα ρωτάει αν θέλουμε να μας δείξει όλα τα αποτελέσματα των μετρήσεων στην οθόνη ή απλά να κλείσει χωρίς να τα δείξει.



Tip: Θα χρειαστεί να φτιάξετε αρκετές **μεταβλητές** για να αποθηκεύετε τις απαντήσεις του χρήστη και τις μετρήσεις. Θα χρειαστεί να χρησιμοποιήσετε το **while**, το **if** και ίσως το **elif** ή το **else**. Θα χρειαστεί να φτιάξετε μια **λίστα** και να προσθέσετε τις τιμές των μετρήσεων.



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat()</pre>	

7

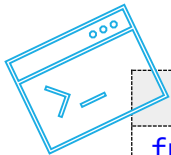
7. **Γράψτε μόνοι σας** ένα πρόγραμμα που θα ρωτάει πόσες μετρήσεις της θερμοκρασίας θέλουμε να κάνουμε (π.χ 5, 10, 20, 100...) και κάθε πόσα δευτερόλεπτα (π.χ. 1, 3, 5...) να τις κάνουμε. Το πρόγραμμα θα δείχνει στην οθόνη τον μέσο όρο όλων των μετρήσεων της θερμοκρασίας και στη συνέχεια θα σταματάει.



Tip: Το πρόγραμμα φαίνεται απλό, αλλά δεν είναι! Σκεφτείτε ότι κάποιος μπορεί να ζητήσει 1000 μετρήσεις! Δεν είναι εύκολο να φτιάξετε 1000 μεταβλητές (π.χ. x1, x2, x3... x1000) και μετά να τις προσθέσετε όλες μαζί και να τις διαιρέσετε με το 1000. Θα ήταν ένα τεράστιο πρόγραμμα!

*Μπορείτε να φτιάξετε το πρόγραμμα αυτό σε λιγότερες από 20 γραμμές κώδικα. Πρέπει να εφαρμόσετε όλα όσα έχετε μάθει για το **while**!*

Θα χρειαστείτε μια μεταβλητή που θα αυξάνεται κάθε φορά που κάνετε μια μέτρηση. Θα χρειαστείτε ακόμη μια μεταβλητή που θα ξεκινάει από το 0 και θα αυξάνεται προσθέτοντας την τιμή της κάθε μέτρησης που κάνετε, όπως στο πρόγραμμα που κάναμε με την ταμειακή μηχανή για να υπολογίσουμε το σύνολο της τιμής όλων των προϊόντων!



Κώδικας	Αποτέλεσμα
<pre>from sense_hat import SenseHat sense = SenseHat()</pre>	