

## Αναπαράσταση έγχρωμης εικόνας

Για να αναπαρασταθούν έγχρωμες εικόνες, χρειάζεται για κάθε pixel να κωδικοποιηθεί με μαθηματικό τρόπο η πληροφορία που αποτυπώνει την τιμή της απόχρωσής του. Αυτό διευθετήθηκε με την υιοθέτηση συγκεκριμένων χρωματικών μοντέλων.

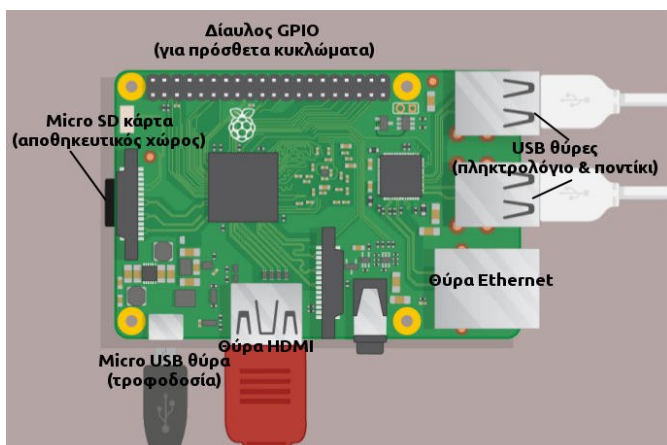
Κάθε pixel έχει ένα μοναδικό χρώμα που παράγεται ως συνδυασμός βασικών χρωμάτων. Θα αναφερθούμε στα χρωματικά πρότυπα RGB και CMYK. Τα ακρωνύμια αφορούν στα βασικά χρώματα που χρησιμοποιούνται για να δημιουργήσουν αποχρώσεις.

**RGB (Red Green Blue):** Τα τρία βασικά χρώματα αυτού του μοντέλου είναι το Κόκκινο, το Πράσινο και το Μπλε. Χρησιμοποιείται για απεικόνιση σε ηλεκτρονικά μέσα, όπως οθόνες υπολογιστών, κινητών κλπ. Ονομάζεται και προσθετικό μοντέλο, με το Λευκό να είναι το πρόσθετο χρώμα που προκύπτει από το συνδυασμό και των τριών βασικών χρωμάτων. Αντίστροφα, το Μαύρο προκύπτει από την απουσία και των τριών χρωμάτων και αντιστοιχεί στην ουδέτερη κατάσταση του pixel μιας σβηστής οθόνης (μαύρο).

**CMYK (Cyan Magenta Yellow black):** Το CMYK χρησιμοποιείται στην έντυπη τυπογραφία. Απαρτίζεται από 4 βασικά χρώματα, το Κυανό, το Ματζέντα (φούξια), το Κίτρινο και το Μαύρο. Πρόκειται για αφαιρετικό μοντέλο. Απουσία και των τεσσάρων βασικών χρωμάτων αποδίδει Λευκό, που είναι και το φυσικό χρώμα του χαρτιού. Ενώ η προσθήκη μελανιών, αποδίδει το αντίστοιχο χρώμα.

Θα πειραματιστούμε με το χρωματικό μοντέλο RGB. Θα δούμε πως χρησιμοποιείται σε απλά γραφικά ακόμη και σήμερα σε κονσόλες που υποστηρίζουν οθόνες από LED (Light Emmiting Diode) φωτάκια. Θα γνωρίσουμε τον μικροϋπολογιστή RaspberryPi και το πρόσθετό του SenseHAT, που διαθέτει μια οθόνη από LED. Στο τέλος θα γράψουμε ένα μικρό πρόγραμμα σε γλώσσα προγραμματισμού Python, για να εμφανίσουμε ένα μήνυμα, να φωτίσουμε μερικά pixels και να σχηματίσουμε μια εικόνα.

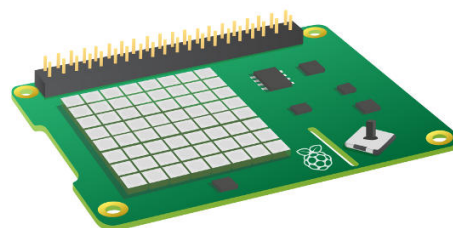
### Raspberry Pi



Το Raspberry Pi είναι ένας μικροϋπολογιστής σε μέγεθος πιστωτικής κάρτας. Διαθέτει 4 θύρες USB για τη σύνδεση πληκτρολογίου, ποντικιού κτλ., θύρα Ethernet για σύνδεση σε τοπικό δίκτυο, ενώ υποστηρίζει και ασύρματη πρόσβαση σε δίκτυο, μία θύρα τροφοδοσίας Micro USB και μία HDMI. Για αποθηκευτικό χώρο έχει μια Micro SD κάρτα και ένα διάυλο (GPIO – General Purpose Input / Output) για πρόσθετα κυκλώματα που επεκτείνουν τις λειτουργίες του υπολογιστή.

### Sense HAT

Ένα τέτοιο



πρόσθετο (HAT – Hardware Attached on Top) που μεταξύ ενός joystick, άλλων χρήσιμων αισθητήρων (θερμοκρασίας, μαγνητόμετρο, επιταχυνσιόμετρο, υγρόμετρο, βαρόμετρο, και γυροσκόπιο) διαθέτει και μια οθόνη από LED (8x8) είναι το SenseHAT. Αναπτύχθηκε στα πλαίσια αποστολής της Ευρωπαϊκής Υπηρεσίας Διαστήματος, της Διαστημικής Υπηρεσίας του Ηνωμένου Βασιλείου και του Ιδρύματος Raspberry Pi που πραγματοποιεί πειράματα με δύο αντίγραφα αυτής της πλατφόρμας, τα οποία φιλοξενούνται στο Διεθνή Διαστημικό Σταθμό (ISS).

Δε θα εντρυφήσουμε σε περισσότερες τεχνικές λεπτομέρειες, καθώς θα χρησιμοποιήσουμε ένα διαδικτυακό εξομοιωτή του Sense HAT. Ιδιαίτερα θα χρησιμοποιήσουμε τη LED οθόνη, για να προγραμματίσουμε ένα μήνυμα εξόδου και για να εμβυθύνουμε στην αναπαράσταση έγχρωμων εικόνων.

### Γλώσσα προγραμματισμού Python

Η γλώσσα προγραμματισμού Python είναι μία μοντέρνα επαγγελματική γλώσσα προγραμματισμού. Δε χρησιμοποιεί κάποιο περιβάλλον πλακιδίων όπως το Scratch. Σκοπός μας δεν είναι να βουτήξουμε στα βαθιά νερά του προγραμματισμού, αλλά να πάρουμε μια πρώτη γεύση από αυτήν και ένα τυπικό ολοκληρωμένο περιβάλλον ανάπτυξης προγραμμάτων, ώστε να προγραμματίσουμε το μήνυμα εξόδου και την αναπαράσταση γραφικών στη LED οθόνη του Sense HAT.

**Δραστηριότητα 1:** Χρησιμοποιήστε το φυλλομετρητή Chrome (Chromium για το Linux) ή Safari και επισκεφθείτε τη διεύθυνση: <https://trinket.io/sense-hat>. (Άλλοι φυλλομετρητές παρουσιάζουν πιο φτωχή ή προβληματική 3D αναπαράσταση του εξομοιωτή).

Η οθόνη χωρίζεται στα δύο. Δεξιά παρουσιάζεται ένας εξομοιωτής του Sense HAT. Αριστερά, ένας συντάκτης (editor) προγραμμάτων για να γράψουμε κώδικα σε Python. Σύρετε και περιστρέψτε όπως θέλετε την πλατφόρμα στα δεξιά για να βλέπετε καλά τη LED οθόνη.

**Δραστηριότητα 2:** Στο αριστερό μέρος, κάτω από την καρτέλα main.py σβήστε τον προσυμπληρωμένο κώδικα. Προσθέστε τον ακόλουθο κώδικα:

```
from sense_hat import SenseHat
from time import sleep
sense = SenseHat()
```

Η πρώτη γραμμή εισάγει την προγραμματιστική βιβλιοθήκη SenseHat (προγραμματιστικές εντολές που έχουν σχεδιαστεί ειδικά για την κονσόλα στο δεξί μέρος).

Παρομοίως η δεύτερη γραμμή εισάγει ειδικές εντολές που έχουν να κάνουν με τη διαχείριση χρόνου. Η τρίτη εντολή δημιουργεί ένα νοητό ισοδύναμο αντικείμενο της πλατφόρμας. Απο 'δω και στο εξής, όπου χρησιμοποιούμε τον όρο sense στο πρόγραμμά μας, αυτό έχει όλα τα καλούδια, ιδιότητες και μεθόδους του πραγματικού SenseHat.

**Δραστηριότητα 3:** Προσθέστε τον ακόλουθο κώδικα:

```
sense.show_message("Hello World")
```

Ως αποτέλεσμα η συνάρτηση show\_message() του αντικείμενου sense, θα περάσει το μήνυμα «Hello World» από την οθόνη LED. Δυστυχώς, η εντολή δέχεται χωρίς πρόβλημα μόνο λατινικούς χαρακτήρες.

**Αλλάξτε το περιεχόμενο του μηνύματος.**

**Δραστηριότητα 4:** Η προηγούμενη εντολή δέχεται και άλλες παραμέτρους. (Τις βλέπουμε από το προγραμματιστικό περιβάλλον, που επεξηγεί το συντακτικό, μετά το αντικείμενο "sense.", αν περιμένουμε μετά την εισαγωγή της τελείας). Με ένα κόμμα μετά τα διπλά εισαγωγικά θέστε ένα νούμερο που περιγράφει το scroll speed (την ταχύτητα με την οποία περνάει το μήνυμα από την οθόνη – όσο πιο μικρό το νούμερο τόσο πιο γρήγορα), όπως στο παράδειγμα:

```
sense.show_message("Hello from Karlovassi",0.05)
```

**Τροποποιήστε την ταχύτητα εμφάνισης του μηνύματος.**

**Δραστηριότητα 5:** Προκειμένου να «ανάψουμε» ένα συγκεκριμένο LED, θα χρησιμοποιήσουμε τη συνάρτηση set\_pixel(). Παρατηρούμε ότι δέχεται 5 παραμέτρους ως εξής: **sense.set\_pixel(x,y,r,g,b)**.

Οι πρώτες δύο παράμετροι αναφέρονται στο ζεύγος συντεταγμένων που απαιτείται για να προσδιοριστεί ποιο συγκεκριμένο pixel από τα 64 του 8x8 πίνακα LED θέλουμε να ανάψουμε. Η αρχή του συστήματος αξόνων είναι πάνω αριστερά, όπου το πρώτο πάνω αριστερά pixel έχει συντεταγμένες x,y = (0,0), ενώ το τελευταίο κάτω δεξιά x,y = (7,7). Συνεπώς, οι επιτρεπτές τιμές για κάθε μία από τις δύο πρώτες παραμέτρους προέρχονται από το κλειστό διάστημα [0,7].

Οι άλλες τρεις απαραίτητες παράμετροι αφορούν στο χρωματικό μοντέλο RGB. Χρειάζεται να τεθεί μια τιμή για κάθε ένα από τρία βασικά χρώματα (Κόκκινο, Πράσινο, Μπλε). Αυτό το χρωματικό μοντέλο χρησιμοποιεί 8 bits για την κωδικοποίηση απόχρωσης για κάθε βασικό χρώμα, οπότε προκύπτουν για κάθε χρώμα  $2^8=256$  διαφορετικές τιμές από το σύνολο [0,255]. Το 0 αντιστοιχεί σε απουσία αυτού του χρώματος και το 255 στην πιο έντονη απόχρωση που μπορεί να πάρει. Για παράδειγμα, για να φωτιστεί το pixel της 2ης στήλης και 3ης γραμμής με έντονο μπλε χρώμα θα χρειαστεί η ακόλουθη εντολή:

```
sense.set_pixel(1,2,0,0,255)
```

**«Φωτίστε» το pixel της 3ης στήλης και 4ης γραμμής με έντονο κόκκινο χρώμα.**

**«Φωτίστε» το pixel της 4ης στήλης και 5ης γραμμής με έντονο πράσινο χρώμα.**

**«Φωτίστε» το pixel της 5ης στήλης και 6ης γραμμής με έντονο κίτρινο χρώμα.**

Υπόδειξη: Το Κίτρινο χρώμα θα το δημιουργήσετε από την κατάλληλη ανάμιξη δύο βασικών χρωμάτων.

**Δραστηριότητα 6:** Παρατηρούμε ότι αμέσως μετά το μήνυμα με τους εμφανιζόμενους χαρακτήρες, εμφανίζονται τα αναμμένα pixels στη LED οθόνη. Αν θέλουμε να μεσολαβεί ένα μικρό κενό χρονικό διάστημα, μπορούμε να χρησιμοποιήσουμε τη συνάρτηση `sleep()` της βιβλιοθήκης `time`. Η απαιτούμενη παράμετρος είναι ο χρόνος που θα περιμένει ο επεξεργαστής του Raspberry Pi σε δευτερόλεπτα, μέχρι να εκτελέσει την επόμενη εντολή. Έτσι, αν θέλω μεταξύ μηνύματος και αναμμένων pixel να μεσολαβούν 2 δευτερόλεπτα θα προσθέσω ενδιάμεσα την προαναφερόμενη εντολή ως εξής:

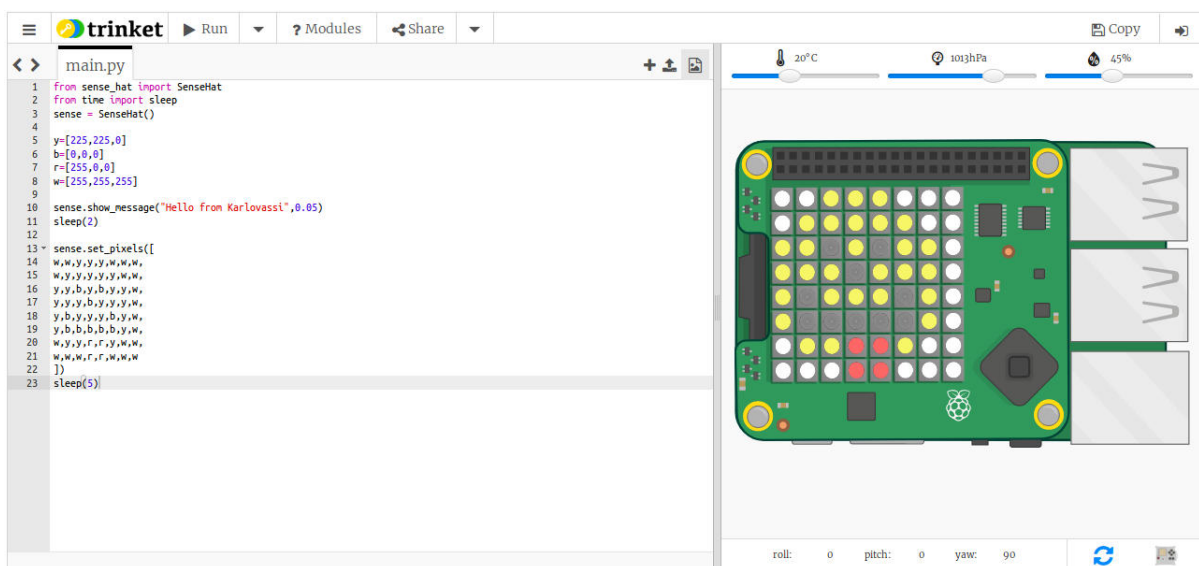
```
sense.show_message("Hello from Karlovassi",0.05)
sleep(2)
sense.set_pixel(1,2,0,255)
```

Εισάγετε χρονοκαθυστερήσεις μεταξύ των εντολών `set_pixel()`, ώστε να μην «ανάβουν» όλα ταυτόχρονα.

**Δραστηριότητα 7:** Για να εμφανιστεί μία εικόνα στον πίνακα LED διαστάσεων 8x8 pixels, όπως στη διαδικτυακή εφαρμογή ρετρό γραφικών `PixilArt` (βλ. προηγούμενο φύλλο εργασίας) θα χρειαστεί να φωτίσουμε 64 pixels. Επιπλέον, εφόσον θέλουμε να είναι έγχρωμη, θα χρειαστεί για κάθε ένα από αυτά να δώσουμε τις τρεις παραμέτρους σύμφωνα με το χρωματικό μοντέλο RGB. Δηλαδή θα χρειαστούμε συνολικά  $64 \times 3 = 192$  διαφορετικές τιμές αποχρώσεων. Για να μειώσουμε το φόρτο εργασίας μας μπορούμε να αντιστοιχίσουμε από πριν ένα γράμμα με συγκεκριμένη χρωματική απόχρωση. Αυτό θα το επιτύχουμε δηλώνοντας μεταβλητές στην αρχή του προγράμματος, μετά τις εισαγωγές βιβλιοθηκών. Οι μεταβλητές αυτές θα αποθηκεύουν από τρεις ακέραιες τιμές (για κάθε βασικό χρώμα) από το  $[0,255]$ . Δηλαδή, όπως λέμε στον προγραμματισμό θα δηλώσουμε για κάθε απόχρωση μια μεταβλητή πίνακα. Για παράδειγμα, για το κίτρινο, το μαύρο, το κόκκινο και το λευκό αντίστοιχα, θα χρειασθούν οι ακόλουθες δηλώσεις μεταβλητών:

```
y=[225,225,0] # Απαλό κίτρινο
b=[0,0,0] # Μαύρο
r=[255,0,0] # Κόκκινο
w=[255,255,255] # Άσπρο
```

Ακολούθως θα χρειαστεί η εντολή `sense.set_pixels()`, που ως παράμετρο δέχεται ένα πίνακα (συμβολίζεται με `[]`), που περιέχει 64 διαφορετικές τιμές (που διαχωρίζονται με κόμματα). Αν θέλετε μια χαμογελαστή φατσούλα 8x8 pixels ο συνολικός κώδικας Python θα πρέπει να έχει διαμορφωθεί κάπως έτσι:



**Δημιουργήστε τη δική σας εικόνα.**

Το έργο με τίτλο «Φύλλο Εργασίας 5: Αναπαράσταση έγχρωμης εικόνας» από τον δημιουργό Δημήτριο – Αδαμάντιο Δρίτσα (dritsasda [at] gmail [τελεία] com, homoinformaticus.eu) διατίθεται με την άδεια Creative Commons Αναφορά Δημιουργού – Μη Εμπορική Χρήση – Παρόμοια Διανομή 4.0 Διεθνές. Τα σχεδιαγράμματα του Raspberry Pi και του Sense Hat διατίθενται με την άδεια Creative Commons Αναφορά Δημιουργού – Μη Εμπορική Χρήση – Παρόμοια Διανομή 3.0 Διεθνές. Το παρόν φύλλο εργασίας έχει δημοσιευτεί και στα πλαίσια της εισήγησης του εκπαιδευτικού σεναρίου «Αναπαράσταση εικόνας. Από ασπρόμαυρα bitmap γραφικά σε έγχρωμα γραφικά LED σε οθόνη μικροϋπολογιστή (Raspberry Pi)». Την πλήρη εισήγηση μπορείτε να βρείτε στα πρακτικά εργασιών του 4ου Διεθνούς Συνεδρίου, για την Προώθηση της Εκπαιδευτικής Καινοτομίας, Λάρισα 12-14 Οκτωβρίου 2018, Γ' τόμος: <https://synedrio.eepek.gr/el/to-synedrio/praktika-synedriou>

