



Τμήμα Ηλεκτρολόγων Μηχανικών
& Μηχανικών Υπολογιστών

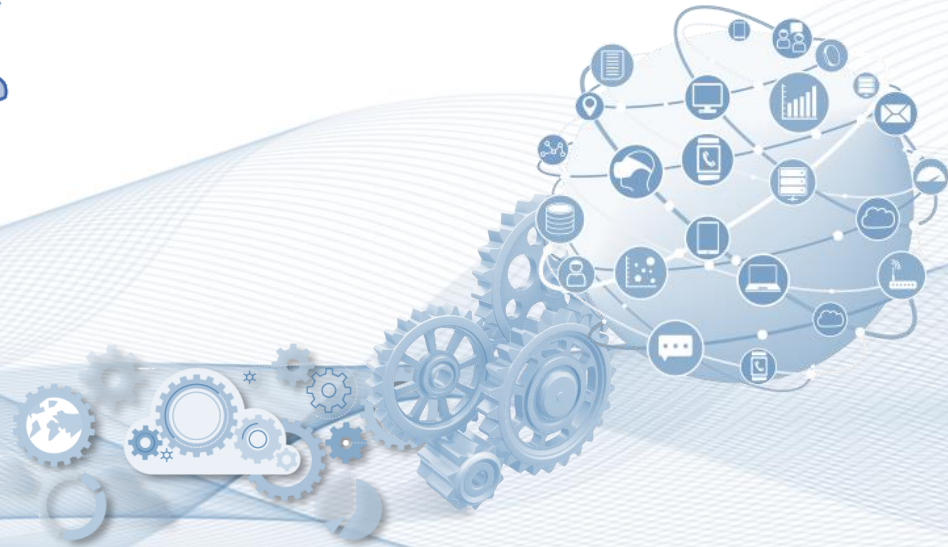
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ

ΤΕΧΝΙΚΕΣ ΠΡΟΓΡΑΜΜΑΤΙΜΟΥ ΥΠΟΛΟΓΙΣΤΩΝ

Διάλεξη I

Scratch - Ξεκινώντας τον Προγραμματισμό

Γιάννης Τζήμας, Καθηγητής



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ
UNIVERSITY of the PELOPONNESE

Τι θα δούμε σήμερα...

- Ο πρώτος μας **στόχος** είναι να πάρετε μία πρώτη ιδέα σε σχέση με:
 - τον **Προγραμματισμό**
 - και την **Επιστήμη των Υπολογιστών**.
- Προχωρώντας θα δούμε πιο αναλυτικά τις **έννοιες** της:
 - **Επιλογής** και
 - της **Επανάληψης**.



Το πρώτο πράγμα που πρέπει πάντα να κάνετε... είναι να ρωτάτε το γιατί;

- **Γιατί** να ασχοληθείτε με την **Πληροφορική**;
 - Για να επεκτείνετε το **μαθηματικό** σας υπόβαθρο.
- Για να ενισχύσετε τη **λογική** στον τρόπο που σκέφτεστε (**logical thinking**).
- Για να κερδίσετε **βαθύτερη γνώση** σε σχέση με την **τεχνολογία**:
 - Πώς να τη **χρησιμοποιείτε**.
 - Πώς να **δημιουργείτε** με αυτή.
 - Πώς να την **αξιοποιείτε** για **υπολογισμούς**.



Γιατί βρεθήκαμε όλοι μαζί εδώ;

- Για να μάθουμε να **προγραμματίζουμε**.
- Για να το κάνουμε αυτό χρειαζόμαστε μία γλώσσα προγραμματισμού, ώστε να εκφράσουμε τις εντολές μας προς τον υπολογιστή.

Ποια γλώσσα όμως είναι η **σωστή**;



Τι θα χρησιμοποιήσουμε για αρχή;

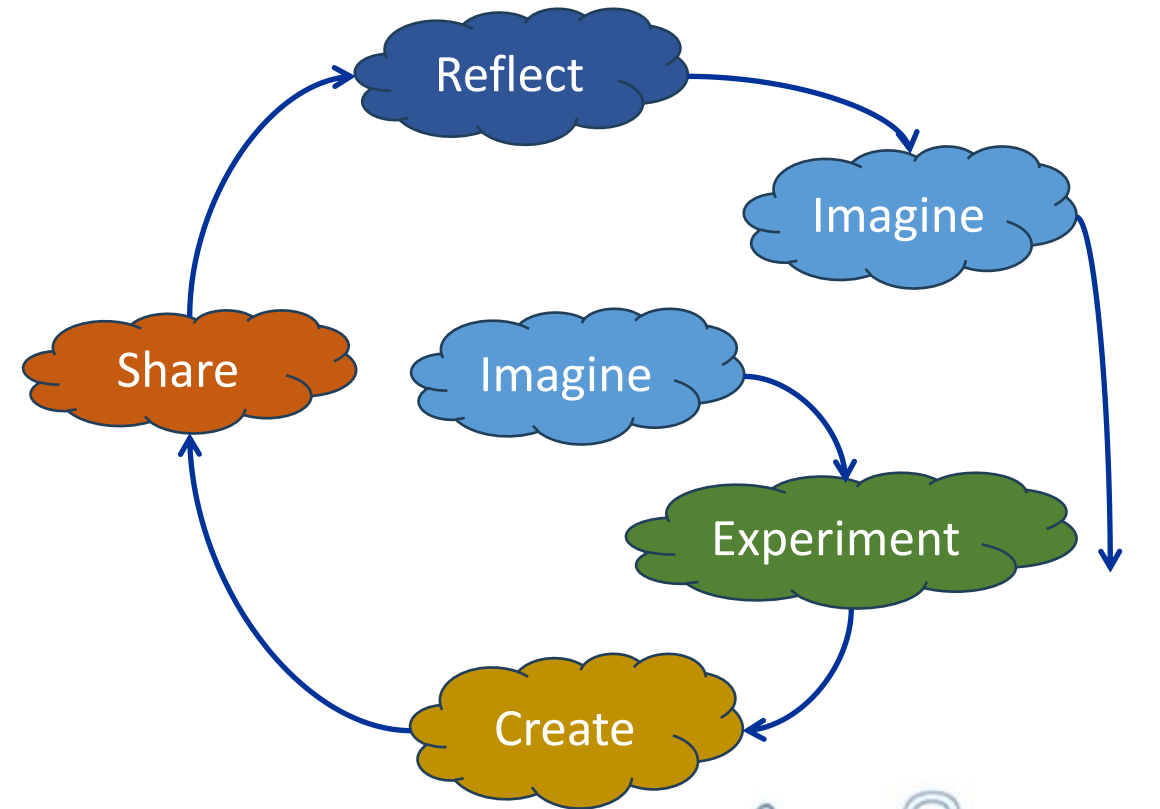
- Αναπτύχθηκε από μερικούς πολύ έξυπνους ανθρώπους @ MIT
- Λειτουργεί σε Linux, Mac, Windows και πλέον σε οποιαδήποτε συσκευή μέσω ενός Web Browser.

SCRATCH



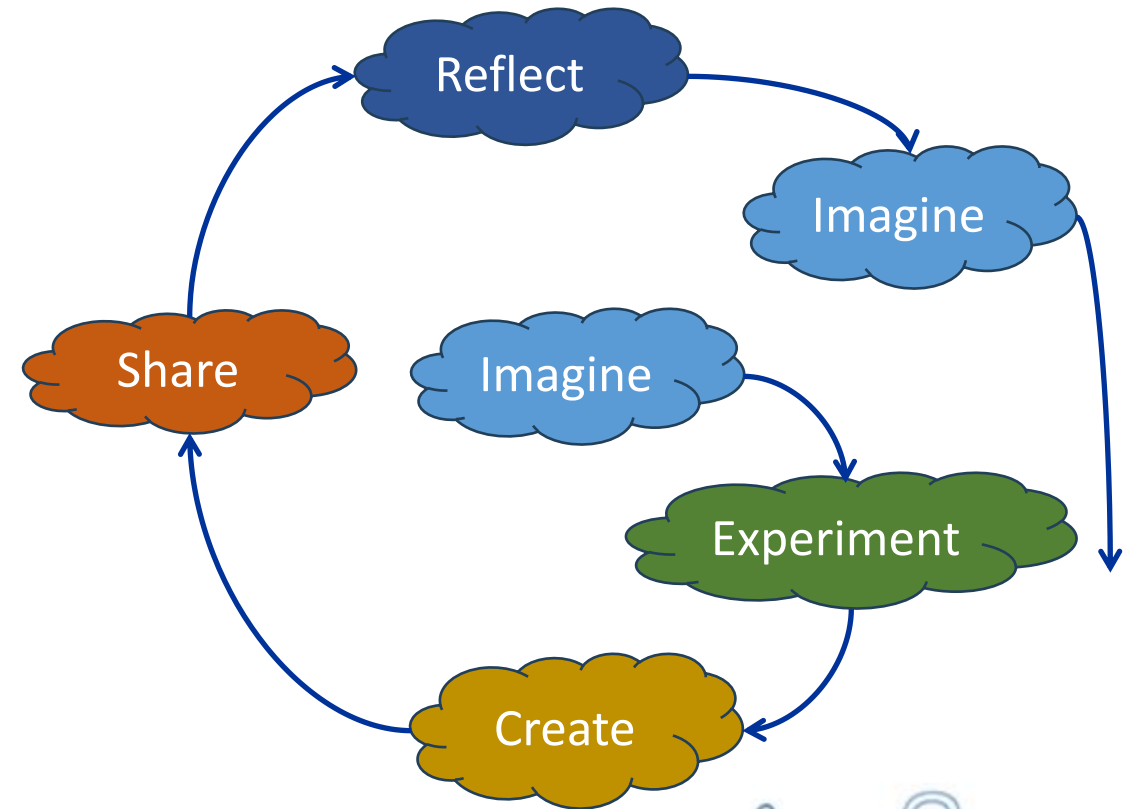
Γιατί Scratch; (1/2)

- Εντυπωσιακό
- Δωρεάν
- Χρησιμοποιείται από εκατομμύρια ανθρώπους παγκόσμια



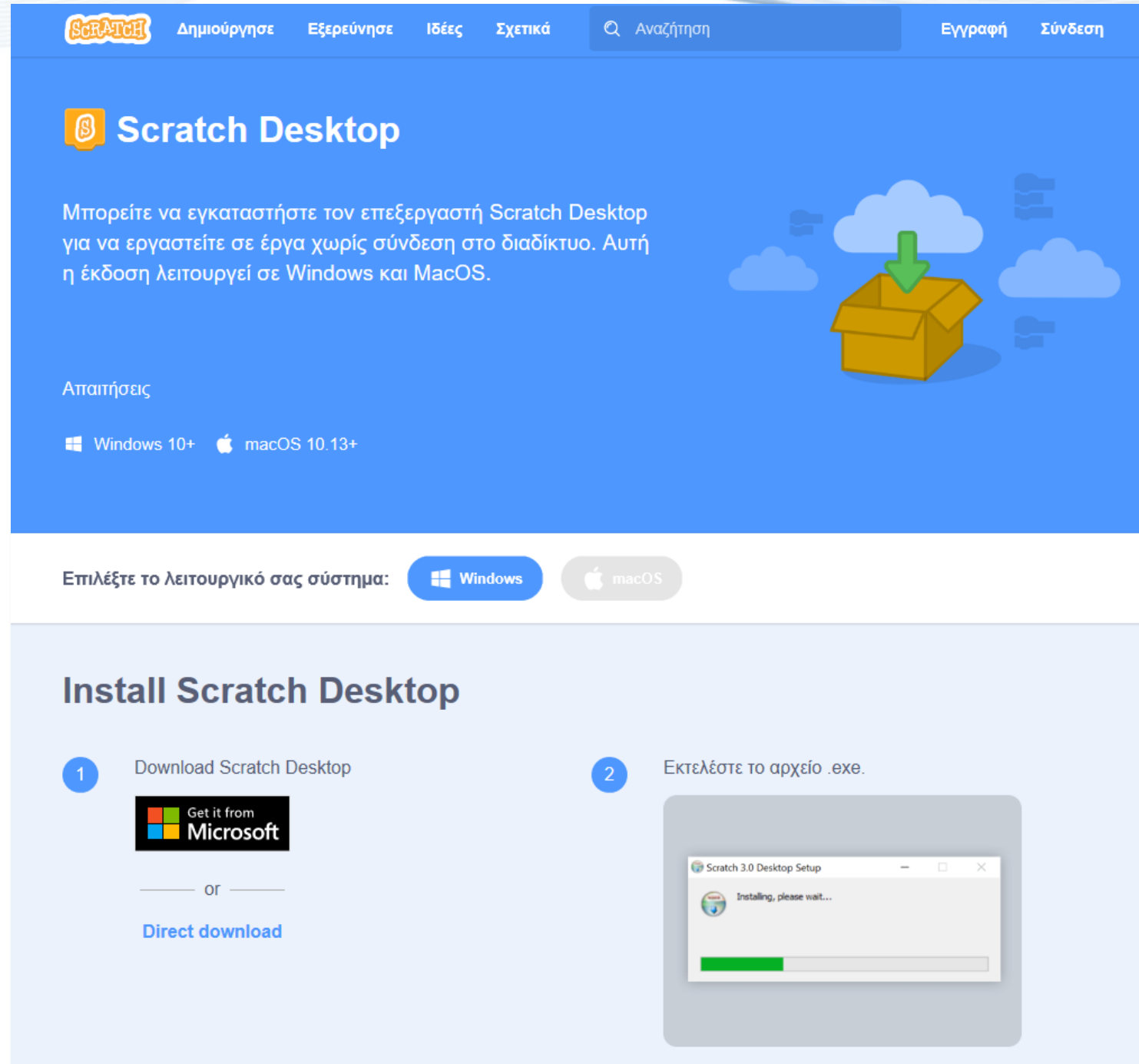
Γιατί Scratch; (2/2)

- Visual, drag & drop (λιγότερη πληκτρολόγηση, περισσότερες δοκιμές)
- Μείξη διαφορετικών μέσων (γραφικά, ήχος και βίντεο) “scratching”
- Διαμοίραση και συνεργασία
- **Low floor, high ceiling, and wide walls**
 - Εύκολο από την αρχή.
 - Δεν είναι παιχνίδι – μπορεί να λύσει σύνθετα προβλήματα.
 - Επεκτάσιμο σε διάφορα πεδία εφαρμογών.



Εγκατάσταση!

- <http://scratch.mit.edu>
- Για να παίζετε στο PC σας μπορείτε να κατεβάσετε την offline από εδώ:
<https://scratch.mit.edu/download>



Scratch Δημιουργήσε Εξερεύνησε Ιδέες Σχετικά Αναζήτηση Εγγραφή Σύνδεση

Scratch Desktop

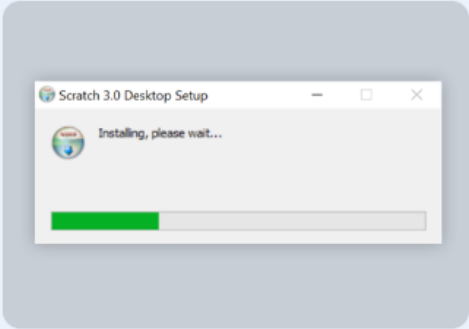
Μπορείτε να εγκαταστήσετε τον επεξεργαστή Scratch Desktop για να εργαστείτε σε έργα χωρίς σύνδεση στο διαδίκτυο. Αυτή η έκδοση λειτουργεί σε Windows και MacOS.

Απαιτήσεις

Windows 10+ macOS 10.13+

Επιλέξτε το λειτουργικό σας σύστημα: Windows macOS

Install Scratch Desktop

- 1 Download Scratch Desktop
Get it from Microsoft
or
Direct download
- 2 Εκτελέστε το αρχείο .exe.


...and Scratch!!!

The image shows the Scratch Desktop application window. The title bar reads "Scratch Desktop" and includes standard window controls. The menu bar contains "Scratch", "File", "Edit", "Tutorials", and "Scratch Project". Below the menu bar are tabs for "Code", "Costumes", and "Sounds".

The interface is divided into several sections:

- Left Panel (Block Palette):** A vertical list of block categories with their respective icons: Motion (blue), Looks (purple), Sound (pink), Events (yellow), Control (orange), Sensing (teal), Operators (green), Variables (orange), and My Blocks (pink).
- Scripts Area:** A large grid for assembling code blocks. It contains several blue blocks from the Motion category, such as "move 10 steps", "turn 15 degrees", "go to random position", "glide 1 secs to random position", "point in direction 90", "point towards mouse-pointer", "change x by 10", "set x to 0", "change y by 10", and "set y to 0". A red text label "Scripts" is overlaid on this area.
- Stage:** A large white area for the project's scene. It features the Scratch cat sprite in the center. A red text label "Stage" is overlaid on the top part of this area.
- Sprite Area:** Located at the bottom right, it includes a "Sprite" dropdown menu set to "Sprite1", and input fields for "x" (0), "y" (0), "Size" (100), and "Direction" (90). There are also "Show" and "Refresh" buttons. Below this is a "Sprites" area with a small thumbnail of the cat sprite and a red text label "Sprites".
- Backdrops:** A section on the far right showing a "Backdrops" list with one item labeled "1".

At the bottom left, there is a blue button with a plus sign and a small icon. At the bottom right, there are two circular buttons: one with a cat head and one with a stage icon.

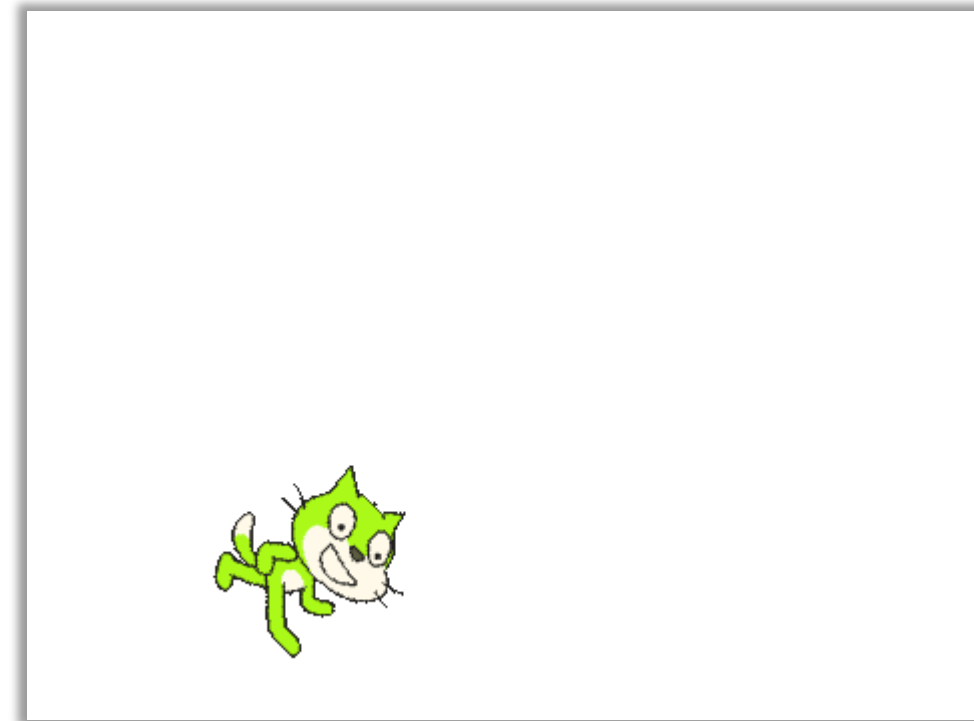
Command
Blocks

Ένα μικρό demo...

```
when clicked  
forever  
  move 30 steps  
  if on edge, bounce  
  wait 0.5 seconds
```

```
when clicked  
forever  
  set instrument to 37  
  set volume to 100 %  
  play note 50 for 0.5 beats  
  play note 53 for 0.5 beats  
  play note 55 for 1 beats  
  play note 50 for 0.5 beats  
  play note 53 for 0.5 beats  
  play note 56 for 0.2 beats  
  play note 55 for 0.8 beats  
  play note 50 for 0.5 beats  
  play note 53 for 0.5 beats  
  play note 55 for 1 beats  
  play note 53 for 0.5 beats  
  play note 50 for 1.5 beats  
wait 0.5 seconds
```

```
when space key pressed  
change color effect by 25
```



'Ενα μικρό demo...

Scratch Desktop

Scratch File Edit Tutorials lecture1_demo1

Code Costumes Sounds

Motion

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer
- change x by 10
- set x to 0
- change y by 10
- set y to 0

when clicked

- forever loop:
 - move 30 steps
 - if on edge, bounce
 - wait 0.5 seconds

when space key pressed

- change color effect by 25
- go to x: 0 y: 0

when clicked

- forever loop:
 - set instrument to 37
 - set volume to 100 %
 - play note 50 for 0.5 beats
 - play note 53 for 0.5 beats
 - play note 55 for 1 beats
 - play note 50 for 0.5 beats
 - play note 53 for 0.5 beats
 - play note 56 for 0.2 beats
 - play note 55 for 0.8 beats
 - play note 50 for 0.5 beats
 - play note 53 for 0.5 beats
 - play note 55 for 1 beats
 - play note 53 for 0.5 beats
 - play note 50 for 1.5 beats
 - wait 0.5 seconds

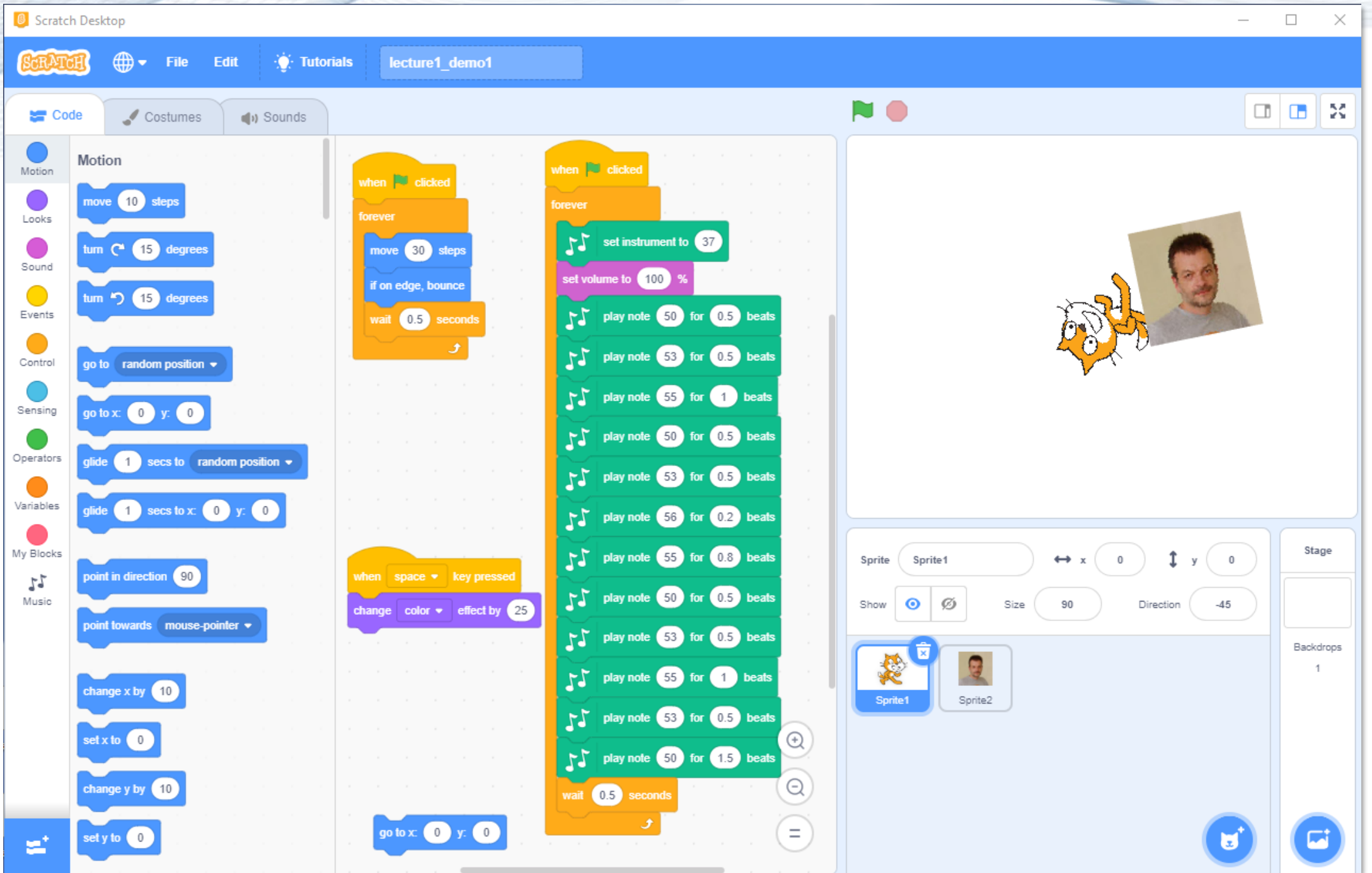
Stage

Sprite Sprite1 x: 0 y: 0

Show Size 90 Direction -45

Sprite1 Sprite2

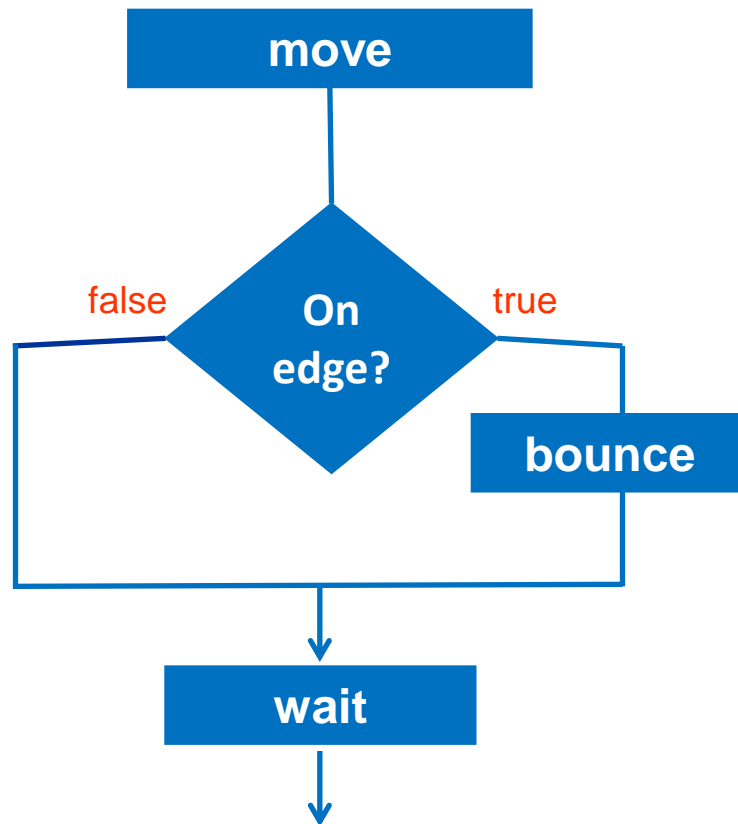
Backdrops 1



Τι είδαμε μέχρι τώρα;

- Στην επίλυση προβλημάτων με υπολογιστή υπάρχουν **τρεις** δεξιότητες τις οποίες θα πρέπει να καλλιεργήσετε:

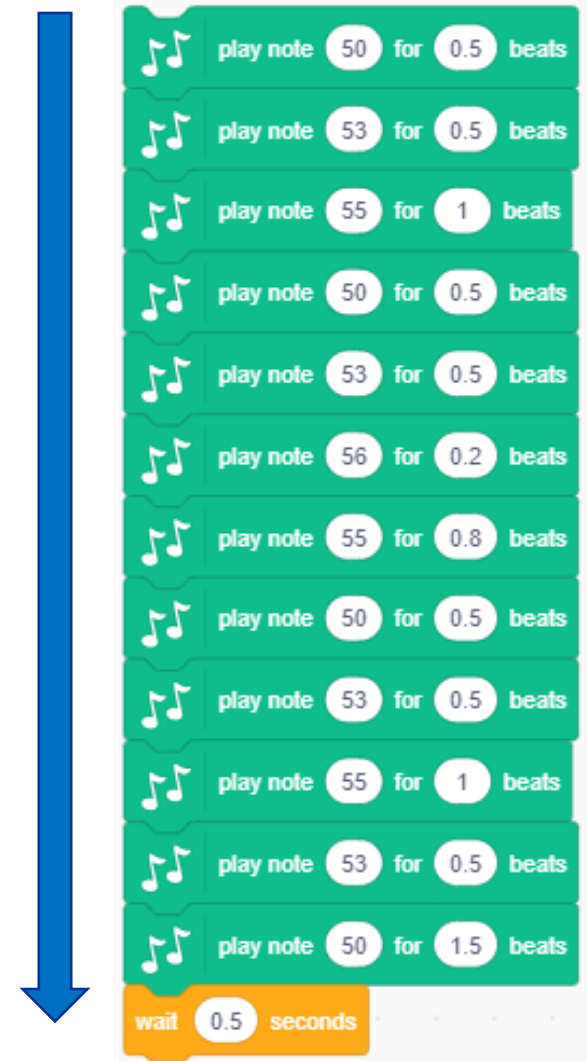
(2) Επιλογή ανάμεσα σε διάφορες εναλλακτικές



(3) Επανάληψη

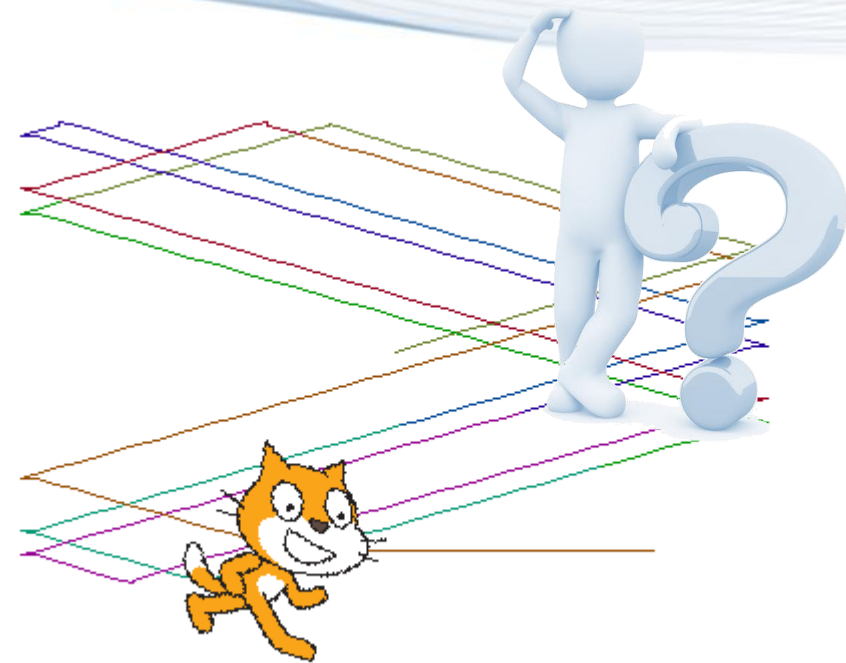


(1) Αλληλουχία
των εντολών



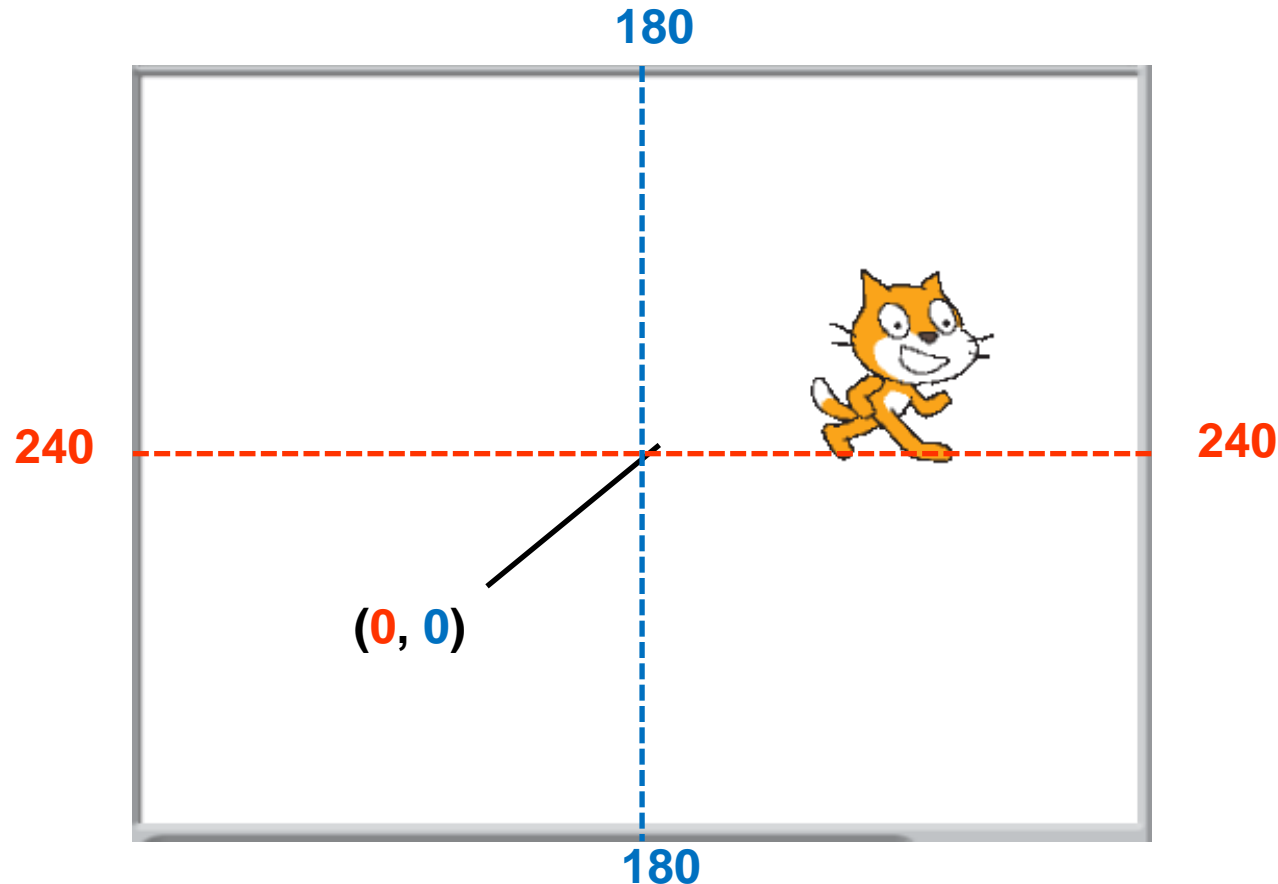
Και λίγη δουλειά για το σπίτι...

- Κατεβάστε και εγκαταστήστε το Scratch: <http://scratch.mit.edu/download>
- Μπείτε στο <http://scratch.mit.edu> και δείτε μερικά projects από άλλους...
- Και μερικές ασκήσεις:
 - Δημιουργήστε ένα πρόγραμμα όπως αυτό που κάναμε μαζί χρησιμοποιώντας ένα διαφορετικό sprite image και βάζοντας τη δικιά σας μουσική (ίσως το αγαπημένο σας τραγούδι). Το πρόγραμμα θα πρέπει να τρέχει για πάντα εκτός και εάν το σταματήσει ο χρήστης.
 - Δημιουργήστε ένα πρόγραμμα το οποίο αρχικά τοποθετεί το sprite στη θέση (0,0), δείχνει σε μία τυχαία κατεύθυνση, κατεβάζει την πένα (pen down) και ξεκινάει να μετακινείται. Όταν φτάνει σε κάποια άκρη θα πρέπει να αλλάζει κατεύθυνση. Επίσης, χρησιμοποιήστε το πλήκτρο SPACE για να αλλάζετε το χρώμα της πένα. Το αποτέλεσμα θα πρέπει να μοιάζει με τη διπλανή εικόνα:



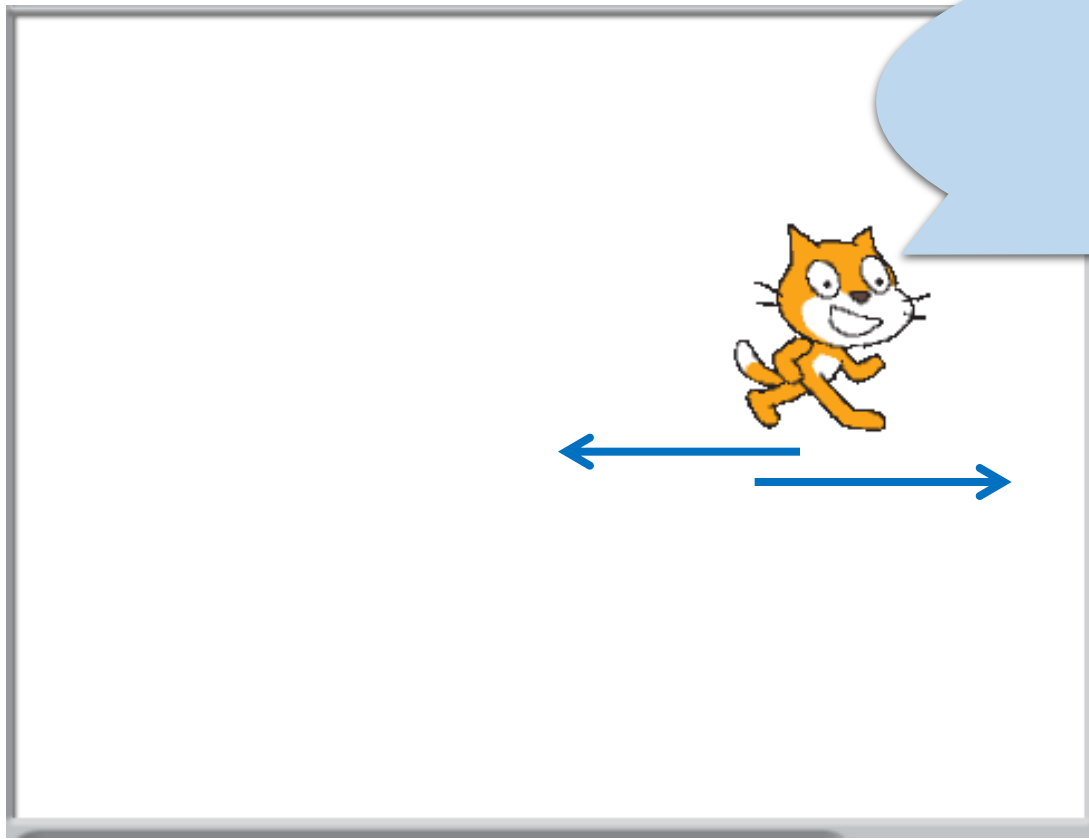
Συστήματα Συντεταγμένων – Τοποθέτηση στη Σκηνή - Stage

- Η θέση των Sprites στη σκηνή βασίζεται σε δύο νούμερα:
 - Τον αριθμό των βημάτων **αριστερά** ή **δεξιά** από το κέντρο.
 - Τον αριθμό των βημάτων **πάνω** ή **κάτω** από το κέντρο.



Θέση = Συντεταγμένες

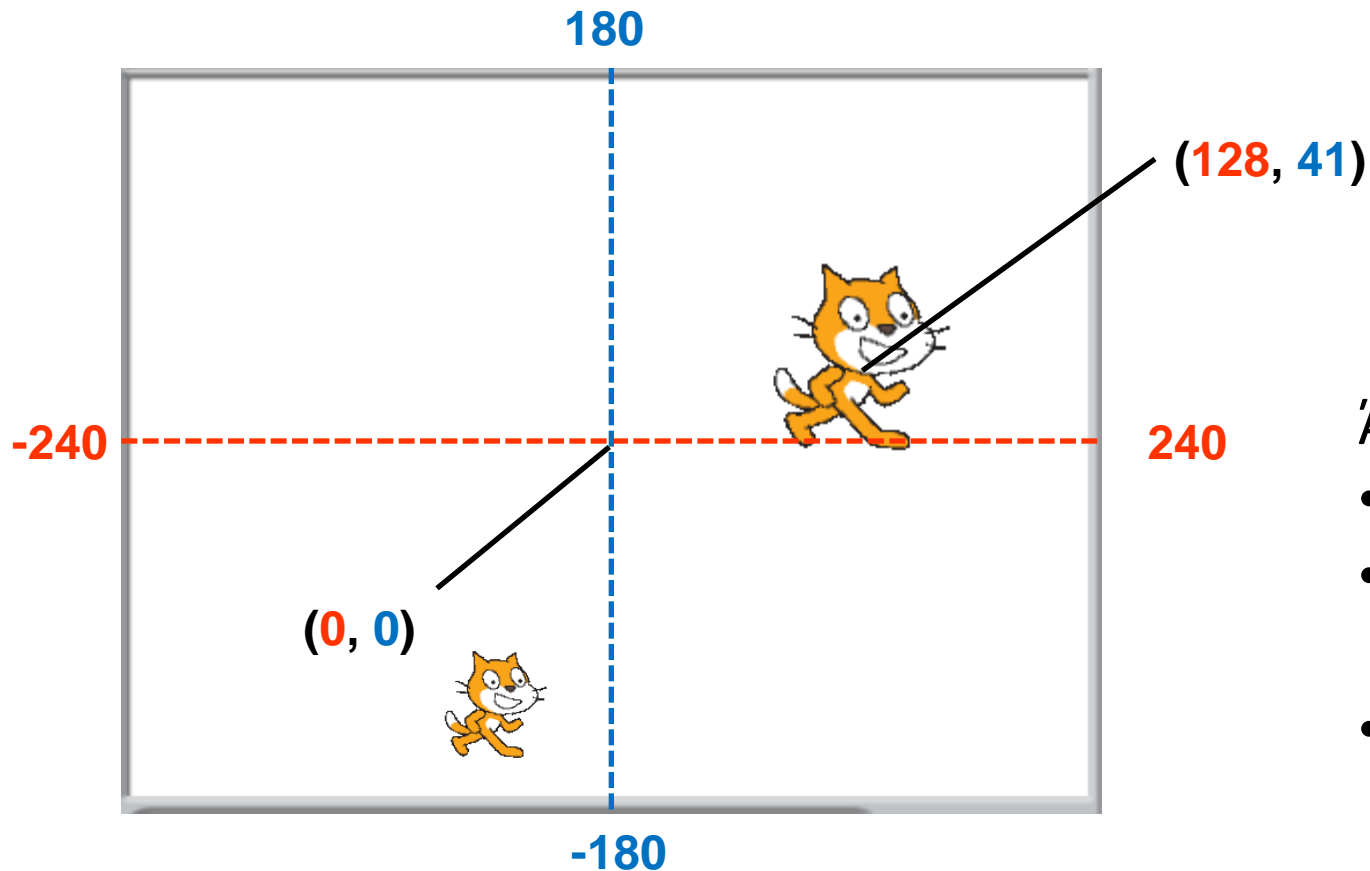
- Η **θέση** είναι γνωστή ως “**συντεταγμένες**”
- Για να γράψουμε καλά προγράμματα χρειαζόμαστε ένα **ακριβές σύστημα συντεταγμένων**, ώστε οι εντολές μας προς τον υπολογιστή να είναι σαφείς.



Όταν λέμε μετακινήσου αριστερά εννοούμε το **δικό μας αριστερά**, ή του **sprite**;

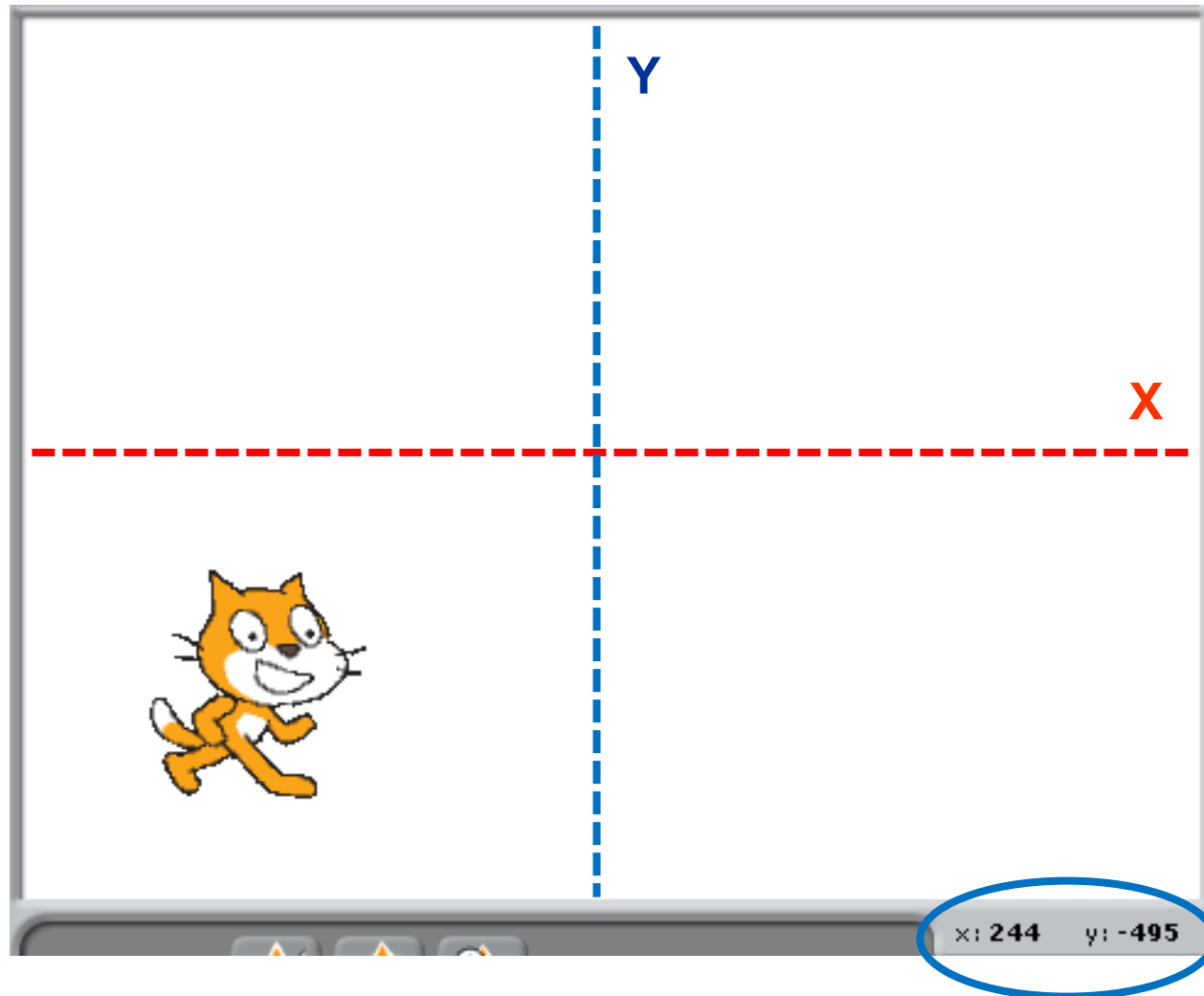
Το Καρτεσιανό Σύστημα Συντεταγμένων

- Χρησιμοποιούμε **θετικούς** και **αρνητικούς** αριθμούς για τις ακριβείς συντεταγμένες.



- Άλλα συστήματα συντεταγμένων:
- Πολικό Σύστημα Συντεταγμένων
 - Κυλινδρικό Σύστημα Συντεταγμένων
 - Σφαιρικό Σύστημα Συντεταγμένων

Συντεταγμένες X και Y



x: 233 y: -368

Blocks εντολών σχετικά με Συντεταγμένες

move 10 steps

turn 15 degrees

turn 15 degrees

point in direction 90

point towards mouse-pointer

go to x: 0 y: 0

go to mouse-pointer

glide 1 secs to x: 0 y: 0

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

y position

direction

Motion

Events

Control

Sensing

Operators

More Blocks

Looks

Sound

Pen

Data

- Πήγαινε στο (x,y)
- Διολίσθησε-glide στο (x,y)
- Άλλαξε το x ή το y κατά...
- Όρισε το x ή το y ίσο με...

go to x: 0 y: 0

glide 1 secs to x: 100 y: 100

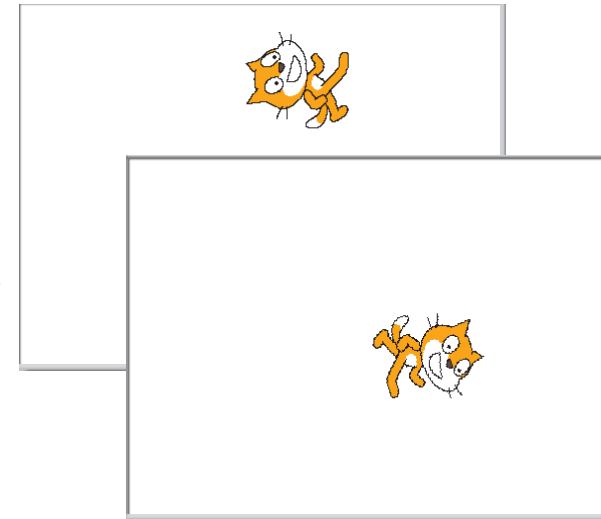
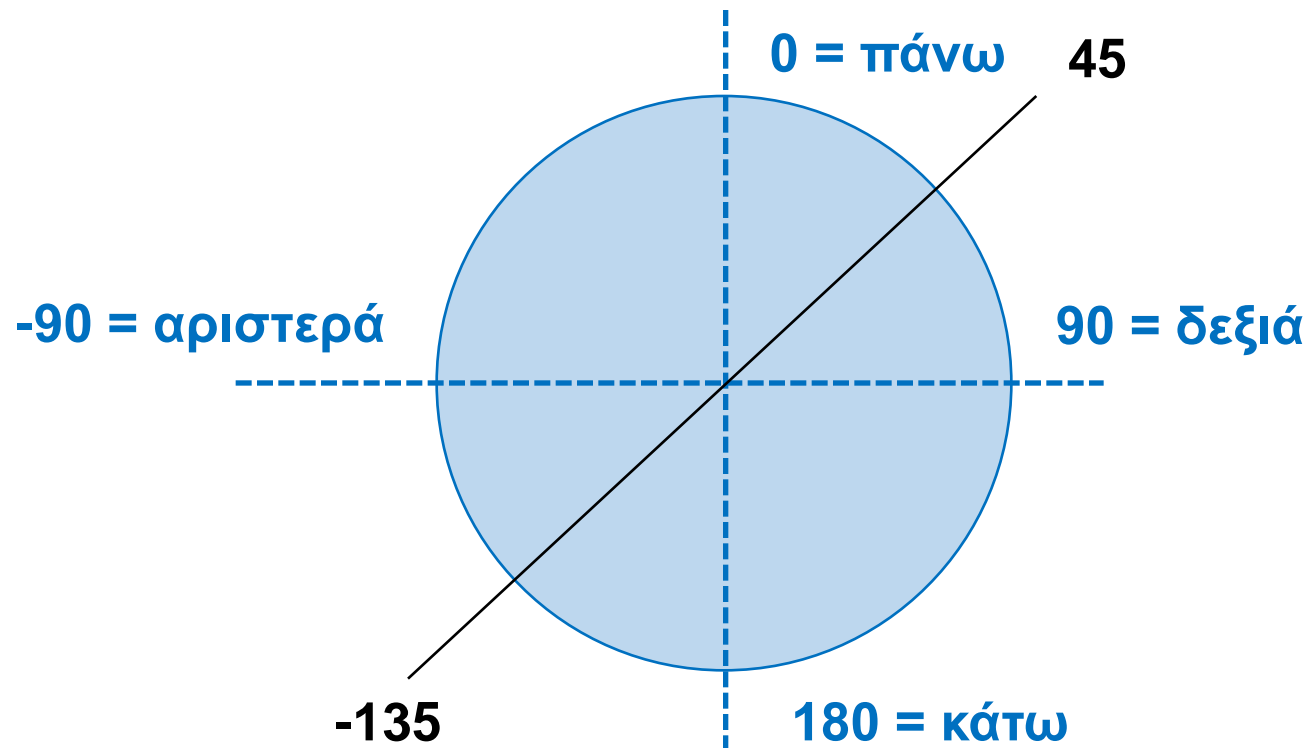
change x by 200

change y by 200

x: 247 y: -484

Κατεύθυνση - Direction

- Τα sprites μπορούν να πάρουν διάφορες κατευθύνσεις.
 - Όπως σε μια πυξίδα έτσι και εδώ, οι κατευθύνσεις αναπαρίστανται σε μοίρες, αλλά...



Blocks εντολών σχετικά με Κατεύθυνση

move 10 steps

turn 15 degrees

turn 15 degrees

point in direction 90

point towards mouse-pointer

go to x: 0 y: 0

go to mouse-pointer

glide 1 secs to x: 0 y: 0

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

y position

direction

Motion

Events

Looks

Control

Sound

Sensing

Pen

Operators

Data

More Blocks

- Στρίψε
 - δεξιόστροφα
 - αριστερόστροφα
 - προς την κατεύθυνση των...
 - προς...



Ένα μικρό demo...

```
when clicked
  go to x: 0 y: 0
  point in direction 90
  forever
    if touching Sprite2 ? then
      say Ζωάρα!!! for 2 secs
      stop this script
    else
      if key up arrow pressed? then
        point in direction 0
        change y by 2
      else
        if key down arrow pressed? then
          point in direction 180
          change y by -2
        else
          if key right arrow pressed? then
            point in direction 90
            change x by 2
          else
            if key left arrow pressed? then
              point in direction -90
              change x by -2
```



```
when clicked
  hide
  set x to pick random -240 to 240
  set y to pick random -180 to 180
  show
```



lecture1_02.sb3

Τι είδαμε μέχρι τώρα;



- Είδαμε το **Καρτεσιανό Σύστημα Συντεταγμένων**. Με τα (x,y) μπορούμε να ορίσουμε **θέσεις**.
- Είδαμε πως μπορούμε να καθορίσουμε την **κατεύθυνση** χρησιμοποιώντας **μοίρες**.
 - Σε αντίθεση με τις πυξίδες που έχουμε $0...360$ μοίρες, στο Scratch έχουμε $0...180$ και $0...-180$
- Μέχρι στιγμής χρειαστήκαμε για ακόμα μία φορά κάθε μία από τις παρακάτω δεξιότητες:
 - **Αλληλουχία** των εντολών
 - **Επιλογή** ανάμεσα σε διάφορες εναλλακτικές
 - **Επανάληψη**

Και λίγη δουλειά για το σπίτι...



- Μερικές ασκήσεις:
 - Δημιουργείστε ένα πρόγραμμα όπως αυτό που κάναμε μαζί βάζοντας τη δικιά σας μουσική. Προσοχή, θα πρέπει να αντικαταστήσετε το “stop script” με ένα “stop all” block για να σταματήσει και ο ήχος. Στη συνέχεια επεκτείνετε το πρόγραμμα ώστε όταν πατάτε το πλήκτρο spacebar το μπολ να πηγαίνει σε νέα θέση.
 - Μετατρέψτε το πρόγραμμα σε παιχνίδι μετατρέποντας τον κώδικα στο μπολ. Το μπολ θα πρέπει να κινείται συνεχώς, ώστε η Scratch να πρέπει να το κυνηγήσει. Για αρχή κάντε την κίνηση απλή. Το μπολ μπορεί να πηγαίνει μόνο οριζόντια στον άξονα X.

Επιλογή - Selection

- Συχνά χρειάζεται να **επιλέξουμε** (*select*) το τι θα κάνουμε βασιζόμενοι σε μία, ή περισσότερες **συνθήκες** (*conditions*).
- Στην πραγματική ζωή:
 - **ΑΝ** (**IF**) πεινάσω **ΤΟΤΕ** (**THEN**) θα φάω κάτι.
 - **ΑΝ** (**IF**) δε βρέξει **ΤΟΤΕ** (**THEN**) θα κάνω ποδήλατο **ΑΛΛΙΩΣ** (**ELSE**) θα κοιμηθώ.
 - **ΑΝ** (**IF**) ο καθηγητής ρωτήσει μια ερώτηση **ΚΑΙ** (**AND**) ξέρω την απάντηση **ΤΟΤΕ** (**THEN**) θα σηκώσω το χέρι.



Η Επιλογή στο Scratch

- Έχουμε ήδη δει μέχρι τώρα την διαδικασία της επιλογής πολλές φορές.



if on edge, bounce



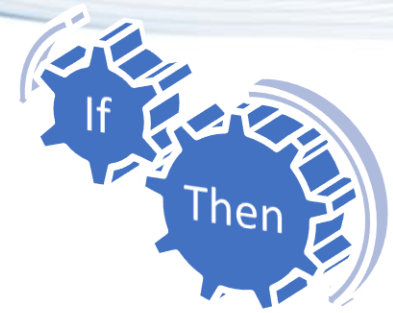
```
if touching Sprite2 ? then
  say ζωάρα!!! for 2 secs
  stop this script
```



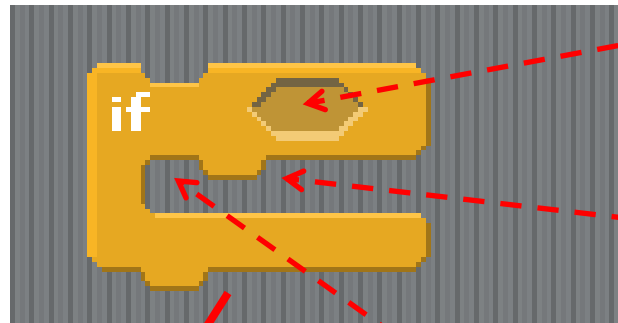
```
if key up arrow pressed? then
  point in direction 0
  change y by 2
else
```

έλεγε τα υπόλοιπα κουμπιά κατευθύνσεων...

Γενική Αρχή #1



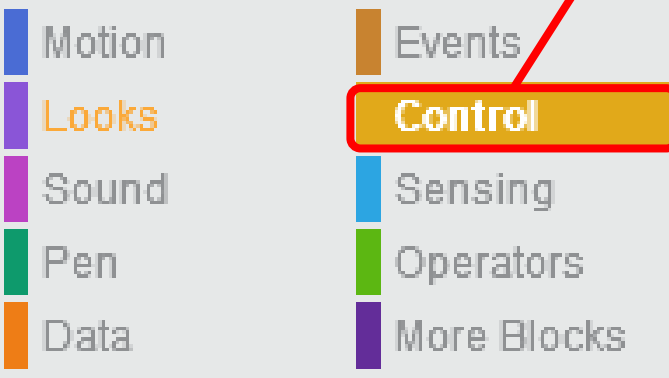
- To **If-Then**



Συνθήκη που θα πρέπει να ελεγχθεί.

Εντολή/ές που θα πρέπει να εκτελεστούν «Αν» (if) η συνθήκη ισχύει (true).

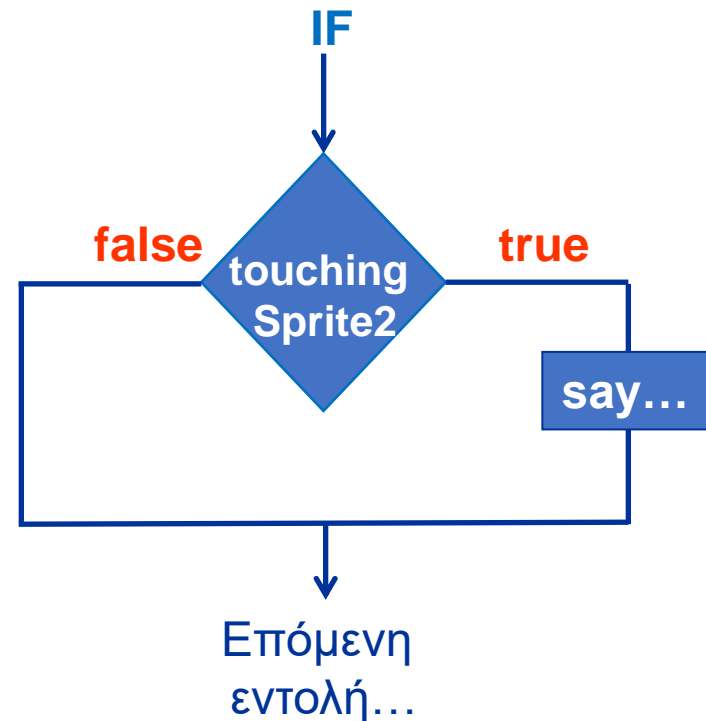
Αναφερόμαστε στις εμπεριεχόμενες εντολές σαν το τμήμα «Τότε» (then).



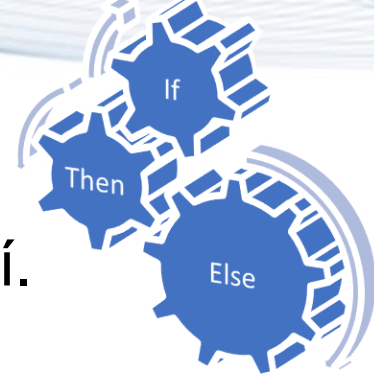
Μοντέλο Εκτέλεσης - Ένα Παράδειγμα...

```
if touching Sprite2 ?  
  say Zώρα!!! for 2 secs  
  stop all
```

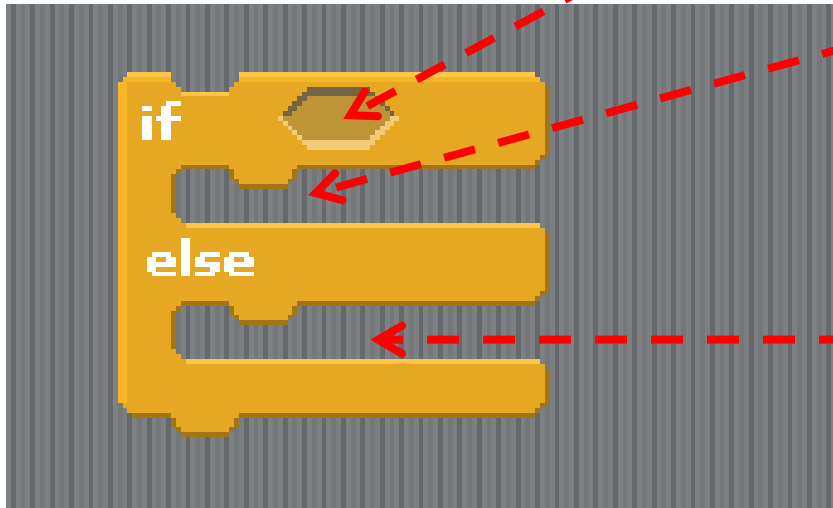
Τι θα συμβεί αν η συνθήκη δεν ισχύει;
Απολύτως τίποτα – Αυτό το τμήμα παραλείπεται.



Γενική Αρχή #2



- Το **If-Then-Else**



Συνθήκη που θα πρέπει να ελεγχθεί.

Εντολή/ές που θα πρέπει να εκτελεστούν «Αν» (if) η συνθήκη ισχύει (true).

«Τότε» (then)

Εντολή/ές που θα πρέπει να εκτελεστούν «Αν» (if) η συνθήκη δεν ισχύει (false).

«Αλλιώς» (else)

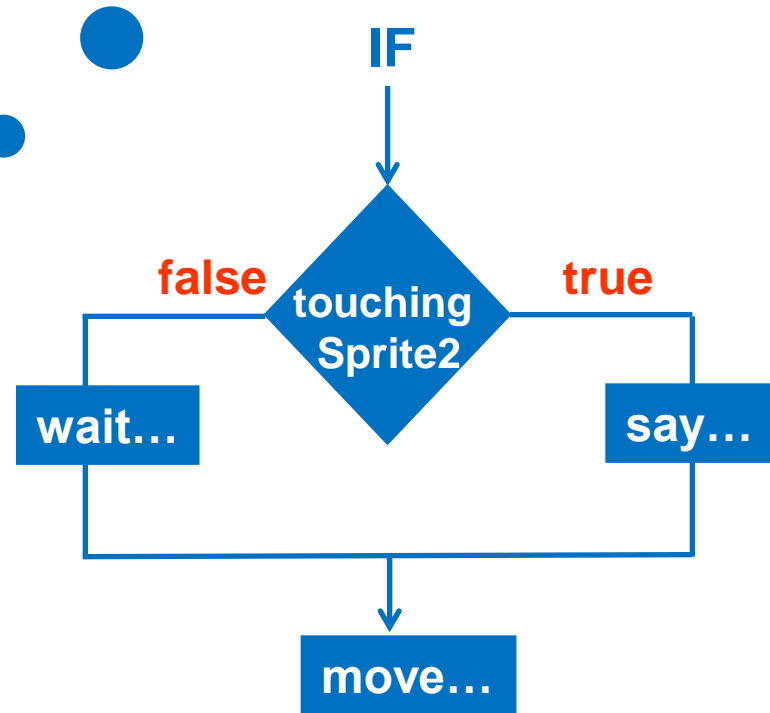
- **Πότε** το χρησιμοποιούμε;

- Όταν υπάρχουν ενέργειες που μπορούν να εκτελεστούν και στις δύο περιπτώσεις...(true, false)

Μοντέλο Εκτέλεσης - Ένα Ακόμα Παράδειγμα...

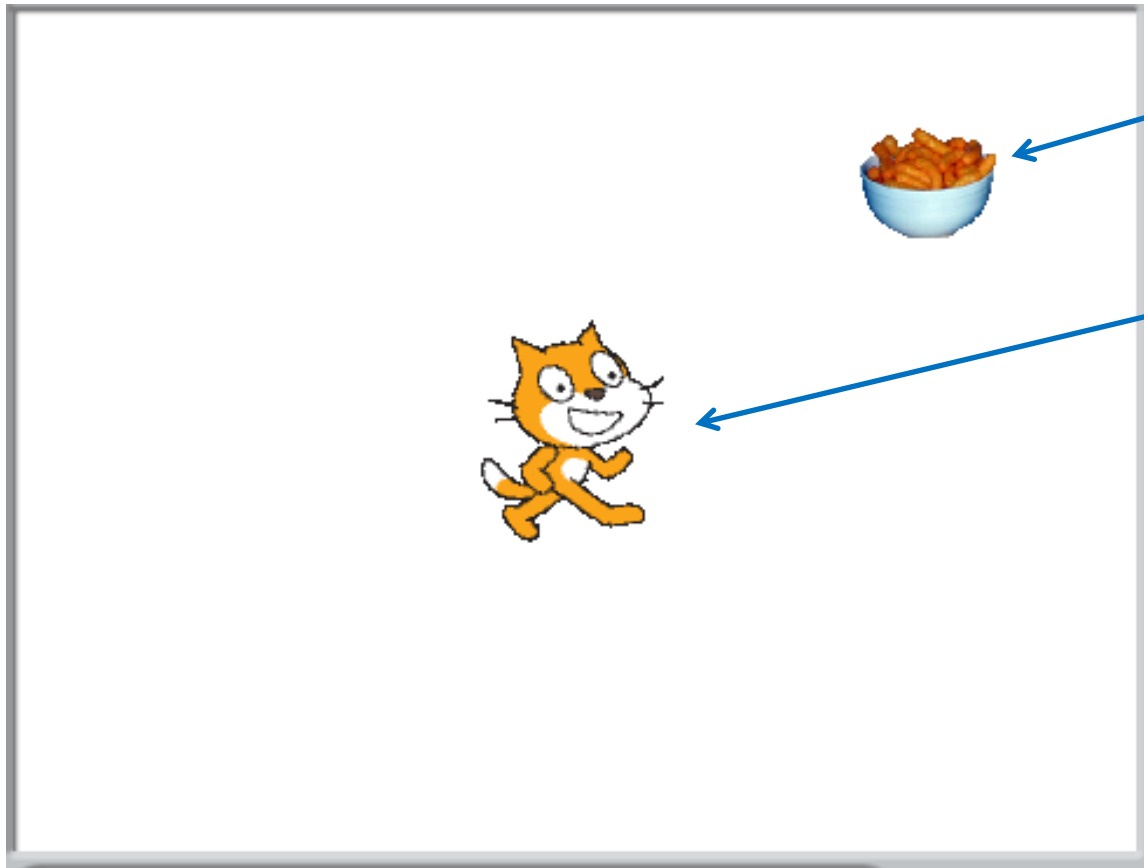
```
if touching Sprite2 ?  
  say Zωάρα!!! for 2 secs  
else  
  wait 1 secs  
move 10 steps
```

Εκτελείται το ένα τμήμα ή το άλλο, αλλά ποτέ και τα δύο μαζί!



Ένα μικρό demo που έχουμε ξαναδεί...

- Από την προηγούμενα φορά είχαμε φτιάξει ένα πρόγραμμα με το οποίο βοηθήσαμε τη Scratch να βρει φαγητό.



Το μπολ εμφανίζεται σε τυχαία θέση.

- Η Scratch εμφανίζεται στο σημείο (0,0).
- Χρησιμοποιούμε τα πλήκτρα για να οδηγήσουμε τη Scratch στο φαγητό.
- Το πρόγραμμα τερματίζεται όταν η Scratch ακουμπήσει το φαγητό.

Ένα μικρό demo που έχουμε ξαναδεί...

```
when clicked
  go to x: 0 y: 0
  point in direction 90
  forever
    if touching Sprite2
      say Zwápa!!! for 2
      stop all
    else
      if key up arrow pressed? then
        point in direction 0
        change y by 2
      else
        if key down arrow pressed? then
          point in direction 180
          change y by -2
        else
          if key right arrow pressed? then
            point in direction 90
            change x by 2
          else
            if key left arrow pressed? then
              point in direction -90
              change x by -2

when clicked
  forever
    set volume to 100 %
    play note 50 for 0.5 beats
    change volume by -50
    play note 53 for 0.5 beats
    set volume to 100 %
    play note 55 for 0.5 beats
    wait 0.5 secs
```

lecture1_03.sb3



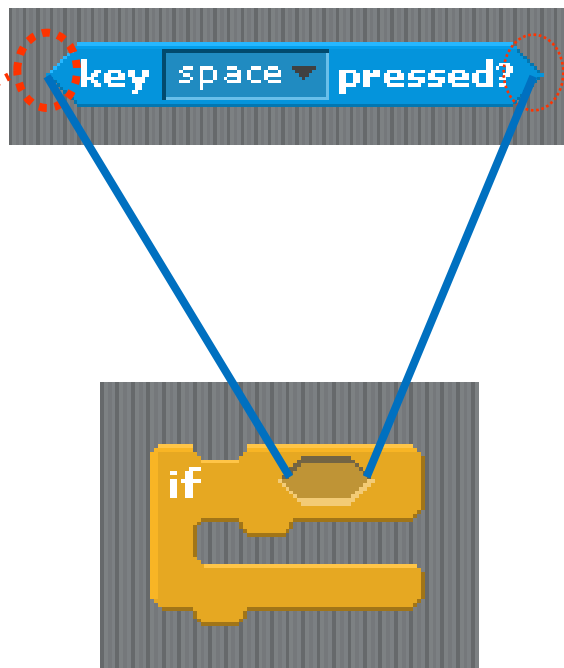
```
when clicked
  hide
  set x to pick random -240 to 240
  set y to pick random -180 to 180
  show

when space key pressed
  hide
  set x to pick random -240 to 240
  set y to pick random -180 to 180
  show
```

Συνθήκες; - Conditions?

The image shows the Scratch Sensing category blocks. At the top, there is a grid of category buttons: Motion, Control, Looks, Sensing (highlighted with a red box), Sound, Operators, Pen, and Variables. Below this, a list of Sensing blocks is shown, including 'touching', 'touching color', 'color is touching', 'ask', 'answer', 'mouse x', 'mouse y', 'mouse down?', 'key space pressed?', 'distance to', 'reset timer', 'timer', 'x position of Sprite1', 'loudness', 'loud?', 'slider sensor value', and 'sensor button pressed?'. A red arrow points from the 'Sensing' button to the 'key space pressed?' block in the list.

Η κατηγορία “Sensing” περιέχει έτοιμα πρότυπα **συνθηκών**. Παρατηρείστε ότι όλες οι συνθήκες οδηγούν σε μία απάντηση ναι ή όχι (true/false).



Παρατηρείστε ότι όσα blocks έχουν **μυτερές άκρες** αφορούν **συνθήκες**.

Δημιουργώντας τις δικές σας συνθήκες...

- Μερικές φορές χρειάζεται να δημιουργήσετε τις δικές σας συνθήκες.
 - Οι σχεδιαστές του Scratch ήταν αρκετά έξυπνοι για να ξέρουν ότι **δε μπορούν να τα προβλέψουν όλα**,
 - έτσι έφτιαξαν ένα **γενικό μηχανισμό** για να δημιουργείτε τις δικές σας συνθήκες.

The image shows the Scratch interface. At the top, a menu contains several categories: Motion, Looks, Sound, Pen, Control, Sensing, Operators, and Variables. The 'Operators' category is highlighted with a red box and labeled 'Τελεστές'. Below the menu, the Operators block palette is visible. It includes arithmetic blocks (+, -, *, /), a 'pick random 1 to 10' block, and logical blocks (<, =, >, and, or, not). A red dashed box highlights the logical blocks, which are labeled 'Σύγκριση' (Comparison) and 'Σύνθεση' (Composition).

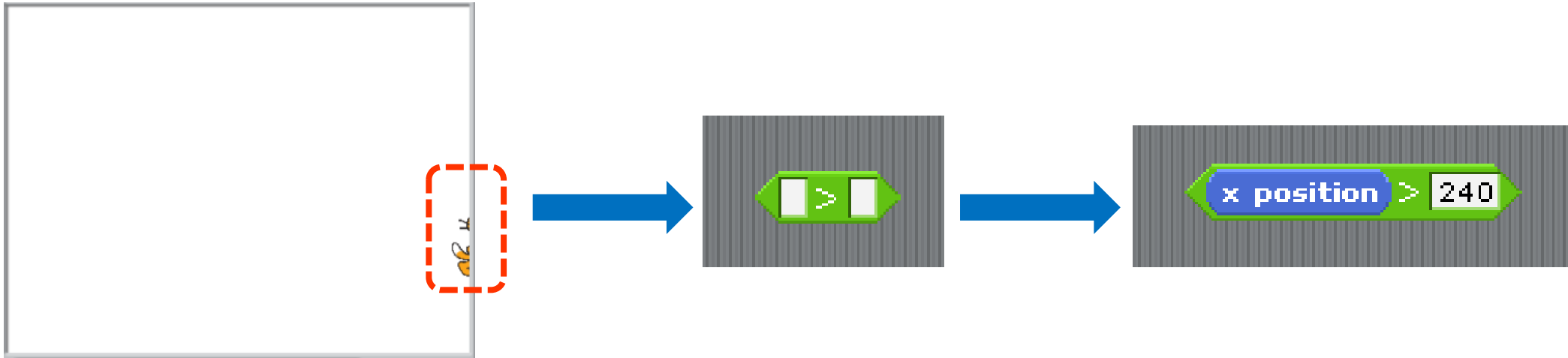
Τελεστές

Σύγκριση

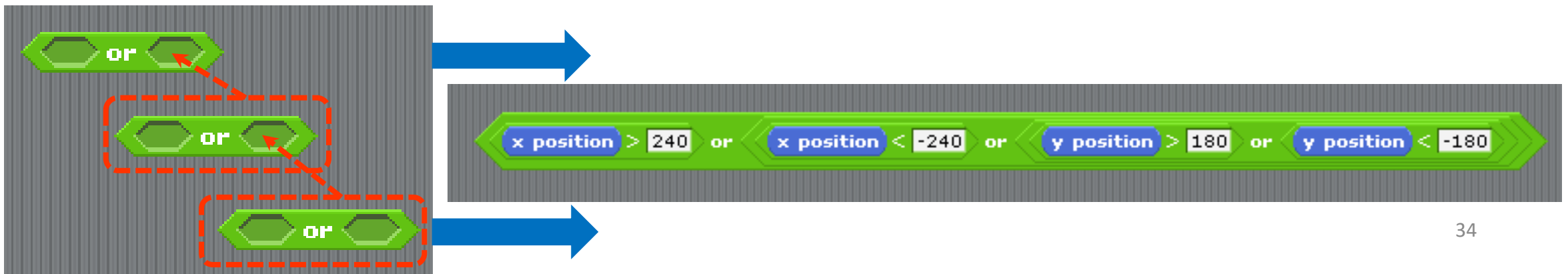
Σύνθεση

Παραδείγματα

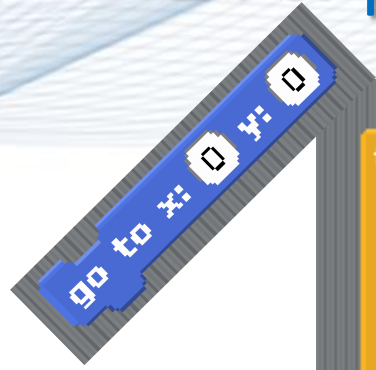
- Είναι η **συντεταγμένη X** του sprite μεγαλύτερη από αυτή της **δεξιάς γωνίας** της σκηνής;



- Είναι η **συντεταγμένες (X, Y)** του sprite εκτός **οποιασδήποτε πλευράς** της σκηνής;
 - Αυτό ισχύει αν $X > 240$ **OR** $X < -240$ **OR** $Y > 180$ **OR** $Y < -180$



Ένα μικρό demo... στην πράξη (1/2)



```
forever
  if <x position < 240>
    move 2 steps
  else
    say x position for 2 secs
    stop script
```



```
forever
  if <x position < 240> or <x position = 240>
    move 2 steps
  else
    say x position for 2 secs
    stop script
```



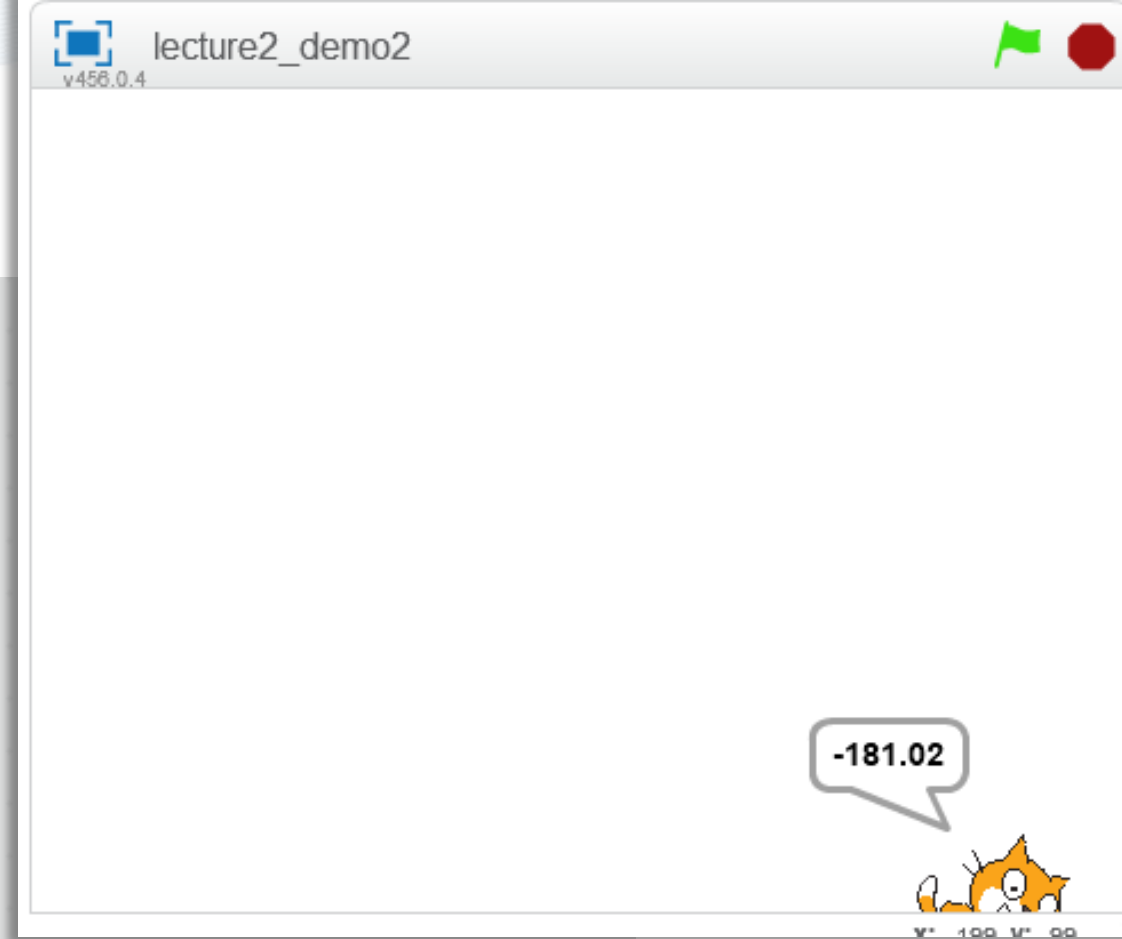
```
forever
  move 2 steps
  if <x position > 240> or <x position < -240> or <y position > 180> or <y position < -180>
    say x position for 2 secs
    say y position for 2 secs
    say direction for 2 secs
    stop script
```

Ένα μικρό demo... στην πράξη (2/2)

go to x: 0 y: 0

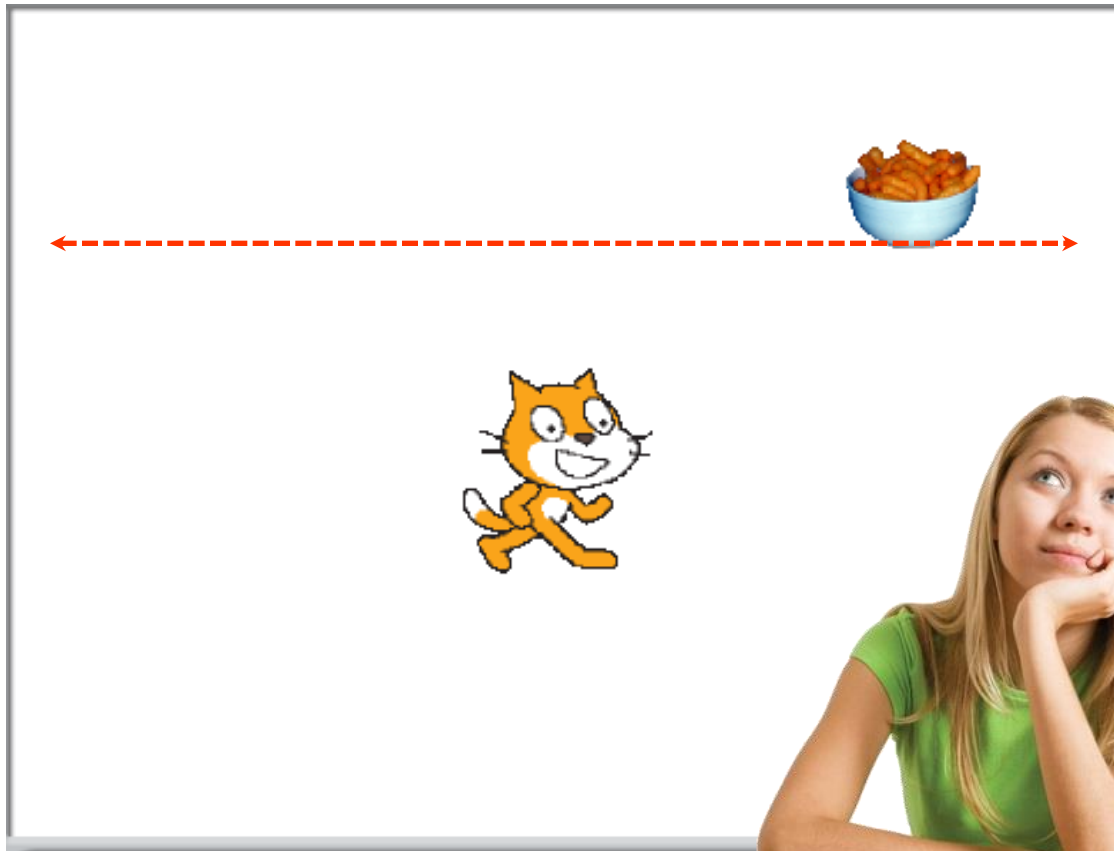
```
forever
  if <x position < 240 or x position = 240> then
    move 2 steps
  else
    say x position for 2 secs
    stop this script
```

```
forever
  move 2 steps
  if <x position > 240 or x position < -240 or y position > 180 or y position < -180> then
    say x position for 2 secs
    say y position for 2 secs
    say direction for 2 secs
    stop this script
```



Βάζοντάς τα όλα σε μία σειρά!

- Στην τελευταία εργασία του προηγούμενου μαθήματος σας ζητήθηκε να φτιάξετε ένα πρόγραμμα στο οποίο το μπολ να κινείται συνεχώς στην κατεύθυνση X.



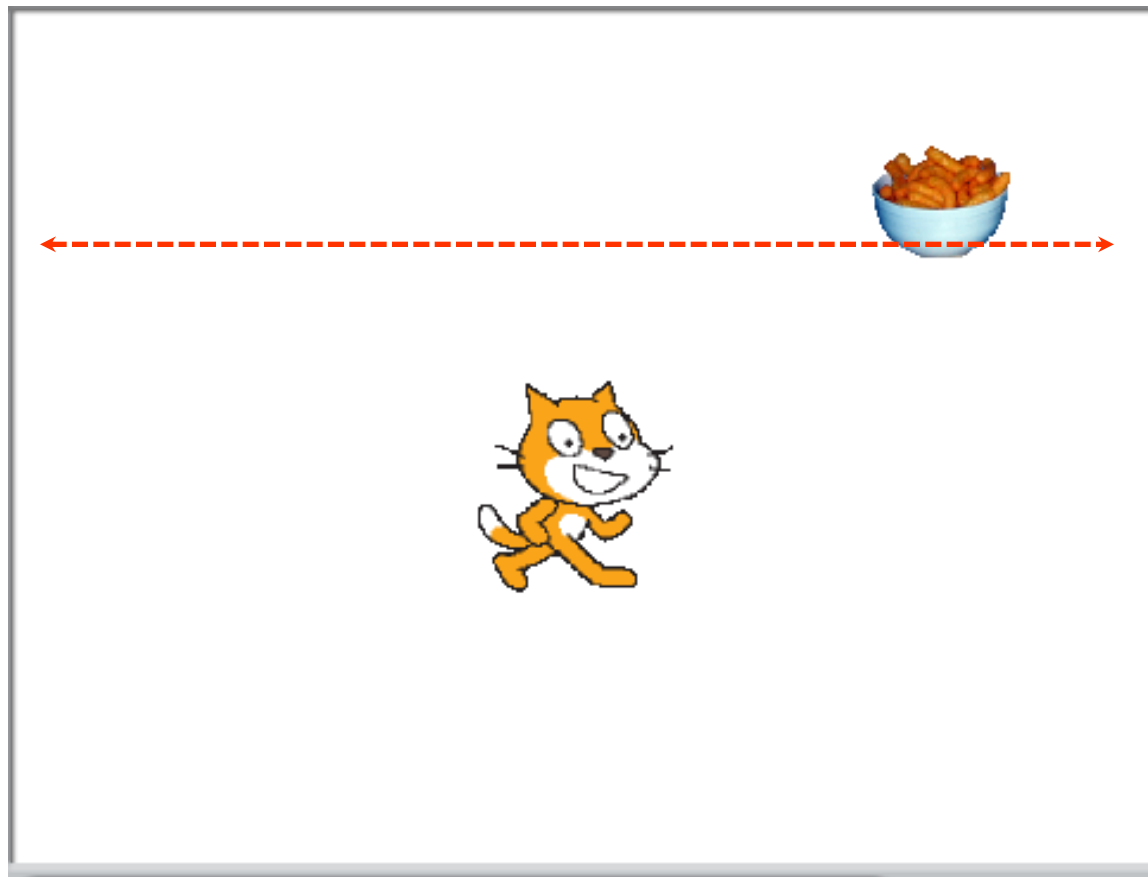
```
forever
  if touching Sprite1 ?
    stop script
  else
    if touching edge ?
      turn 180 degrees
      move 2 steps
    else
      move 2 steps
```



Αυτό θα λειτουργήσει, αρκεί το μπολ να μην ξεκινήσει από τη γωνία. Τότε θα κολλήσει!

Η λύση...

- Μια καλύτερη λύση είναι η ακόλουθη:



```
forever
  if touching Sprite1 ?
    stop script
  else
    if x position > 240
      turn 180 degrees
      set x to 240
      move 2 steps
    else
      if x position < -240
        turn 180 degrees
        set x to 0
        move 2 steps
      else
        move 2 steps
```

```
when clicked
  go to x: 0 y: 0
  point in direction 90
  forever
    if touching Sprite2? then
      say Ζωάρα!!! for 2 secs
      stop all
    else
      if key up arrow pressed? then
        point in direction 0
        change y by 2
      else
        if key down arrow pressed? then
          point in direction 180
          change y by -2
        else
          if key right arrow pressed? then
            point in direction 90
            change x by 2
          else
            if key left arrow pressed? then
              point in direction -90
              change x by -2
```



x: 0
y: 0

```
when clicked
  forever
    set volume to 100 %
    play note 50 for 0.5 beats
    change volume by -50
    play note 53 for 0.5 beats
    set volume to 100 %
    play note 55 for 0.5 beats
    wait 0.5 secs
```

lecture1_05



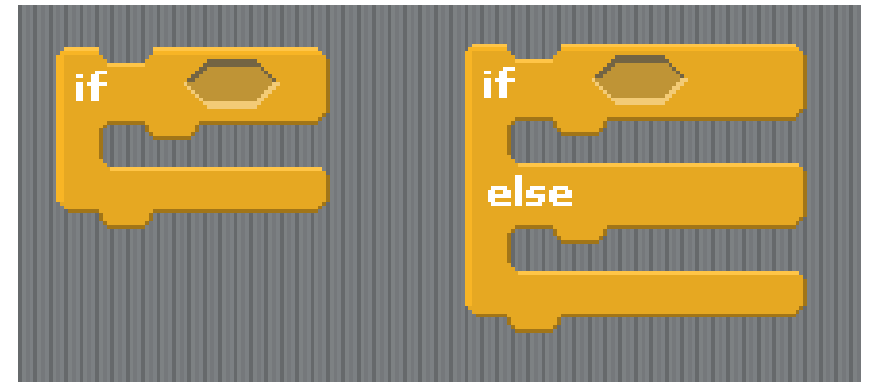
x: -145
y: -101

```
when clicked
  hide
  set x to pick random -240 to 240
  set y to pick random -180 to 180
  point in direction 90
  show
  forever
    if touching Sprite1? then
      stop this script
    else
      if x position > 240 then
        turn 180 degrees
        set x to 240
        move 2 steps
      else
        if x position < -240 then
          turn 180 degrees
          set x to -240
          move 2 steps
        else
          move 2 steps
```

```
when space key pressed
  hide
  set x to pick random -240 to 240
  set y to pick random -180 to 180
  point in direction 90
  show
```

Τι είδαμε μέχρι τώρα;

- Η **επιλογή (selection)** είναι μία από τις **τρεις** προγραμματιστικές δεξιότητες που θα πρέπει να καλλιεργήσετε:
 - **Αλληλουχία** των εντολών
 - **Επιλογή** ανάμεσα σε διάφορες εναλλακτικές
 - **Επανάληψη**
- Στις περισσότερες γλώσσες προγραμματισμού η επιλογή παίρνει τη μορφή **If-Then**, ή **If-Then-Else** εντολών.
- Τέλος, η επιλογή βασίζεται στην αξιολόγηση **συνθηκών (conditions)**. Είναι σημαντικό να καταλάβετε ότι οι συνθήκες είναι **εκφράσεις ή δηλώσεις**, των οποίων η αξιολόγηση έχει σαν αποτέλεσμα το **true** ή **false**.



Και λίγη δουλειά για το σπίτι...

- Και μία **άσκηση**:

- Όπως και στις προηγούμενες ασκήσεις, δημιουργείστε ένα πρόγραμμα το οποίο να περιέχει δύο sprites, τη Scratch και το μπολ.

Όπως και πριν το μπολ θα πρέπει να τοποθετείται σε τυχαίο σημείο στη σκηνή.

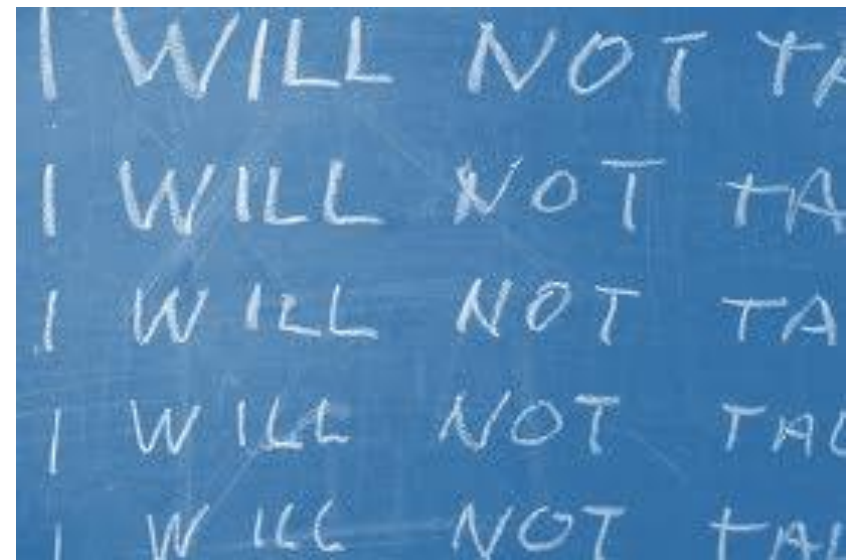
Παρόλα αυτά δε θα οδηγείτε εσείς τη Scratch για να βρει το μπολ, αλλά θα πρέπει να το κάνει μόνη της. Όταν το βρει θα πρέπει να φαίνονται η X και Y συντεταγμένες της και το πρόγραμμα να τελειώνει.

- Ακούστε λίγη μουσική μέχρι η Scratch να βρει το φαγητό 😊
Μπορεί να πάρει κάποια ώρα!



Επανάληψη - Repetition

- Πολλά πράγματα στην πραγματική ζωή επαναλαμβάνονται λόγω της φύσης τους.
 - Συνέχισε να ανακατεύεις μέχρι (until) να γίνει ο απαραίτητος αφρός.
 - Τρέξε μέχρι (until) να πονέσουν τα πόδια σου.
 - Γράψε 100 φορές «δε θα ξαναστείλω SMS την ώρα του μαθήματος»



Η Επανάληψη στο Scratch

- Το έχουμε δει ήδη πολλές φορές μέχρι τώρα!

```
when clicked
  go to x: 0 y: 0
  point in direction 90
  forever
    if x position > 240
      point in direction -90
    else
      move 2 steps
```

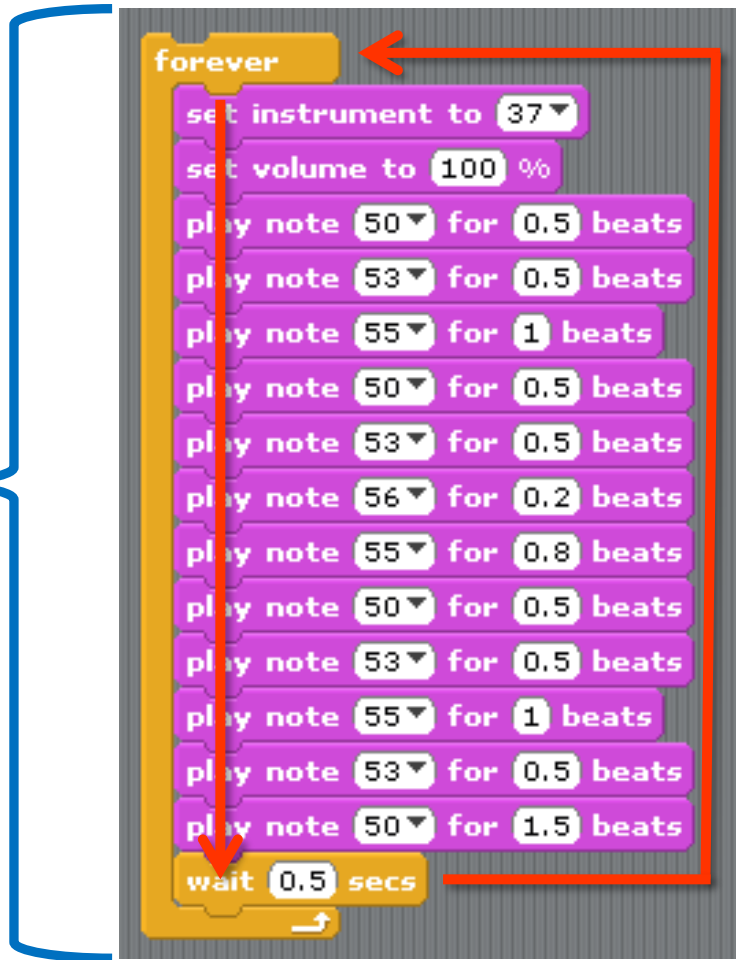
| | |
|--------|-------------|
| Motion | Events |
| Looks | Control |
| Sound | Sensing |
| Pen | Operators |
| Data | More Blocks |

```
forever
  move 10 steps
  if x position > 240
    say x position for 2 secs
```

```
forever
  set instrument to 37
  set volume to 100 %
  play note 50 for 0.5 beats
  play note 53 for 0.5 beats
  play note 55 for 1 beats
  play note 50 for 0.5 beats
  play note 53 for 0.5 beats
  play note 56 for 0.2 beats
  play note 55 for 0.8 beats
  play note 50 for 0.5 beats
  play note 53 for 0.5 beats
  play note 55 for 1 beats
  play note 53 for 0.5 beats
  play note 50 for 1.5 beats
  wait 0.5 secs
```

Ορολογία και Μοντέλο Εκτέλεσης

Loop - Βρόχος



A Scratch code block titled "forever" (Loop - Βρόχος) containing the following instructions: "set instrument to 37", "set volume to 100 %", "play note 50 for 0.5 beats", "play note 53 for 0.5 beats", "play note 55 for 1 beats", "play note 50 for 0.5 beats", "play note 53 for 0.5 beats", "play note 56 for 0.2 beats", "play note 55 for 0.8 beats", "play note 50 for 0.5 beats", "play note 53 for 0.5 beats", "play note 55 for 1 beats", "play note 53 for 0.5 beats", "play note 50 for 1.5 beats", and "wait 0.5 secs". A red arrow on the right side of the code block indicates the loop's path, starting from the top right of the "forever" block, going down through the instructions, and then looping back to the top right of the "forever" block.

Loop body – Σώμα βρόχου

Το Scratch «εκτελεί» το loop body ξανά και ξανά...



Γενικές Αρχές

- Υπάρχουν τρεις τύποι επανάληψης (ή **looping**):
 - **Forever** – Για πάντα (ατέρμονο)
 - **Counting** – Με μέτρηση
 - **Conditional** – Υπό συνθήκες

Forever loop

- Ένα **forever loop** επαναλαμβάνει το σώμα του βρόχου για πάντα.



Πως μπορεί το loop να σταματήσει;

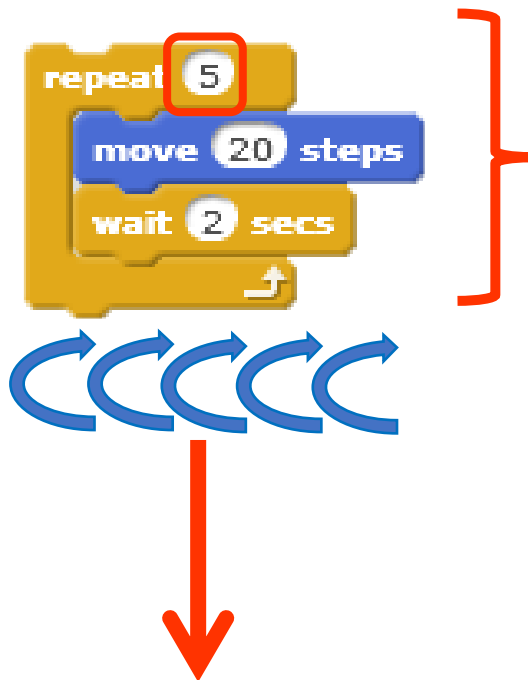
Δε μπορεί, εκτός

- εάν το σταματήσει ο χρήστης,
- ή υπάρχει ένα σταμάτημα εντός του loop,
- ή κλείσει ο υπολογιστής



Counting loop

- Ένα **counting loop** επαναλαμβάνει το σώμα του βρόχου **για συγκεκριμένες επαναλήψεις**.
 - 1, 10 ή 88 φορές...



Επανάληψη του σώματος **5** ακριβώς φορές.

Ένα μικρό demo...



- Counting loops...
 - Μετακίνηση ακριβώς 100 steps
 - Στροφή 90 μοιρών
 - “Flash” π.χ. γρήγορο μεγάλωμα και συρρίκνωση 3 φορές

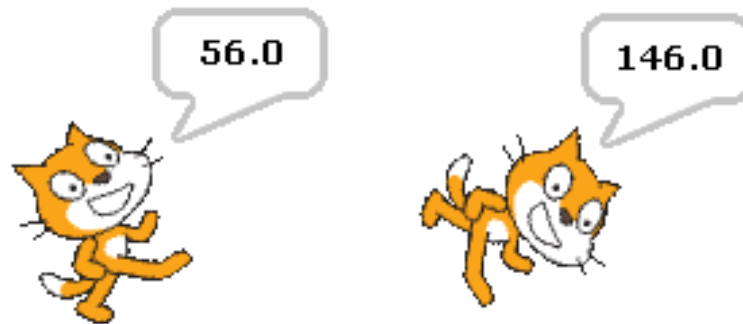
Ένα μικρό demo... στην πράξη

lecture1_06.sb3

```
say x position for 2 secs  
repeat 5  
  move 20 steps  
  wait 0.5 secs  
say x position for 2 secs
```



```
say direction for 2 secs  
repeat 90  
  turn 1 degrees  
say direction for 2 secs
```



```
repeat 3  
  repeat 10  
    change size by 10  
  repeat 10  
    change size by -10
```



Doubly nested loop

Παρατήρηση!

- Παρατηρείστε ότι ένα **forever** **δεν μπορεί** να ακολουθηθεί από άλλη εντολή,
- ενώ ένα **repeat** επιτρέπει επόμενες εντολές.



Conditional Loop

- Ένα **conditional loop** επαναλαμβάνει το σώμα του βρόχου με βάση μία **συνθήκη**.
 - Αν η συνθήκη είναι **αληθής-true** ο βρόχος σταματάει.
 - Αν η συνθήκη είναι **ψευδής-false** ο βρόχος επαναλαμβάνεται.

Η συνθήκη είναι αληθής, σταματάει το loop



Ένα Conditional Loop επαναλαμβάνεται 0 ή περισσότερες φορές.

Ένα μικρό demo...

- **Conditional Loops...**

- Κινήσου μέχρι να φτάσεις στην άκρη της σκηνής.
- Στρίψε μέχρι το sprite να δείξει 90 μοίρες προς τα πάνω.
- Ανίχνευσε το ποντίκι και ζωγράφισε στη σκηνή όσο το κουμπί του ποντικιού είναι πατημένο...

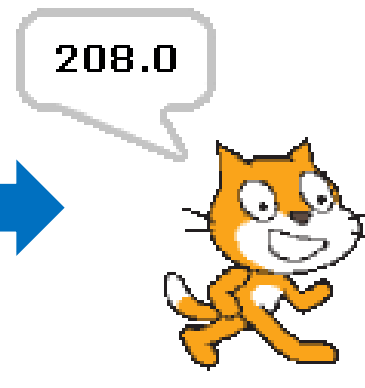


Ένα μικρό demo... στην πράξη (1/3)

DEMO

- Conditional Loops...
 - Κινήσου μέχρι να φτάσεις στην άκρη της σκηνής.

lecture1_07.sb3



Ένα μικρό demo... στην πράξη (2/3)

DEMO

- Conditional Loops...
 - Στρίψε μέχρι το sprite να δείξει 90 μοίρες προς τα πάνω.

```
repeat until direction = 0  
  turn 1 degrees
```

```
repeat until direction = 0  
  if direction > 0 then  
    turn 1 degrees  
  else  
    turn 1 degrees
```

lecture1_07.sb3



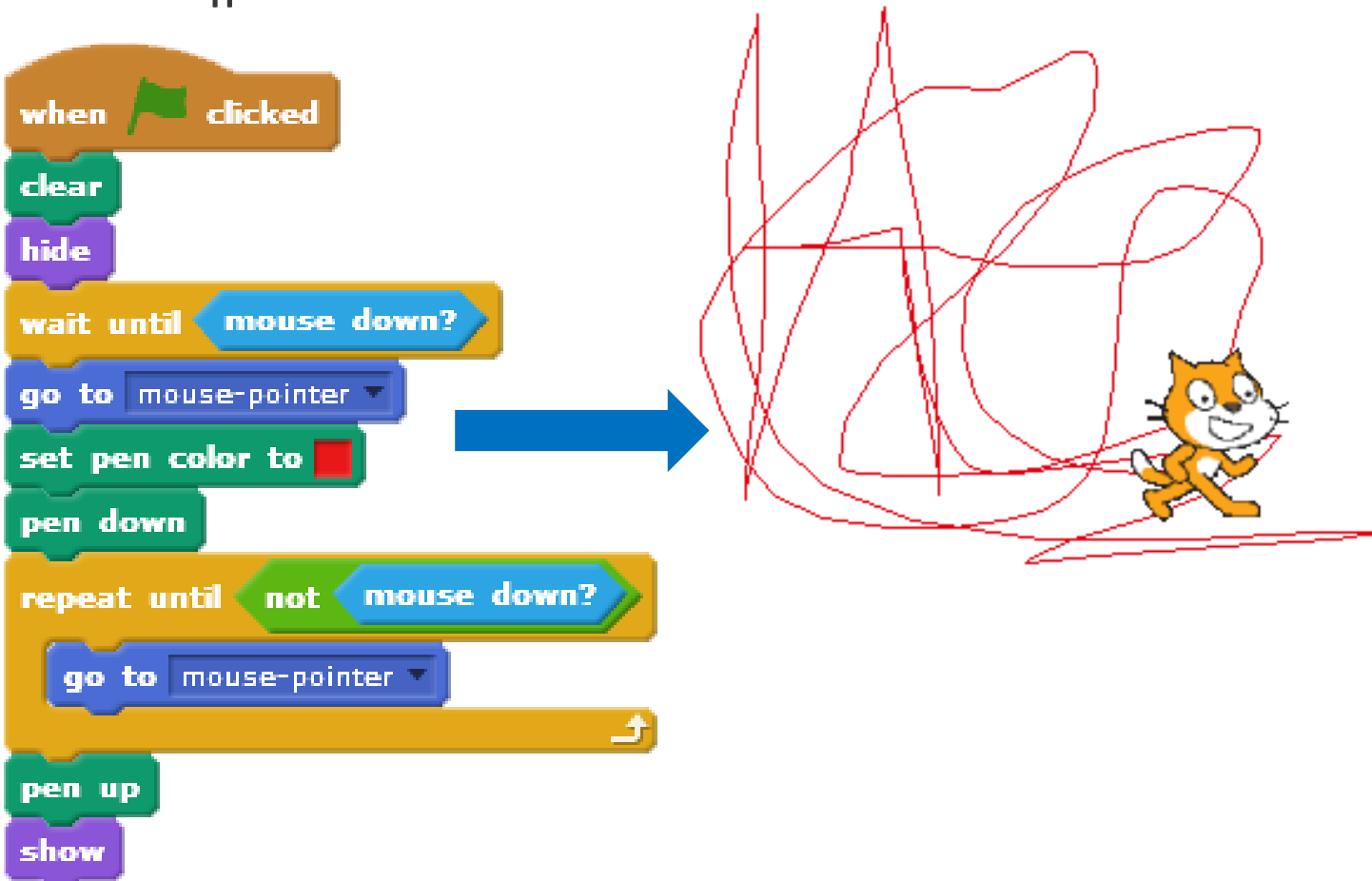
Ένα μικρό demo... στην πράξη (3/3)

DEMO

- Conditional Loops...

- Ανίχνευσε το ποντίκι και ζωγράφισε στη σκηνή όσο το κουμπί του ποντικιού είναι πατημένο...

lecture1_07.sb3



The image shows a Scratch script on the left and its visual output on the right. The script starts with a 'when clicked' event, followed by 'clear', 'hide', and 'wait until mouse down?'. Then, it goes to the mouse pointer, sets the pen color to red, and turns the pen down. A 'repeat until not mouse down?' loop contains a 'go to mouse pointer' block. After the loop, it turns the pen up and shows the cat. The visual output shows a red scribble drawn by the cat character.

```
when clicked
clear
hide
wait until mouse down?
go to mouse-pointer
set pen color to red
pen down
repeat until not mouse down?
  go to mouse-pointer
pen up
show
```

Τι είδαμε μέχρι τώρα;

- Η επανάληψη (repetition) είναι η τρίτη από τις προγραμματιστικές δεξιότητες που θα πρέπει να καλλιεργήσετε:
 - Αλληλουχία των εντολών
 - Επιλογή ανάμεσα σε διάφορες εναλλακτικές
 - Επανάληψη
- Στις περισσότερες προγραμματιστικές γλώσσες η επανάληψη έχει μία λίγο διαφορετική μορφή:

- Τα counting loops ονομάζονται “for” loops
 - για (for) έναν αριθμό επαναλήψεων...



- Τα conditional loops συχνά ονομάζονται “while” loops
 - όσο (while) η συνθήκη είναι αληθής (true)...



Και λίγη δουλειά για το σπίτι...

- Και μία άσκηση:
 - Στην Αγγλία, όταν βρέχει καταρρακτωδώς, χρησιμοποιείται η ακόλουθη έκφραση για να το περιγράψει: `“it’s raining cats and dogs”`. Φτιάξτε ένα πρόγραμμα με 10 ή περισσότερα sprites (μία μείξη από γάτες και σκύλους) που πηγαίνουν από την κορυφή προς το κάτω μέρος της σκηνής, συνέχεια...
 - Το αποτέλεσμα θα πρέπει να φαίνεται όπως στην παρακάτω εικόνα 😊



Remember!
This is only the start!!!



forever
imagine
program
share

