Areadne
Lifelong Learning centre

Erasmus+
KA1 courses

Scratch for the Inclusive Classroom

Inclusive Classroom

Inclusive teaching and **learning** practices are instrumental in **creating** and **maintaining a learning environment** in which:

- **all participants** are **fully engaged** and **respected.**
- all participants are **open to ideas**, **perspectives**, and **ways of thinking** that are **distinct from their own**.

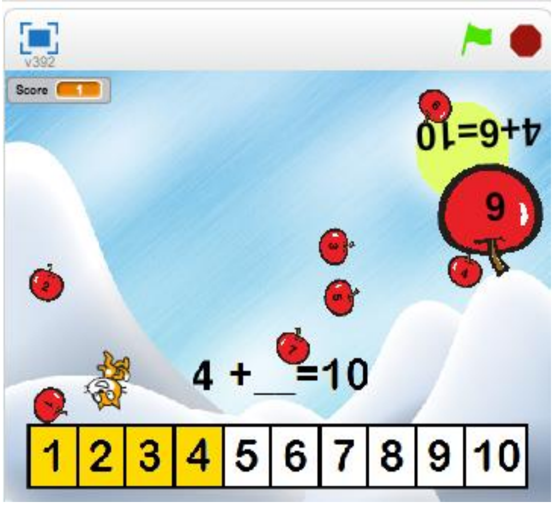
What is Scratch?

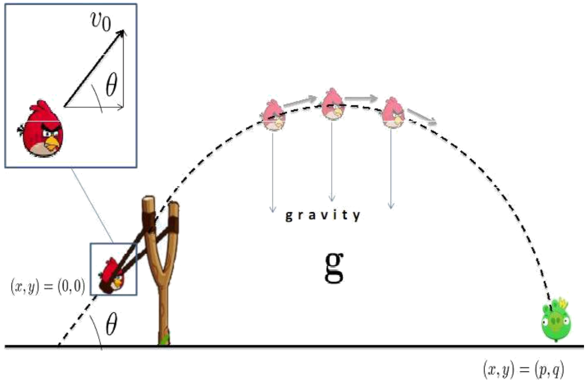
- Free **visual programming language** and **programming environment**.
- Project of the Lifelong Kindergarten Group at the **MIT Media Lab.**
- Designed **especially for ages 8 to 16**, but is used by people of **all ages**.
- An **engaging tool** for **children** of any learning style.
- Can be **easily personalized** by children with **their own media content**.

  They can add their own photos, recorded voices or sounds and graphics.
- Aims to promote **cooperation through online communities**.
- Scratch 3 is **available online** and as an **offline application** for :
  - Windows - MacOS

Scratch in Education

- Scratch enables teachers and pupils to make:
  - **Educational games**
  - **Simulations** of **science experiments**
  - **Animation stories** for a **school subject**
- Cultivates **pupils**' ability to:
  - **think creatively**
  - **reason systematically**
  - **work collaboratively**
- Encourages
  - **Sharing**, **reuse** and **combination** of **code**
  - **Communication** in a **global community**
  - There is also a **special community for education** (ScratchEd) with lesson plans and other relative information.

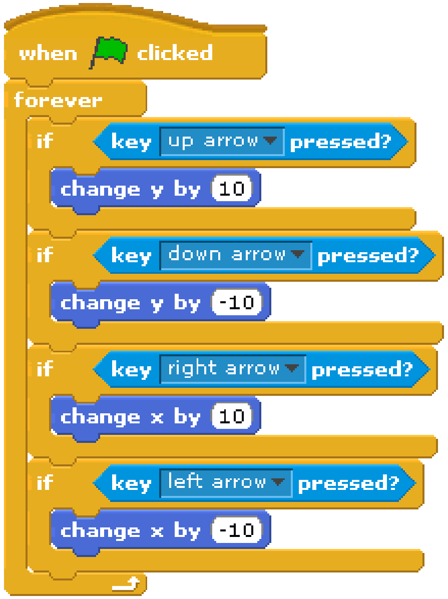




Scratch for the Inclusive Classroom

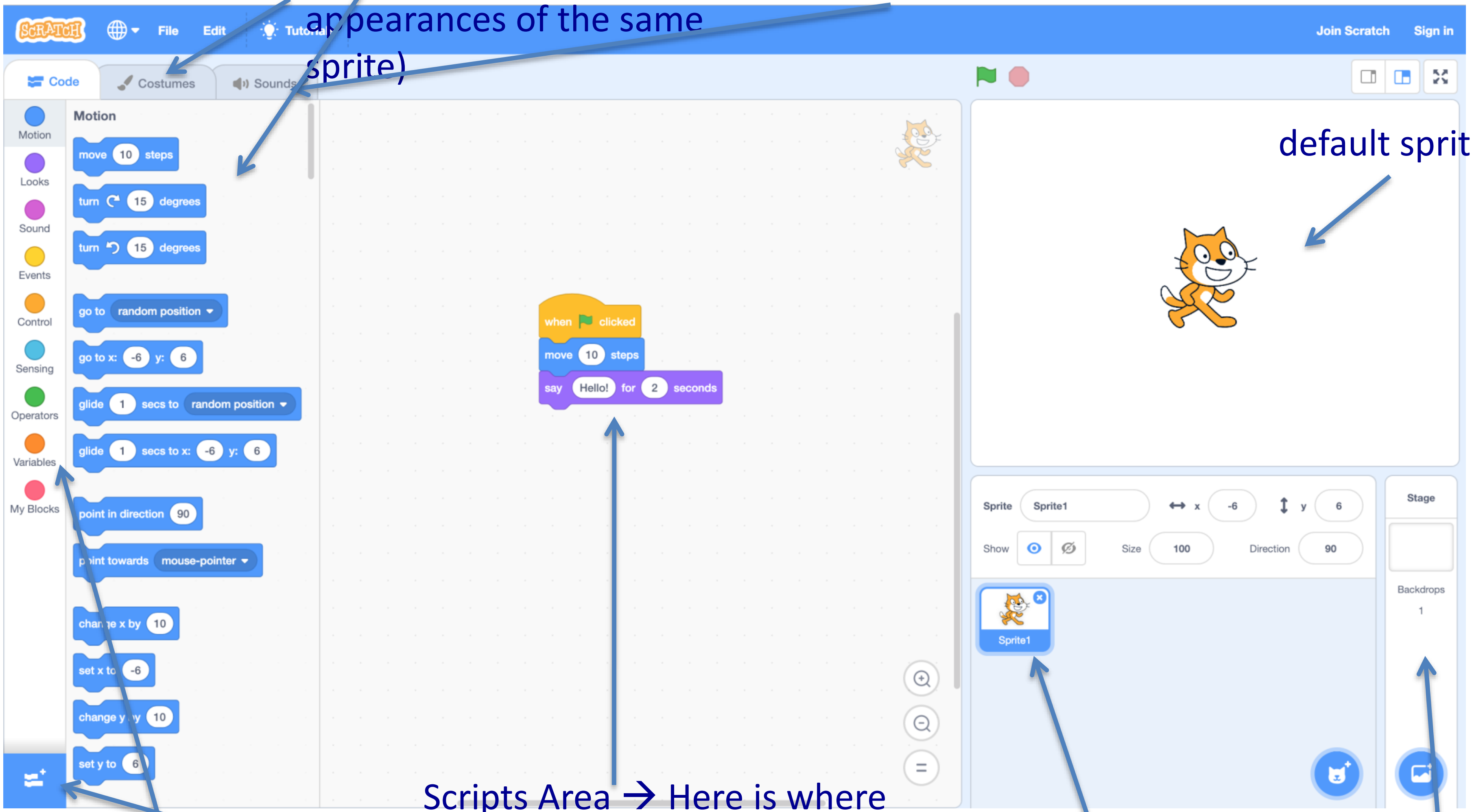
Scratch provides a friendly environment which:

- Helps **pupils** to:
  - **easily adapt** to their **learning space**.
  - get **immediate interaction** of their **actions** or **program changes.**
- Provides **teachers** with the ability to:
  - **easily program** engaging applications for their pupils
  - keep their classroom **motivated** by using
    - **Sounds**
    - **Animations**
    - **Images**
    - **Texts**



```
when 🏁 clicked
forever
    if <key [up arrow ▾] pressed?>
        change y by (10)
    if <key [down arrow ▾] pressed?>
        change y by (-10)
    if <key [right arrow ▾] pressed?>
        change x by (10)
    if <key [left arrow ▾] pressed?>
        change x by (-10)
```

Scratch Environment



```
when ⚑ clicked
move 10 steps
say Hello! for 2 seconds
```

Blocks of commands

Costumes (different appearances of the same sprite)

Sounds

default sprite

Scripts Area → Here is where command blocks are put

Groups of commands

More groups (Extensions)

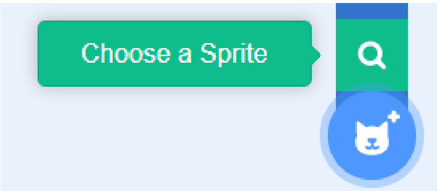
Active sprites

Stage (Background)

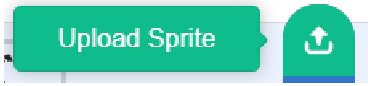
Sprite Selection

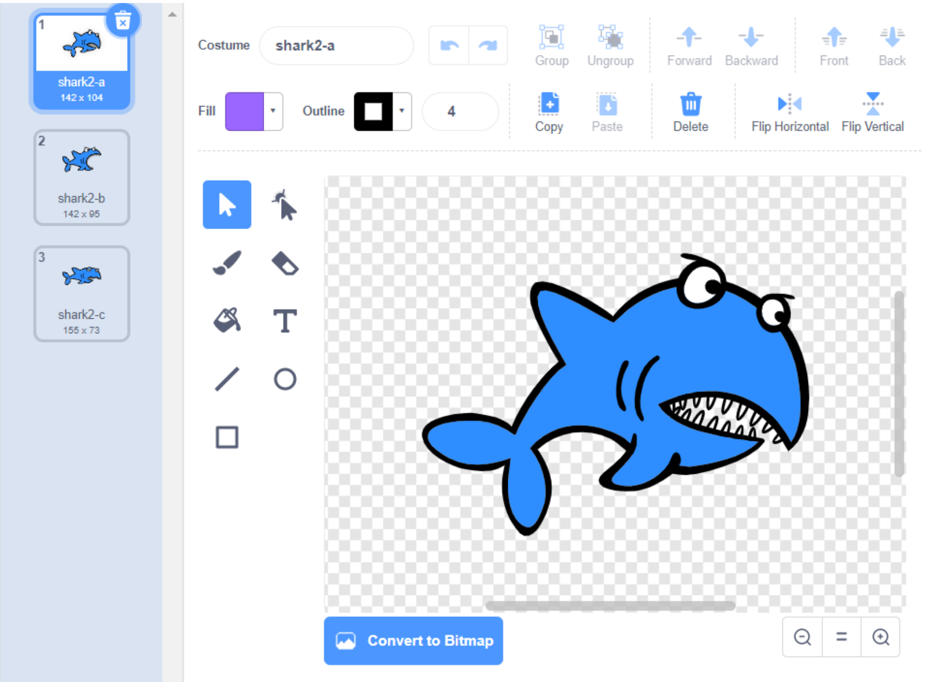
- A **sprite** is **a small graphic that can execute commands.**
- One or more sprites can be:
  - **Drawn** or

  - **Introduced** from:
    - **A Scratch library**



    - **A file**



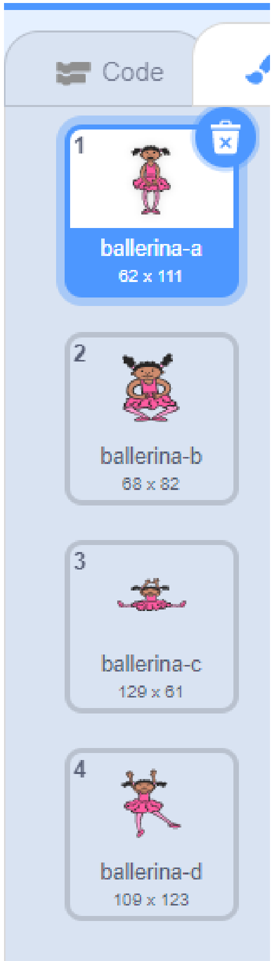


Sprite Selection

Sprite Costumes

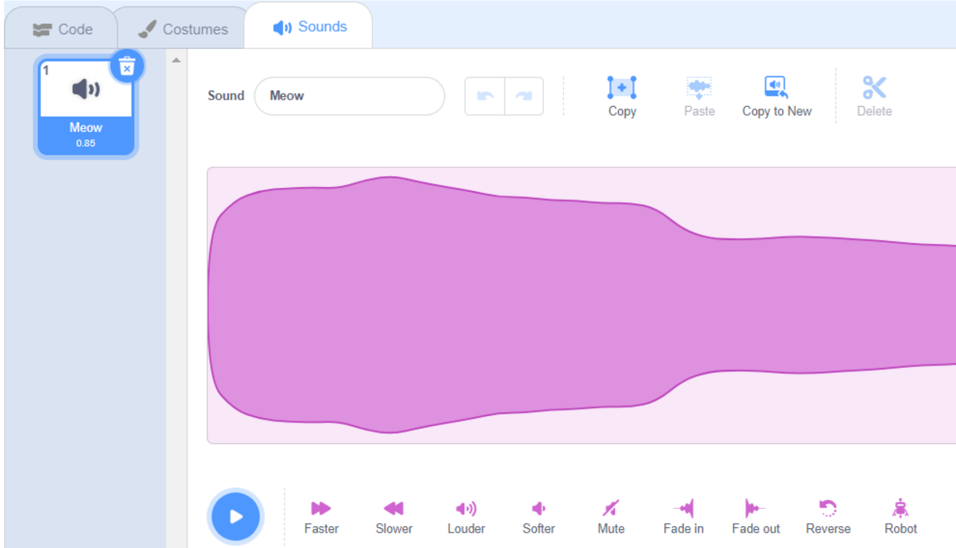
- A **sprite** has a set of different **costumes** (appearances).
- One or more **costumes** can be:
  - **Drawn**
  - **Introduced** from:
    - **A Scratch library**
    - **A file**

Every time you select a sprite, you have to check that its **costumes** have the form you need.



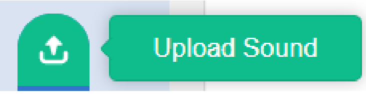
Sprite Selection
Sounds, Size

- A sprite may also have been **loaded** with a **set of sounds**.



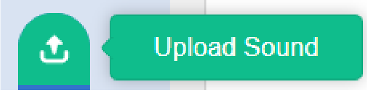
Sound: Meow

- A sound can be :
  - **Loaded** from a **library.**



  - **Recorded.**

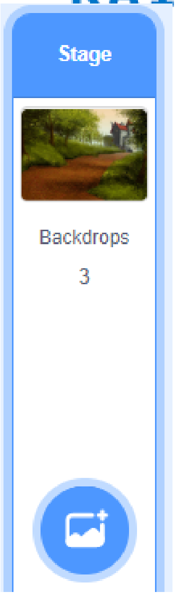
  - **Imported** from an **audio file.**

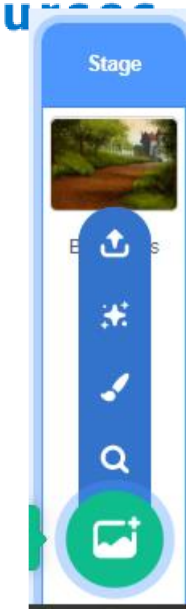


Check that you have loaded **all** the sounds that the **sprite** is **supposed to make**.

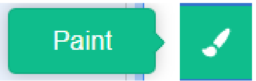
Setting the background

- Click on **Stage**
- Select the "**Backdrops**" tab
- Each **backdrop** is a **different appearance (look)** of the Stage

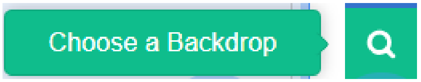




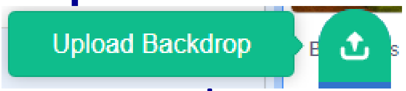
- A backdrop can be:
  - **Drawn**



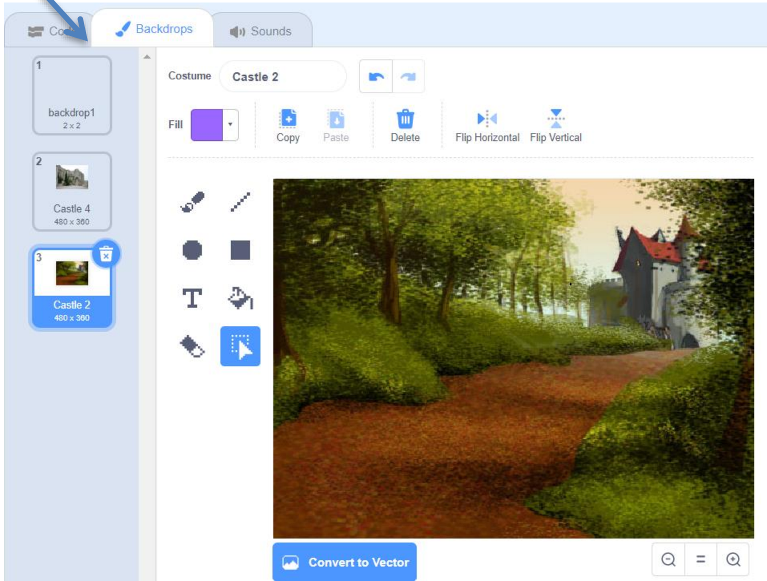
  - **Added** from **Scratch library**



  - **Imported** from **a file**

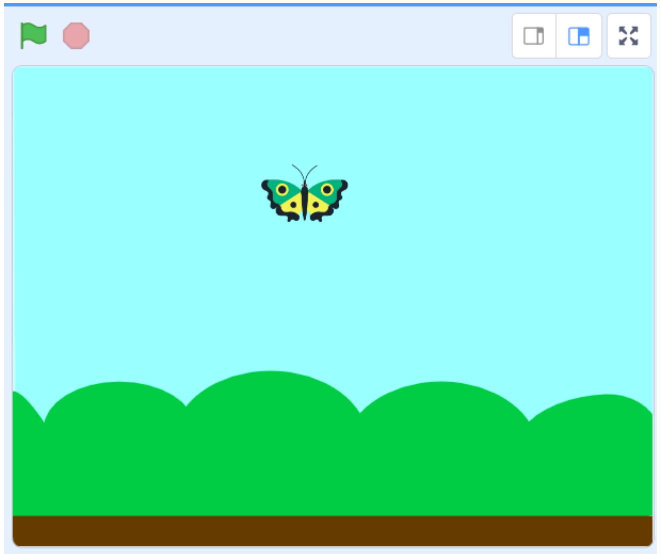


  - We can have **more than one backdrops,** if a change of the scene is necessary to be made in the application i.e. in a game.
- A backdrop can be **modified**.

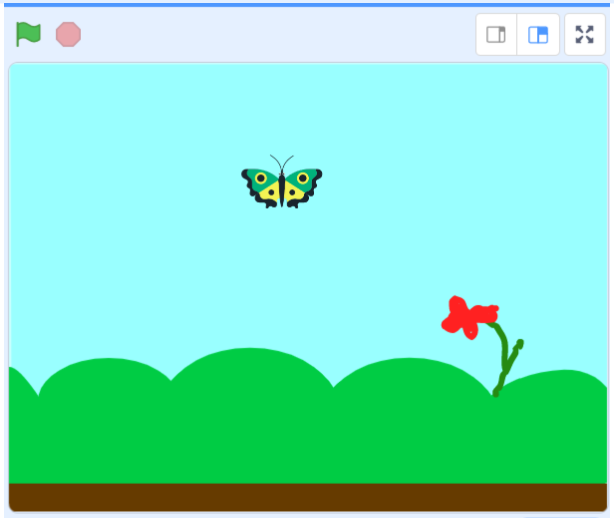


Exercise 1

- Choose the appropriate backdrop and sprite so that the screen of your application will look like this.

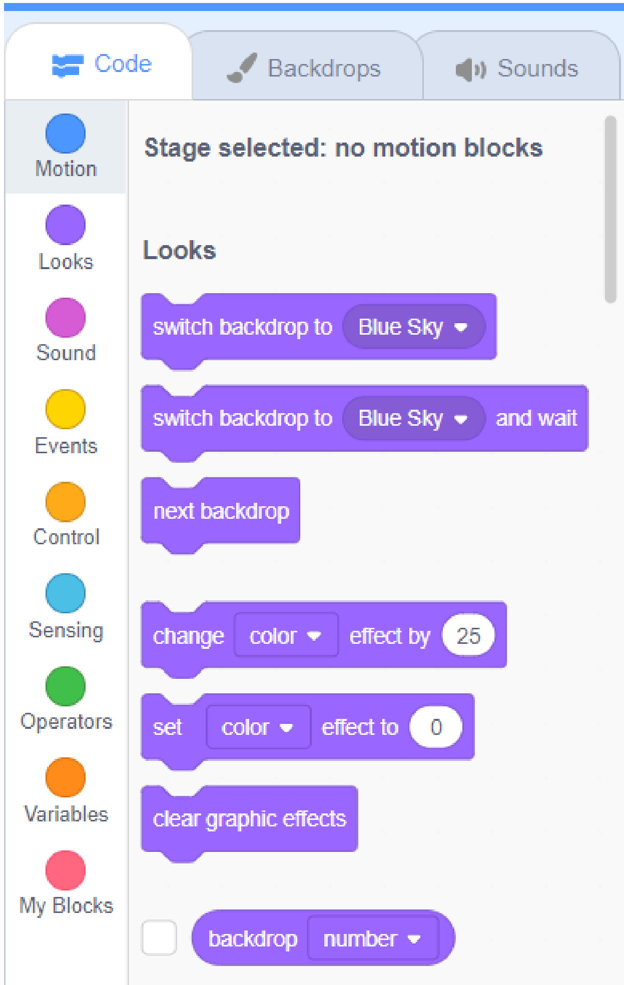


- How could we then modify the backdrop in order to look like this?



Command blocks

- Various **types of blocks** associated with:
  - **Motion**
  - **Looks**
  - **Sound**
  - **Events**
  - **Control**
  - **Sensing etc…**
- When a **sprite** is selected, **all** types of blocks are **available.**
- When **the stage** is selected, **neither motion nor pen blocks** are available.



Stage selected: no motion blocks

Looks

switch backdrop to Blue Sky ▾

switch backdrop to Blue Sky ▾ and wait

next backdrop

change color ▾ effect by 25

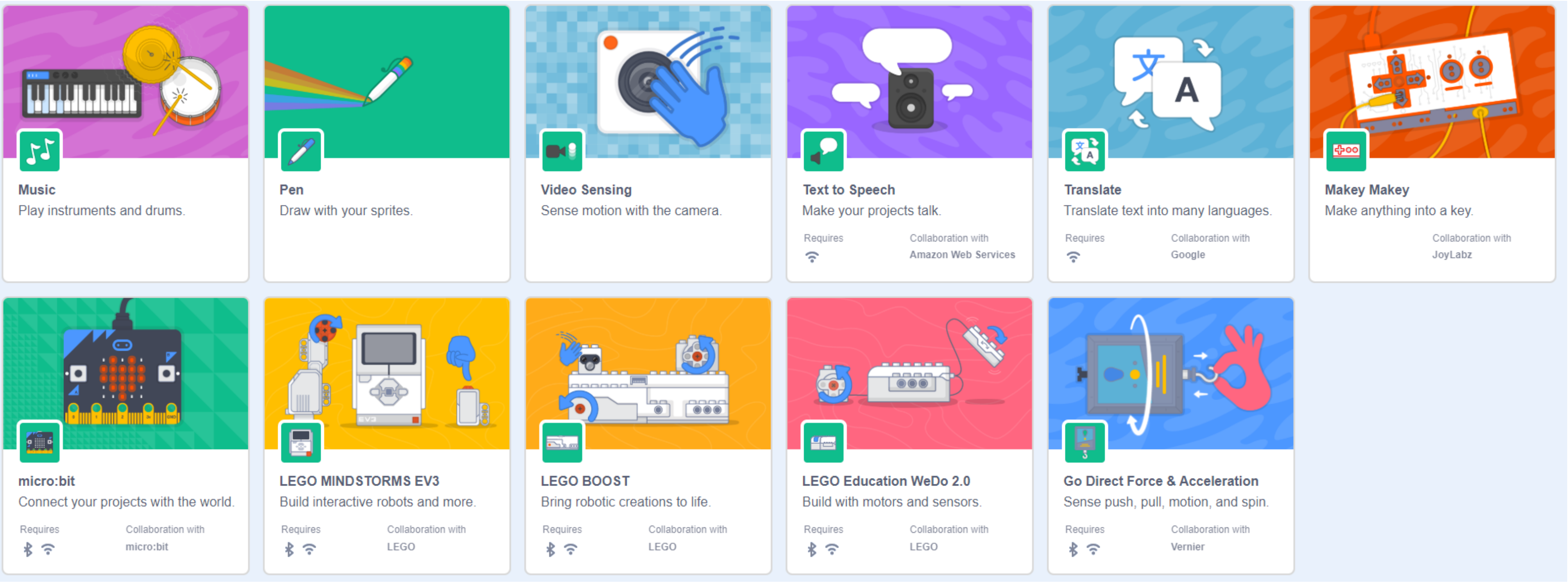
set color ▾ effect to 0

clear graphic effects

□ backdrop number ▾

Command blocks

You can **add more command blocks** through **extensions** button.



Music
Play instruments and drums.

Pen
Draw with your sprites.

Video Sensing
Sense motion with the camera.

Text to Speech
Make your projects talk.
Requires | Collaboration with Amazon Web Services

Translate
Translate text into many languages.
Requires | Collaboration with Google

Makey Makey
Make anything into a key.
Collaboration with JoyLabz

micro:bit
Connect your projects with the world.
Requires | Collaboration with micro:bit

LEGO MINDSTORMS EV3
Build interactive robots and more.
Requires | Collaboration with LEGO

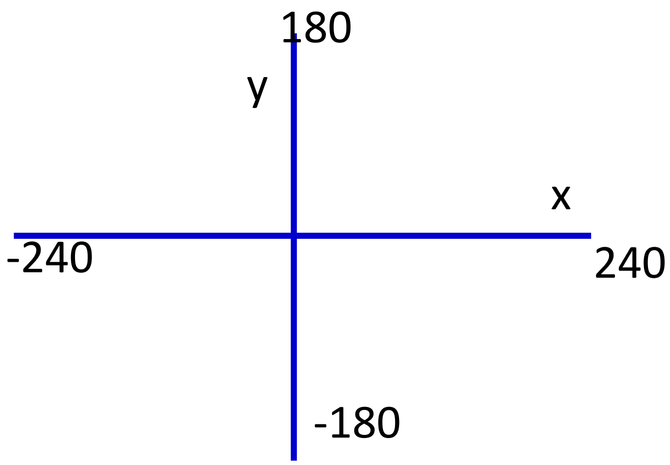
LEGO BOOST
Bring robotic creations to life.
Requires | Collaboration with LEGO

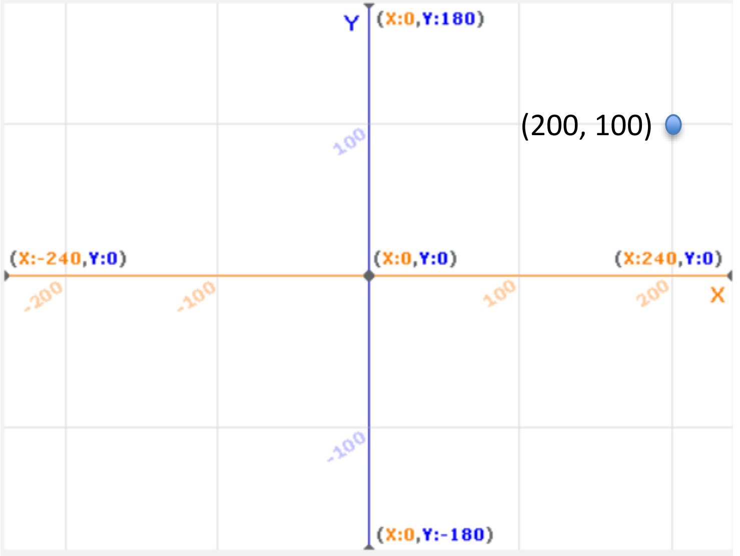
LEGO Education WeDo 2.0
Build with motors and sensors.
Requires | Collaboration with LEGO

Go Direct Force & Acceleration
Sense push, pull, motion, and spin.
Requires | Collaboration with Vernier

Scratch Stage coordinates

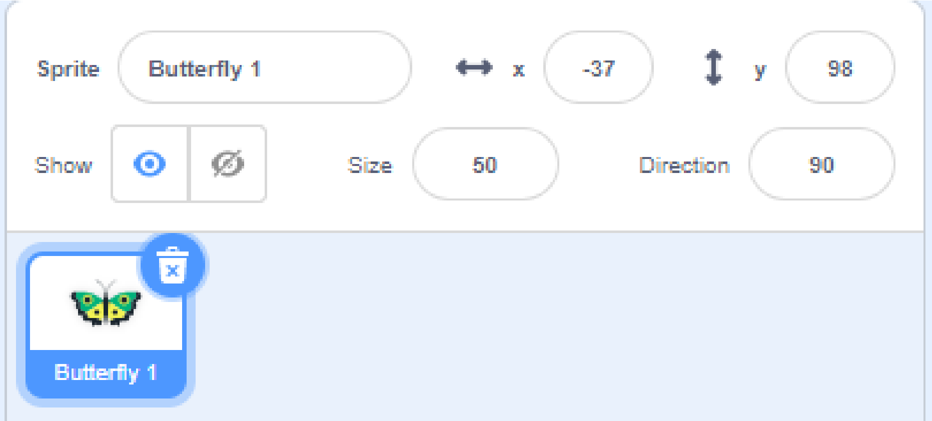
- The stage is 480 pixels wide and 360 pixels high.
- Every **point** is **defined** by its **X-Y coordinates**. **X coordinate** get values in the range [-240, 240] and **Y coordinate** gets values in the range [-180, 180].
- The **center** of the stage has **(0,0)** X-Y coordinates.
- When we refer to **a point a sprite** should **move to**, we actually mean that its **center** should reach that point.





Sprite Information

- The following information about the sprite is available:
  - The **x-y coordinates** of its **centre** .
  - Its **direction.**
  - Its **size.**
  - If it is meant to be **visible** when the application starts.

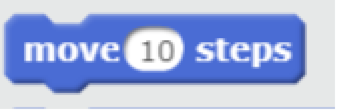


Sprite Butterfly 1 ↔ x -37 ↕ y 98

Show ● ∅ Size 50 Direction 90

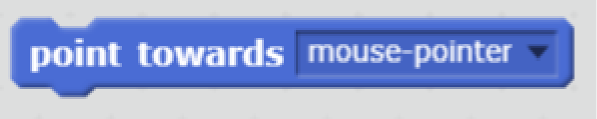
Motion blocks

- A **sprite** can:
  - **Move a number of steps** (pixels) in the specified direction



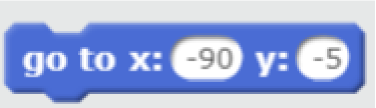
move 10 steps

  - **Point towards** the **mouse pointer** or **another sprite**.

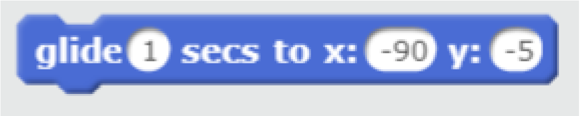


point towards mouse-pointer ▾

  - **Move immediately** to a point or **glide** to a point within a specified time.



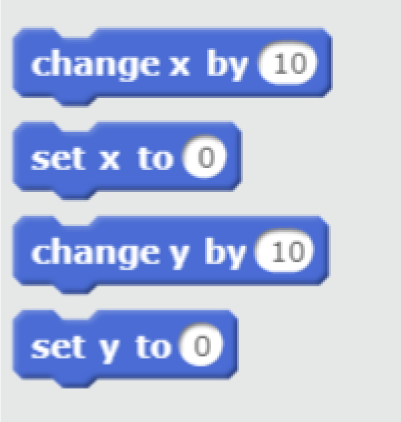
go to x: -90 y: -5



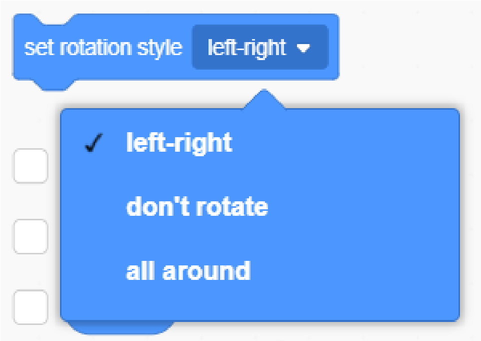
glide 1 secs to x: -90 y: -5

Motion blocks

- A sprite can:
  - **Change** its **coordinates**.

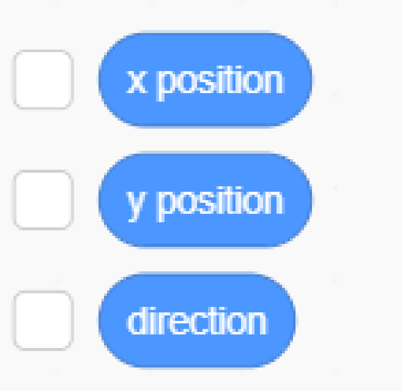


change x by 10
set x to 0
change y by 10
set y to 0



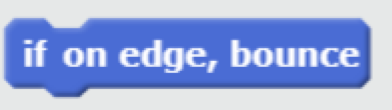
set rotation style left-right ▾

✓ left-right
don't rotate
all around



□ x position
□ y position
□ direction

  - **Bounce on stage edges** (mind the **rotation style**!)



if on edge, bounce

- We can **see** how **direction** and the **coordinates** of a sprite **change** while it **moves** by **ticking the appropriate boxes**.

Exercise 2

Extend Exercise 1 as follows:

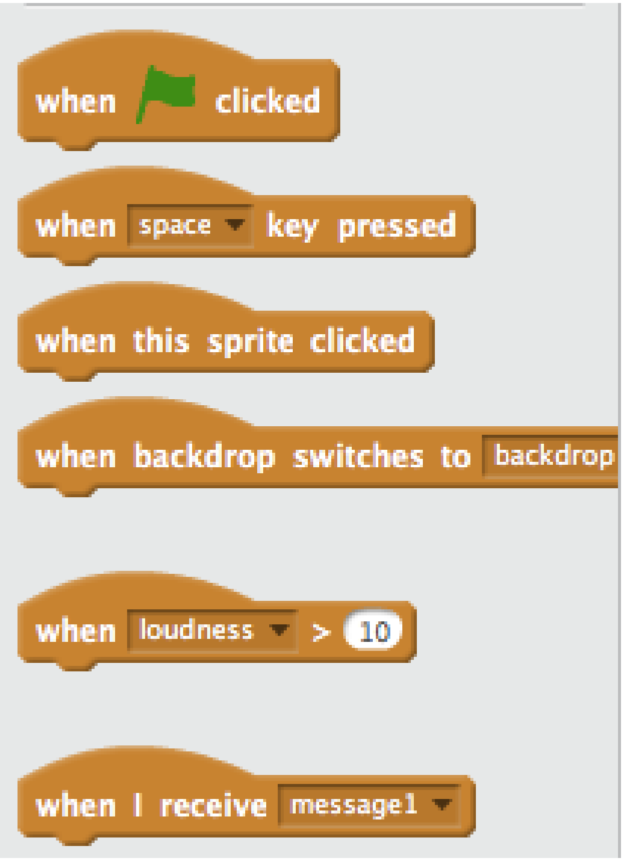
Provide the appropriate blocks of commands so that the butterfly

1. Points in direction of 90^{o} (right)
2. Goes initially to point (0,0)
3. Glides to the flower in 2 seconds

Hint: To find the x-y coordinates of the flower, position the sprite (the butterfly) over the flower. From the sprite information, write down the x-y coordinates.

Event Blocks

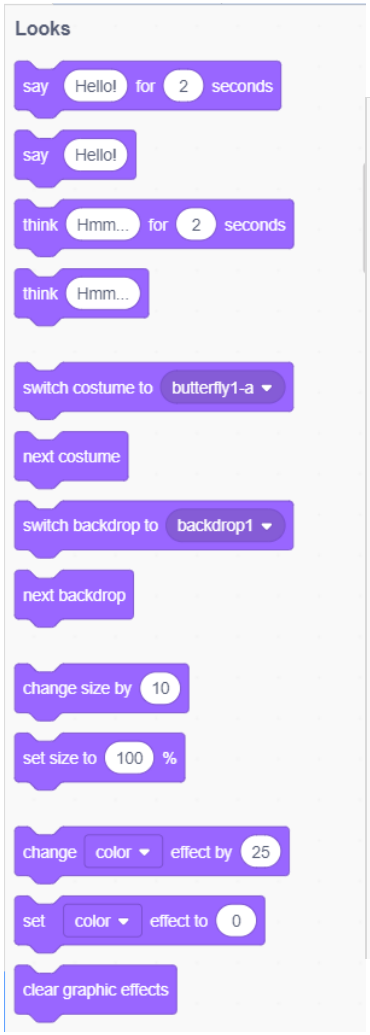
- Event blocks are activated when an event happens. They are followed by a set of blocks, which will be executed. Event blocks are activated:
  - **When the green flag has been clicked.**
  - **When** one **key is pressed.**
  - **When the user clicks on a sprite.**
  - **When the backdrop changes** to the one specified.

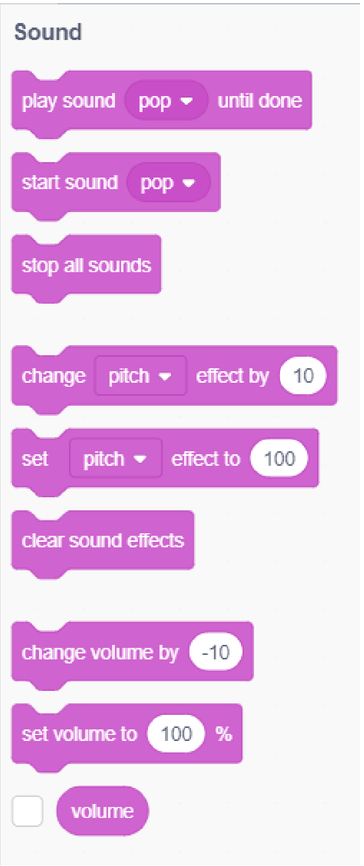


```
when 🚩 clicked
when space ▾ key pressed
when this sprite clicked
when backdrop switches to backdrop
when loudness ▾ > 10
when I receive message1 ▾
```

Looks and Sound

- A Sprite can:
  - **speak** or **think** as in a comic book
  - **Disappear** and **appear**
  - **Change color**, **size** or **costume**
  - **Change** the **backdrop** of the Stage.
- The Stage can:
  - **Change backdrop** or **color**
- Sprites and Stage can:
  - Produce the **sounds of instruments.**
  - **Play** an **imported** or **recorded sound.**
  - **Change volume** and **tempo** of the **sound**.

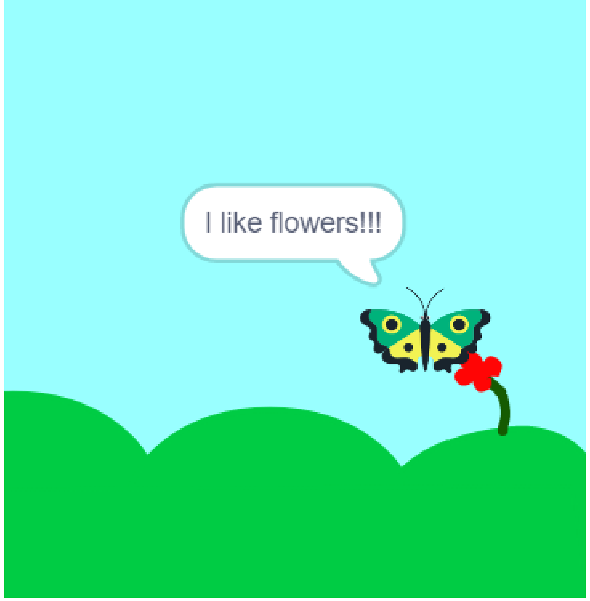




Exercise 3

Extend exercise 2 as follows:

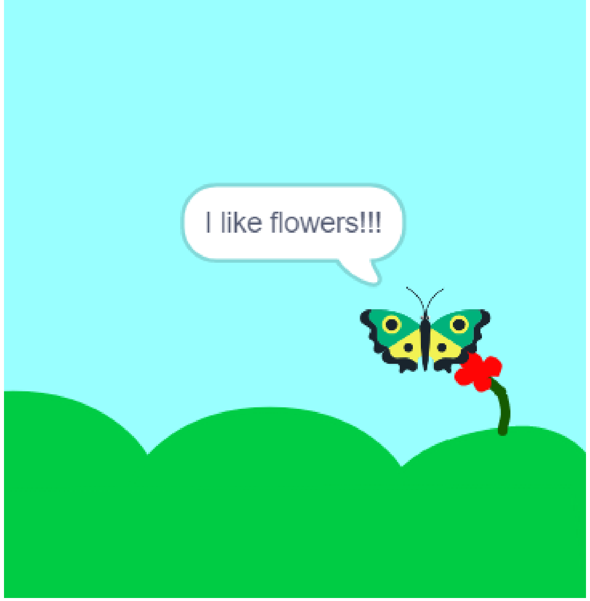
- When the butterfly is at the center, it should display the message “I am a butterfly!”
- When the butterfly reaches the flower it should display the message “I like flowers!!!”.



Exercise 4

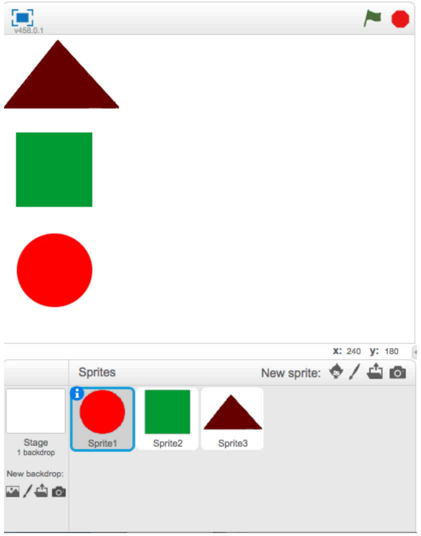
Extend exercise 2 as follows:

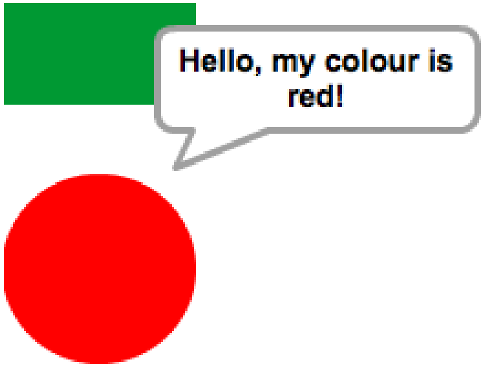
- Create and save a sound file with your own voice that says "I am a butterfly". The audio file will be played when the butterfly is at the center of the stage.
- Create and save a sound file with your own voice that says "I like flowers!!!". The audio file will be played when the butterfly reaches the flower.



Exercise 5

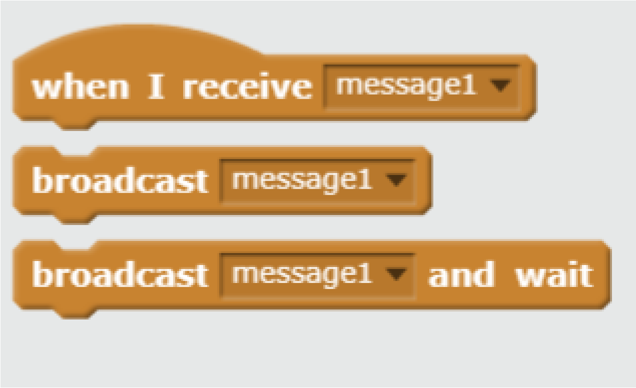
- Draw 3 sprites:
  - A red circle
  - A green square
  - A brown triangle
- The sprites will be initially placed on the left side of the screen.
- When someone clicks on a sprite:
  - A "cloud" will appear saying its shape and its color, as in a comic book text.
  - A recorded voice will repeat the above.
  - The sprite will glide in a second to the right side of the screen, and then return back to its initial position.





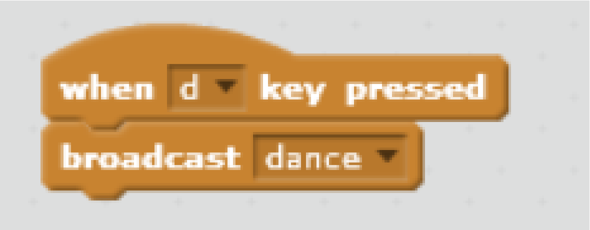
Communication between Sprites

- Sprites can communicate between them and also with the Stage through **messages**.



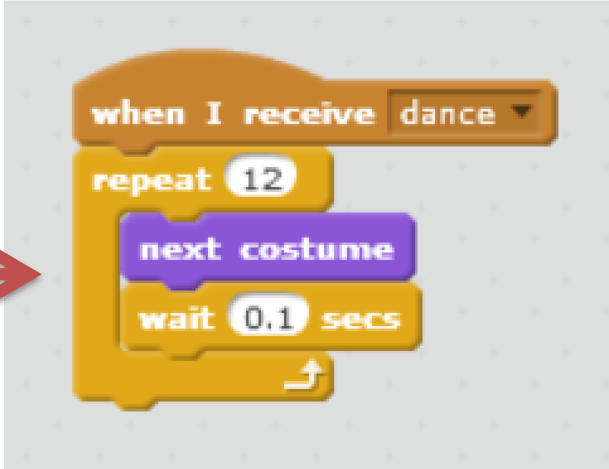
```
when I receive [message1 ▾]
broadcast [message1 ▾]
broadcast [message1 ▾] and wait
```

- A sprite or the stage **broadcasts** a **specific message.** In example, a sprite broadcasts the message ***dance***.



```
when [d ▾] key pressed
broadcast [dance ▾]
```

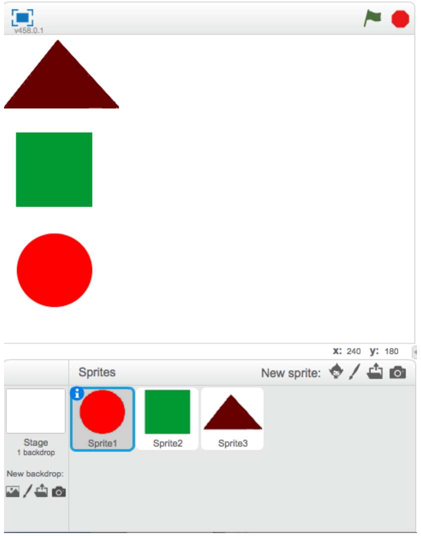
- The sprites that must react to this message will have a **When I receive *dance*** event block, with more code to be executed underneath.

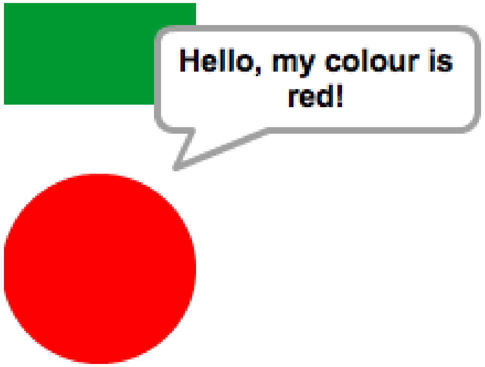


```
when I receive [dance ▾]
repeat (12)
    next costume
    wait (0.1) secs
```

Exercise 5

- Add the concert backdrop
- Add the balarina sprite.
- Resize the cat to 50% and the balarina to 80%.
- Write code so that when the green flag is pressed the cat will go to position x: -144, y=- 86 and the balarina to the center of the stage.
- Write code so that when the space key is pressed the balarina will broadcast the message “startDance” and it will start dancing by changing its custom 12 times.
- Write code so when the cat receives the message “startDance” it will start walking by walking





Remaining Command Blocks

- We did not cover any blocks that are associated with:
  - **Control functions** (except the ones that we have already examined)
  - **Sensing**
  - **Operators** (numerical, logical or comparison)
  - The programming of **custom blocks**
  - **Variables**
  - and **all the extensions.**

Remaining Command Blocks

- These commands though can be also very useful. Par example, you will have to use **sensing blocks** and **operators** and **variables** to create a quiz that keeps score of successful answers.

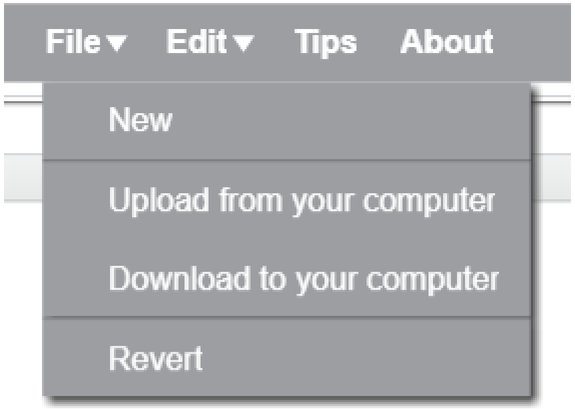
Save and load a project

- To save locally at your computer a project select:

  File→ Download to your computer

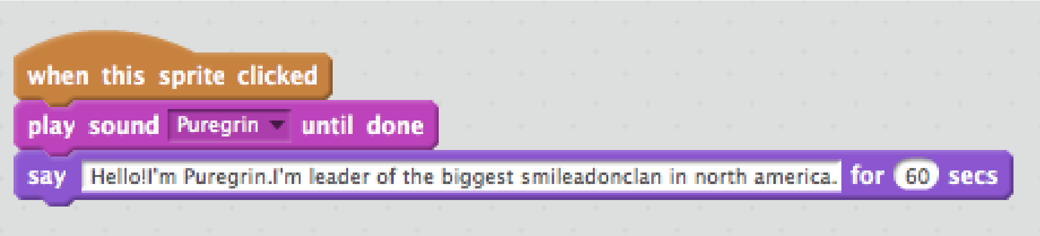
- To load a project from your computer select:

  File→ Upload from your computer

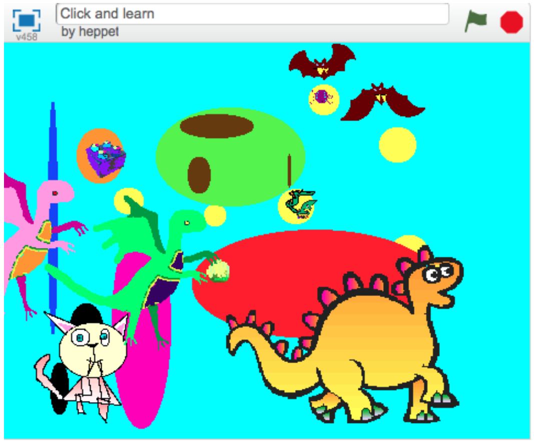


Scratch Applications Examples

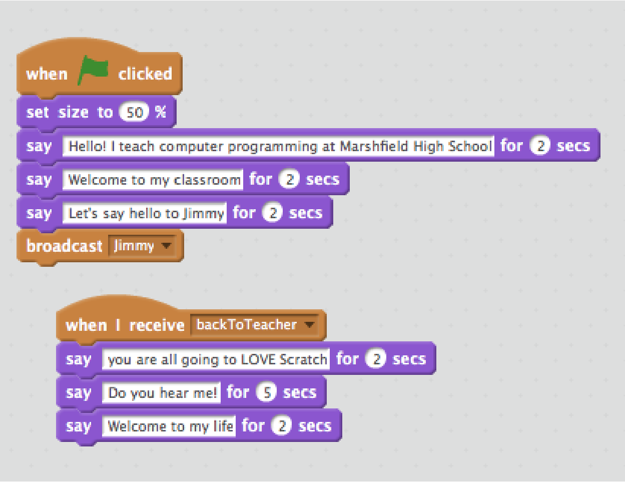
- Click and Learn



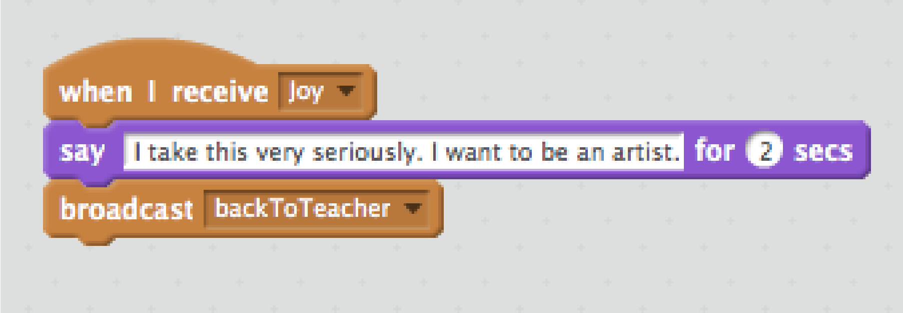
```
when this sprite clicked
play sound Puregrin until done
say Hello!I'm Puregrin.I'm leader of the biggest smileadonclan in north america. for 60 secs
```



- What I do



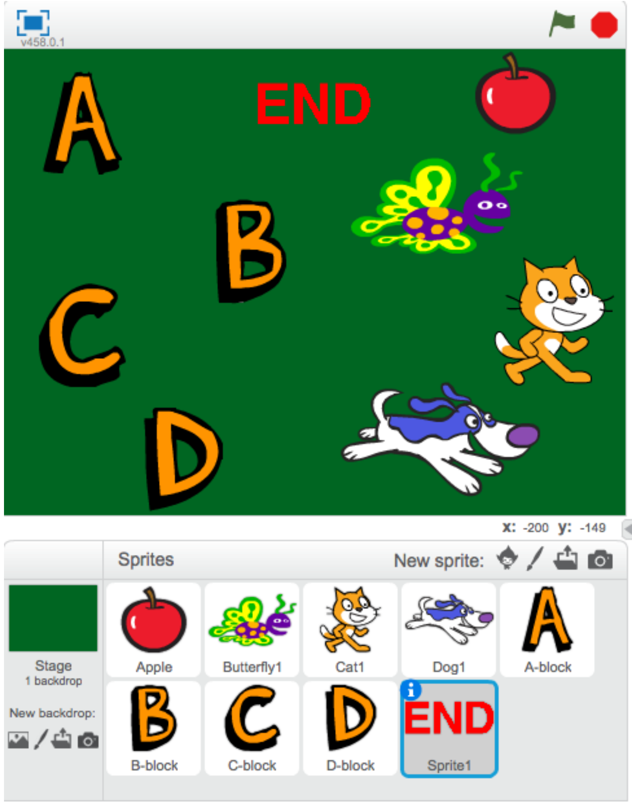
```
when [flag] clicked
set size to 50 %
say Hello! I teach computer programming at Marshfield High School for 2 secs
say Welcome to my classroom for 2 secs
say Let's say hello to Jimmy for 2 secs
broadcast Jimmy

when I receive backToTeacher
say you are all going to LOVE Scratch for 2 secs
say Do you hear me! for 5 secs
say Welcome to my life for 2 secs
```



```
when I receive Joy
say I take this very seriously. I want to be an artist. for 2 secs
broadcast backToTeacher
```

A Little Project for the inclusive classroom…

- You will need the following sprites:
  - The letters A, B, C, D, an apple, a butterfly, a cat, a dog and an "END" button (when the application starts the visible sprites will be the letters and the "END" button).
  - You can place them on any part of the screen by choosing a specific point and including the command "go to x … y … "
- When letter A is clicked
  1. A recorded sound "apple" will be played.
  2. The letter will say "apple" (in a cloud) for 3 seconds.
  3. An apple will appear for 3 seconds and then disappear.
- A similar process will be followed when the other letters are clicked.
- When "END" button is clicked, the sound "End of the game" will be played.



Areadne
Lifelong Learning centre

Erasmus+
KA1 courses

We hope this was a useful session

Please, try to apply your new experience on Scratch and provide your feedback!

Thank you!

You can **email any questions** about the material that we did not have the time to cover (and also about the material that we covered) at **loukas_dimopoulos@hotmail.com.**