

LEZIONE 10

“INTRODUZIONE A SCRATCH”

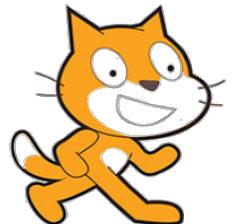
Laboratorio di Informatica per l’Educazione

A. A. 2014/2015

INTRODUZIONE

- *Scratch* è un nuovo linguaggio di programmazione visuale che ti permette di creare storie interattive, applicazioni educative, animazioni, giochi, etc.
- *Scratch* offre un ambiente di programmazione user-friendly che nella versione 2.0 è sia completamente on-line che in versione desktop (off-line)

SCRATCH

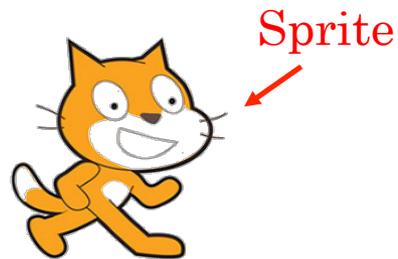


L' AMBIENTE SCRATCH

1. Un linguaggio di programmazione visuale permette di scrivere programmi collegando tra di loro dei blocchi grafici.
2. La metafora utilizzata da Scratch è quella della scrittura di un copione (***script***) per uno o più attori (***sprite***) che agiscono in una o più scene (***scene***) di una rappresentazione teatrale.
3. Uno *script* fornisce le istruzioni (organizzate in forma di programma-algoritmo) per uno *sprite*.
4. Un programma *Scratch* è costituito da più *script* che decidono il comportamento dei singoli *sprite* che, eventualmente, interagiscono tra loro (e.g., utilizzando messaggi o eventi).

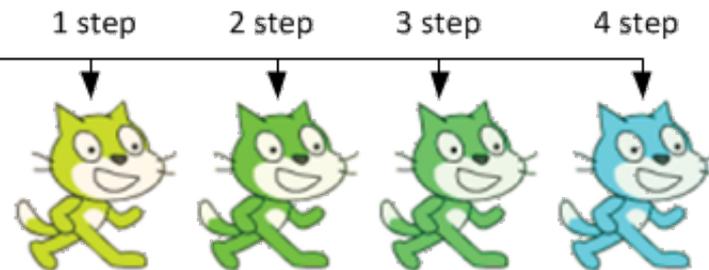
ESEMPIO:

- script che cambia colore quattro volte dello *sprite*

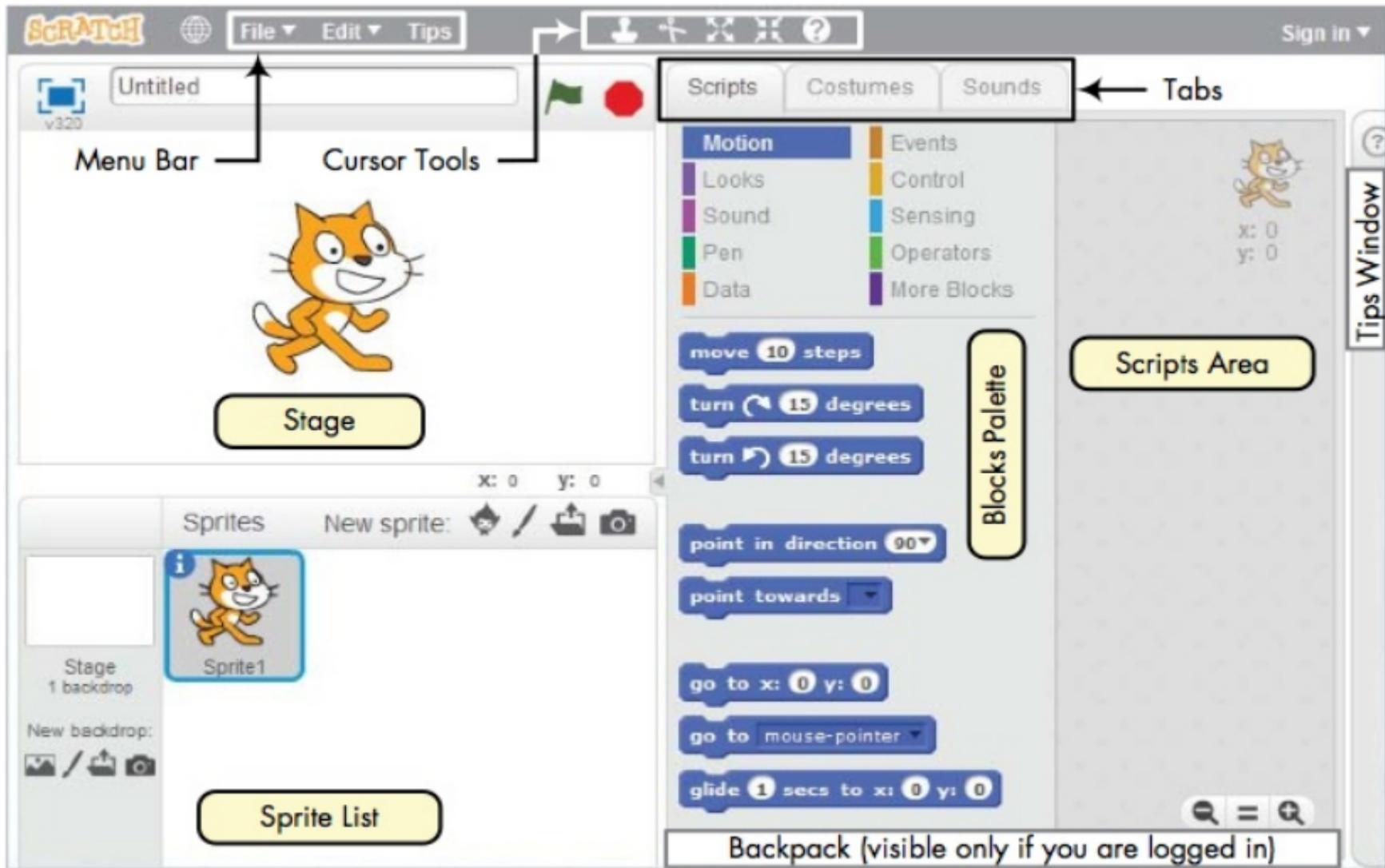


Script

```
rimuovi effetti grafici
ripeti 4 volte
  cambia effetto colore di 20
  attendi 1 secondi
```

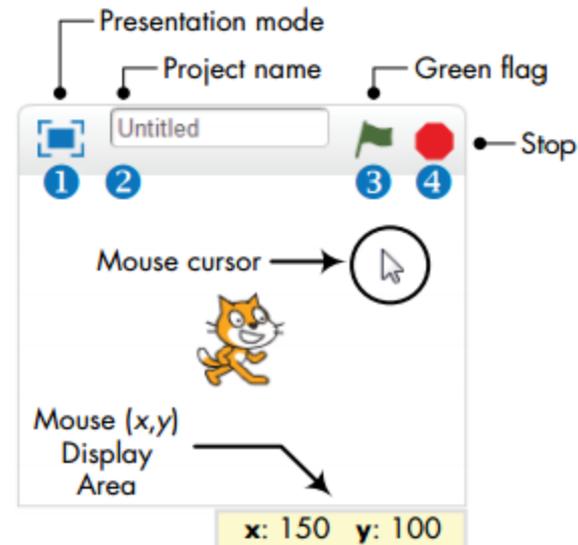
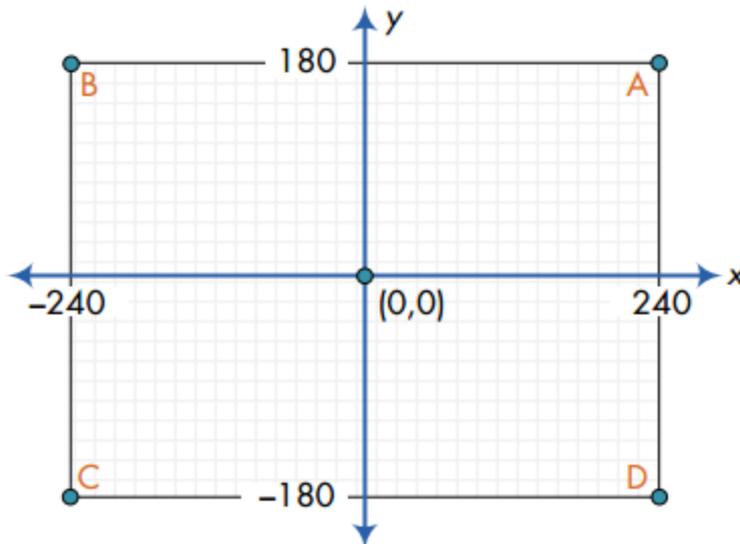


AMBIENTE DI PROGRAMMAZIONE SCRATCH



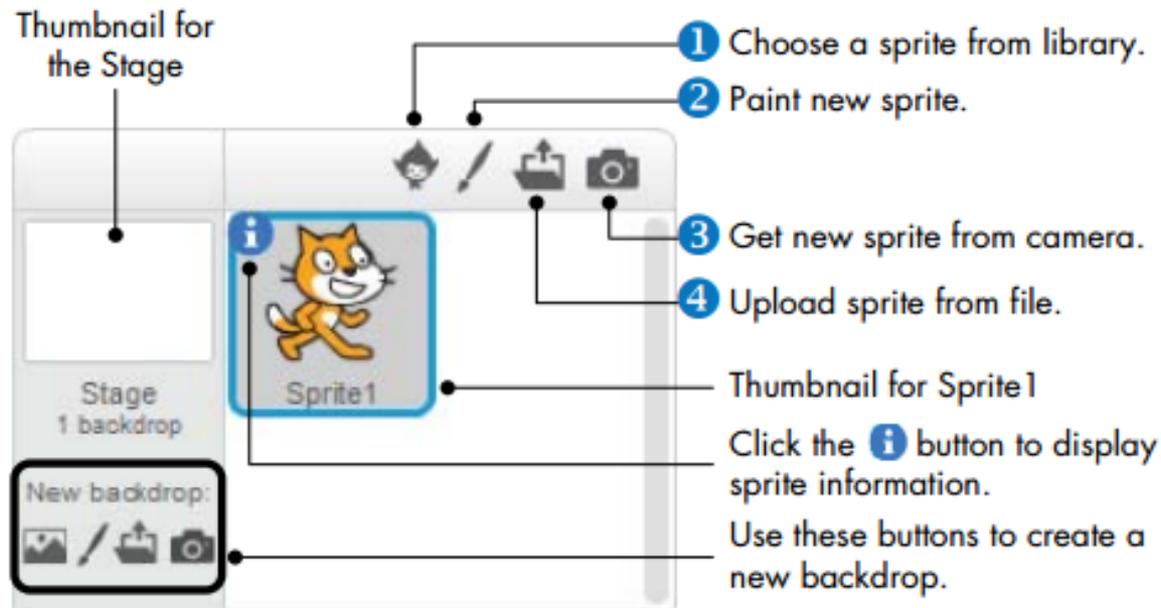
STAGE

- Lo Stage è lo schermo nel quale il risultato del nostro programma (col quale l'utente finale interagirà) sarà visualizzato.
- Ai diversi punti dello Stage si può accedere grazie alle coordinate cartesiane.
- Le dimensioni dello Stage sono di **480** passi in larghezza e di **260** in altezza (ricordiamoci che si misura in passi dato che l'obiettivo è animare un attore detto Sprite).



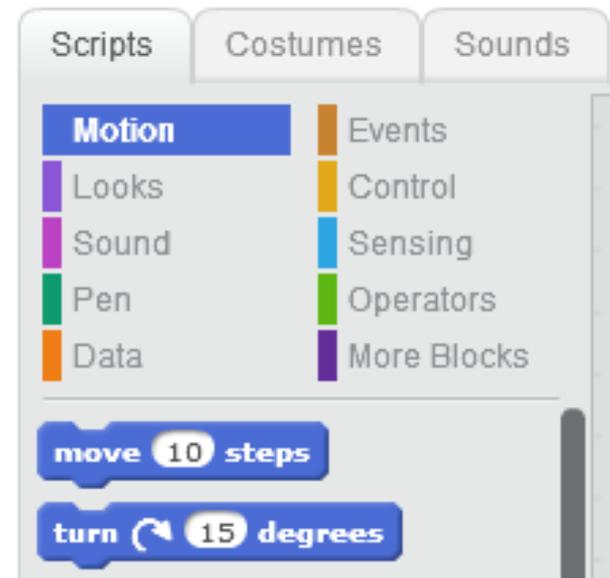
SPRITE

- Nella sezione Sprite List è possibile gestire uno o più *sprite* inclusi nel nostro programma.
- Cliccando sullo *sprite* è possibile visualizzare script, costumi e suoni associati negli *script tab*.



BLOCKS TAB: SCRIPT

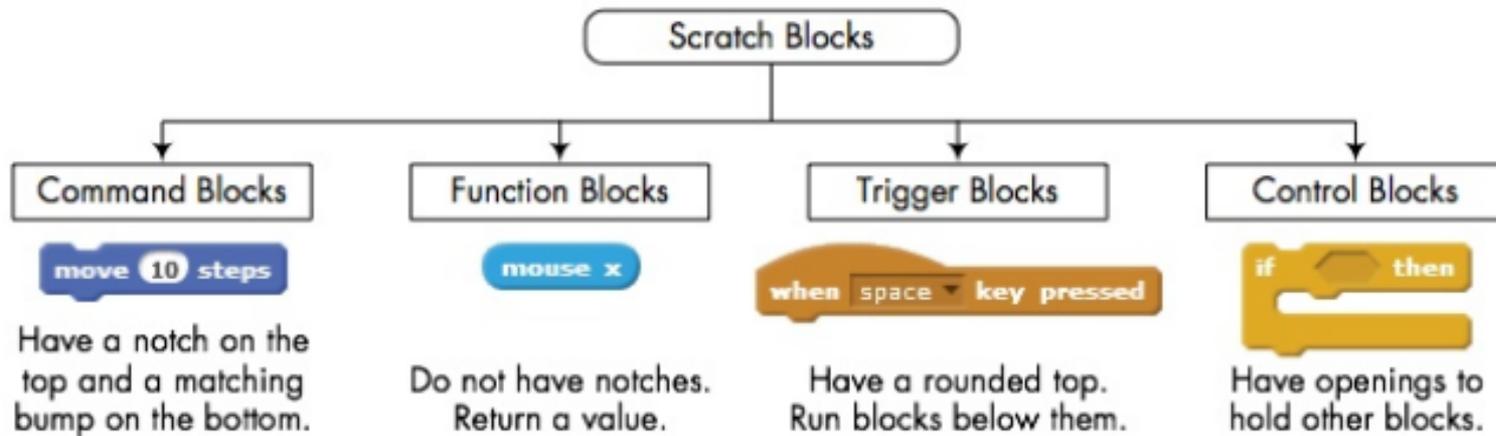
- La Blocks tab prevede 10 categorie di blocchi dette “palette”:
 - Motion (movimento)
 - Looks (aspetto)
 - Sound (suono)
 - Pen (penna)
 - Data (variabili e liste)
 - Events (situazioni)
 - Control (controllo)
 - Sensing (sensori)
 - Operators (operatori)
 - More Blocks (altri blocchi)



I blocchi si incastrano uno nell'altro e uno sull'altro.

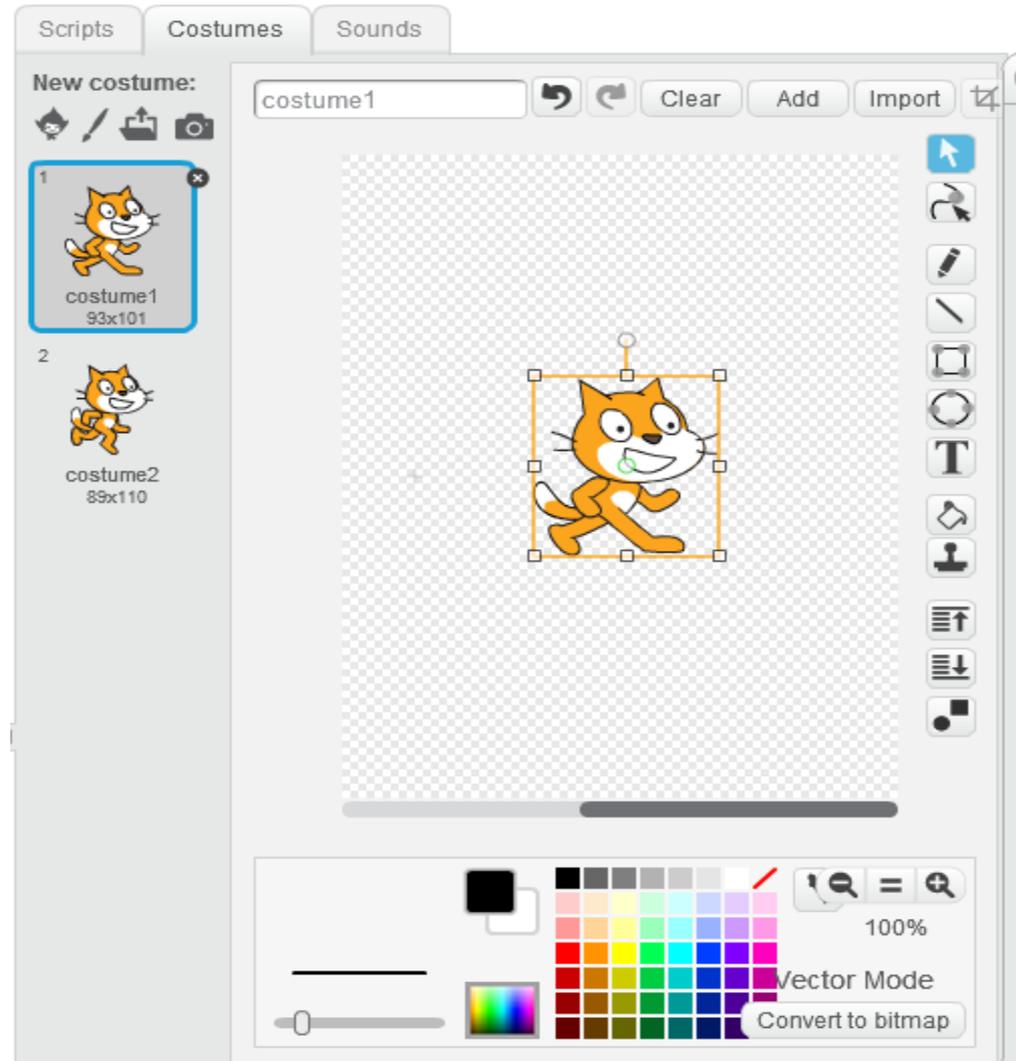
BLOCKS TAB - SCRIPT (2)

- *Command Block* e *Control Block* detti *stack block* (blocchi in pila)
- *Trigger Block*, detti *hats* (cappello)
- *Function Block*, detti *reporters*



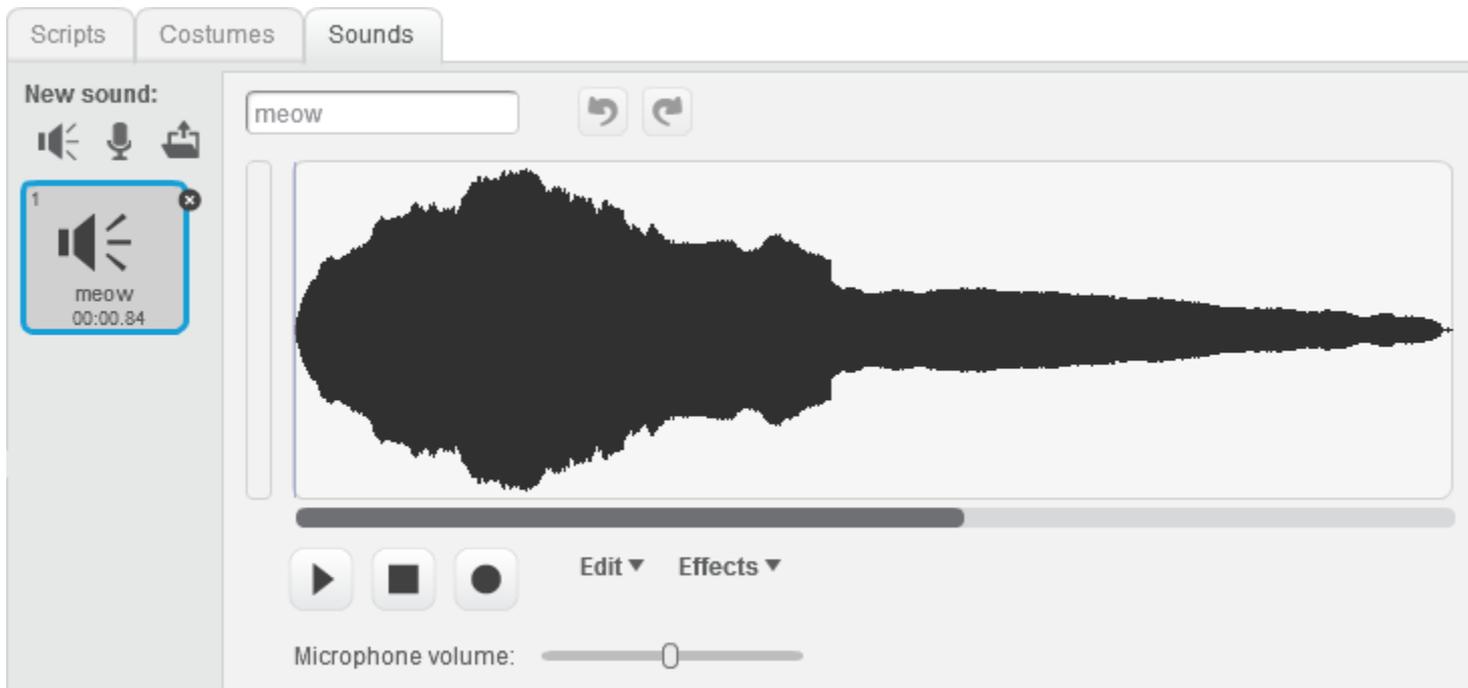
BLOCKS TAB: COSTUMI

- Il *Costumes Tab* permette di associare a ogni *sprite* un aspetto (costume) differente al fine di gestire nuovi stati in cui si trova lo *sprite* così da poter creare, ad esempio, piccole animazioni.



BLOCKS TAB: SOUNDS

- nel *Sounds Tab* è possibile organizzare, registrare e riprodurre i suoni da associare al comportamento degli *sprite*.

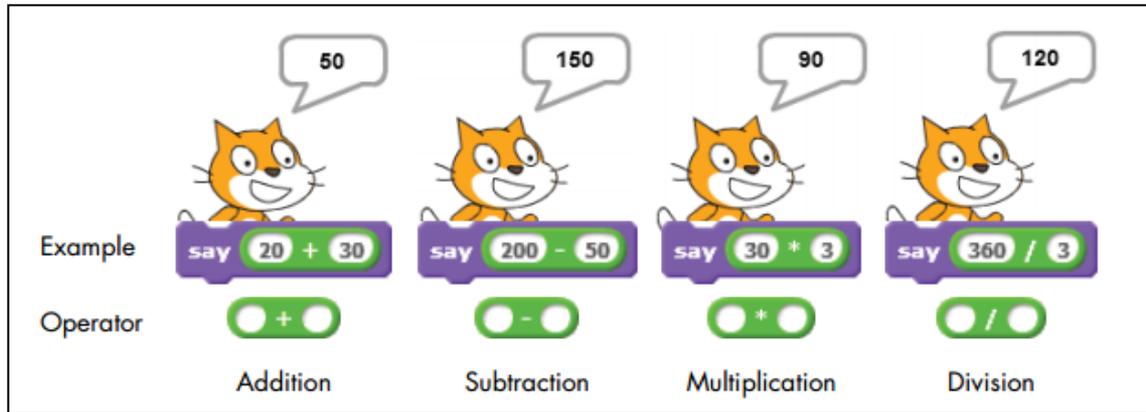


SCRIPT AREA

- Per scrivere lo script è necessario trascinare i blocchi in questo pannello.



ESEMPIO: OPERATORI ARITMETICI E FUNZIONI



- Generazione di numeri casuali

Example	Possible Outcome
pick random 0 to 1	{0, 1}
pick random 0 to 10	{0, 1, 2, 3, ... , 10}
pick random -2 to 2	{-2, -1, 0, 1, 2}
10 * pick random 0 to 10	{0, 10, 20, 30, ... , 100}
pick random 0 to 1.0	{0, 0.1, 0.15, 0.267, 0.3894, ... , 1.0}
pick random 0 to 100 / 100	{0, 0.01, 0.12, 0.34, 0.58, ... , 1.0}

PROBLEMI:

- Qual è il risultato del seguente scratch?
- Sia $x = 2$ e $y = 4$

a) $6 * x$

b) $2 * x + 4 * y$

c) $x * x$

d) $y + 4 / x * x$

e) $y * y / 2 * x + 2$

TO DO:

- Completa la tabella e usa il comando «say» e l'appropriato blocco «operator» per verificare la risposta.

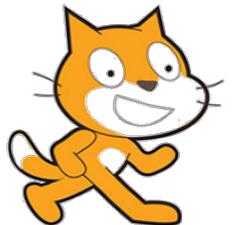
Expression	Value
$3 + (2 \times 5)$	
$(10 / 2) - 3$	
$7 + (8 \times 2) - 4$	
$(2 + 3) \times 4$	
$5 + (2 \times (7 - 4))$	
$(11 - 5) \times (2 + 1) / 2$	
$5 \times (5 + 4) - 2 \times (1 + 3)$	
$(6 + 12) \bmod 4$	
$3 \times (13 \bmod 3)$	
$5 + (17 \bmod 5) - 3$	

Lezione “Motion and Drawing”

Abilità Informatiche e Telematiche
2014/2015

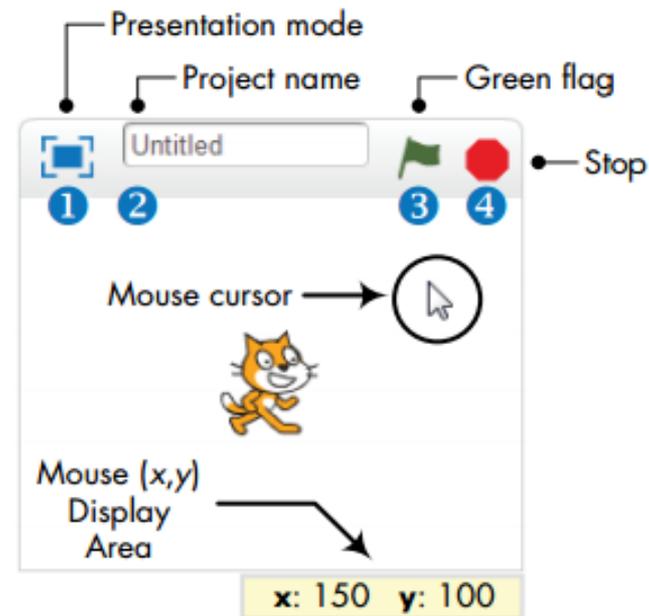
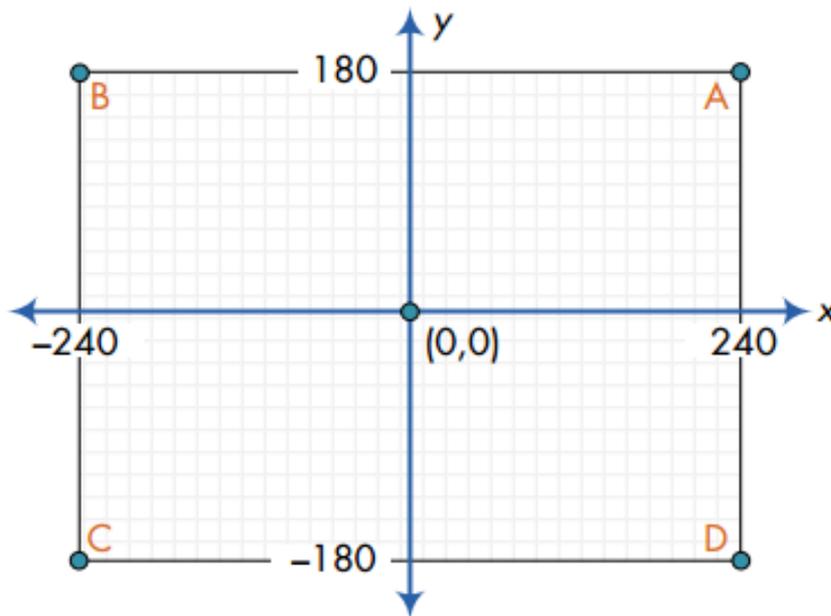
16

SCRATCH



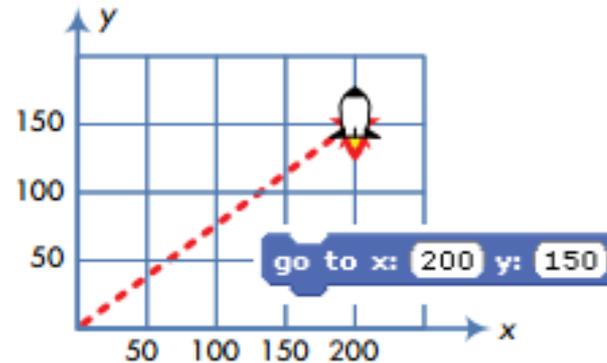
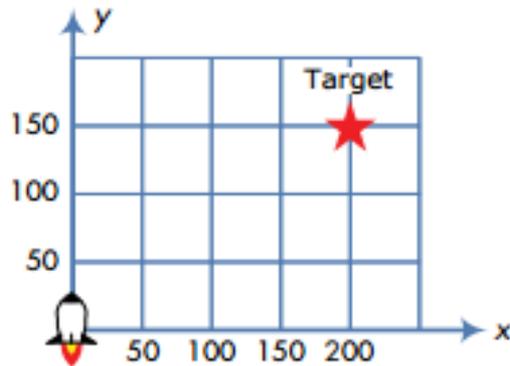
STAGE

- Lo Stage è lo schermo nel quale il risultato del nostro programma (col quale l'utente finale interagirà) sarà visualizzato.
- Ai diversi punti dello Stage si può accedere grazie alle note coordinate cartesiane.

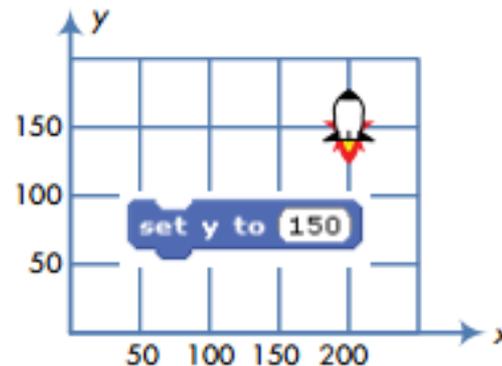
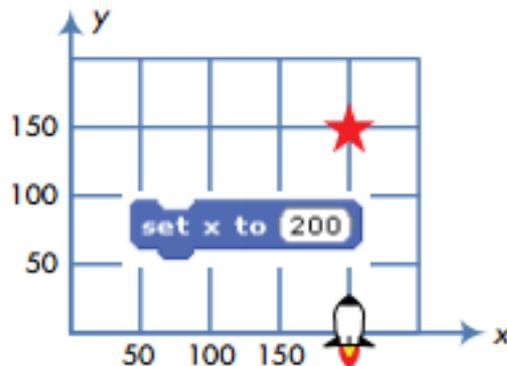


SPOSTAMENTI ASSOLUTI

- Per spostare uno sprite in una posizione specifica dello schermo è possibile utilizzare il seguente blocco (per considerare entrambe le dimensioni)

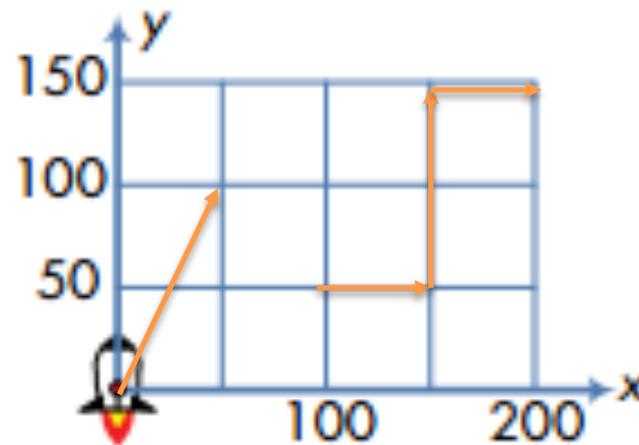


- Oppure i due blocchi:



ESEMPIO

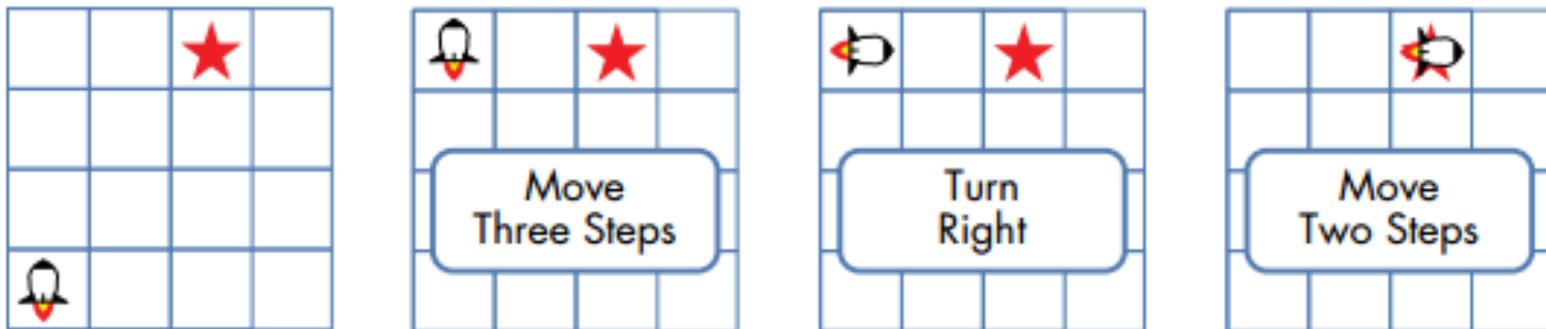
```
go to x: 50 y: 100  
set x to 150  
set y to 150  
set x to 200
```



SPOSTAMENTI RELATIVI

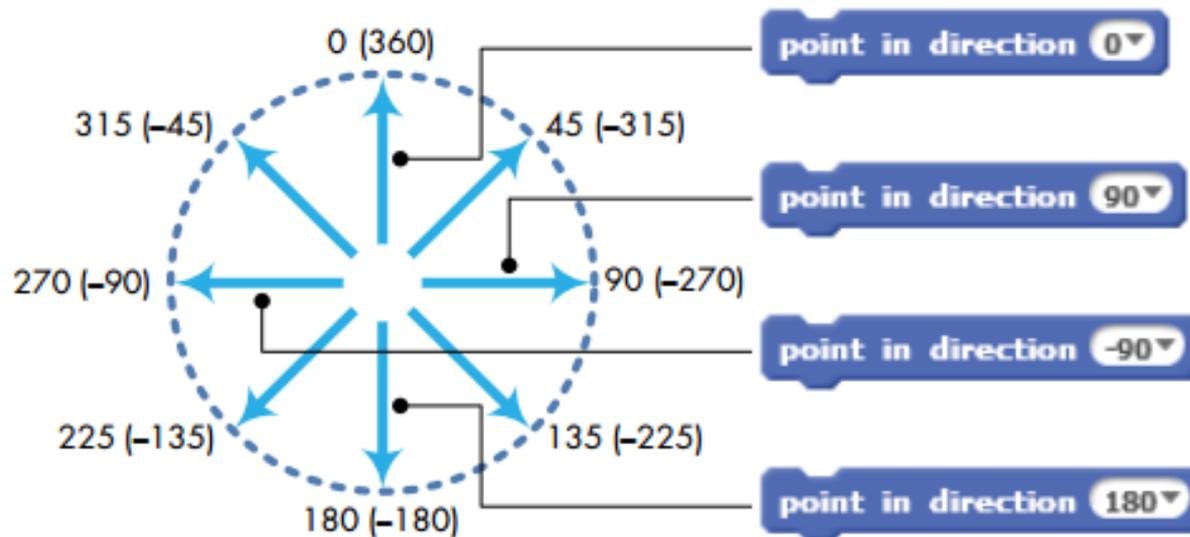
- In Scratch è possibile far compiere agli *sprite* anche degli spostamenti “relativi”, ovvero considerando la loro posizione corrente e la direzione in cui essi puntano (data dalla loro rotazione).

spostamento relativo realizzato in tre passi.

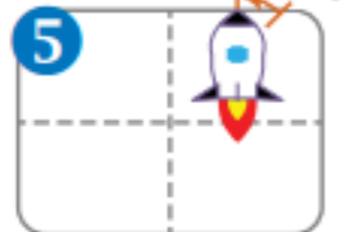
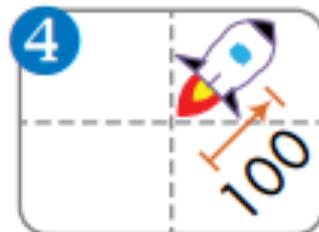
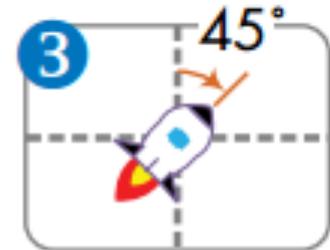
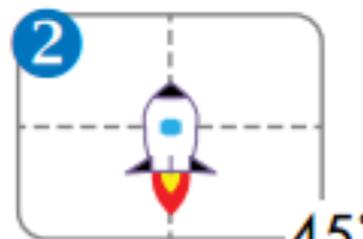
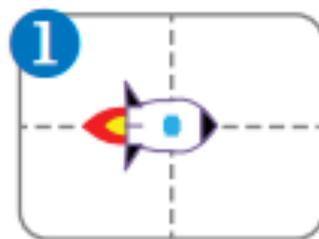
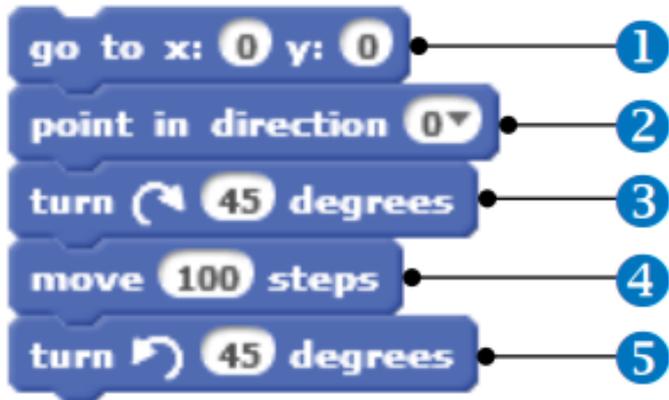


SPOSTAMENTI RELATIVI -2

- Per quanto riguarda, invece, il blocco *PUNTA IN DIREZIONE*, la figura seguente mostra il modo di indicare in maniera assoluta i parametri di rotazione dello *sprite*:

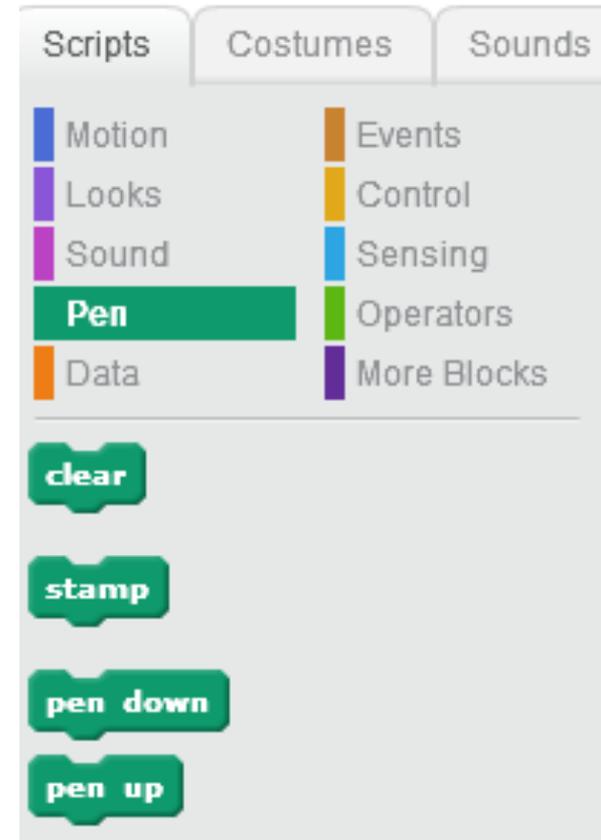


ESEMPIO



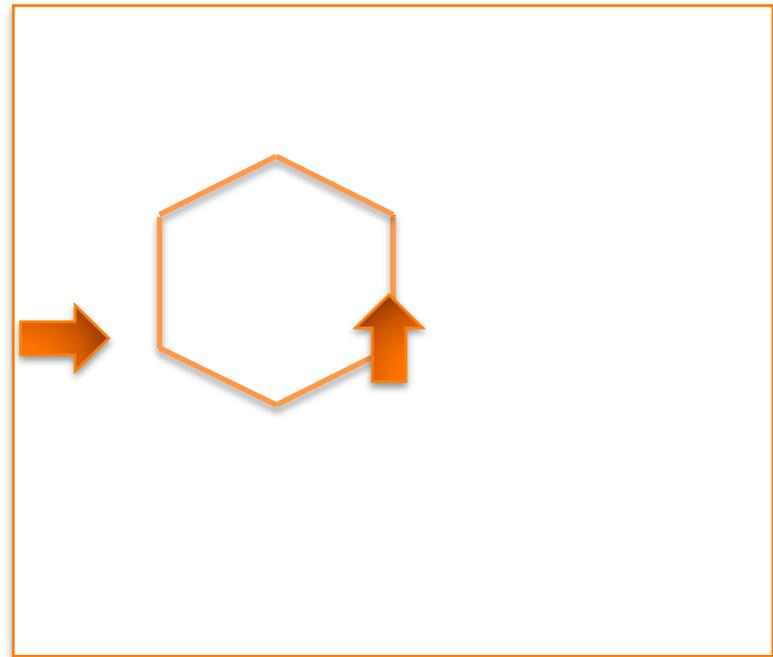
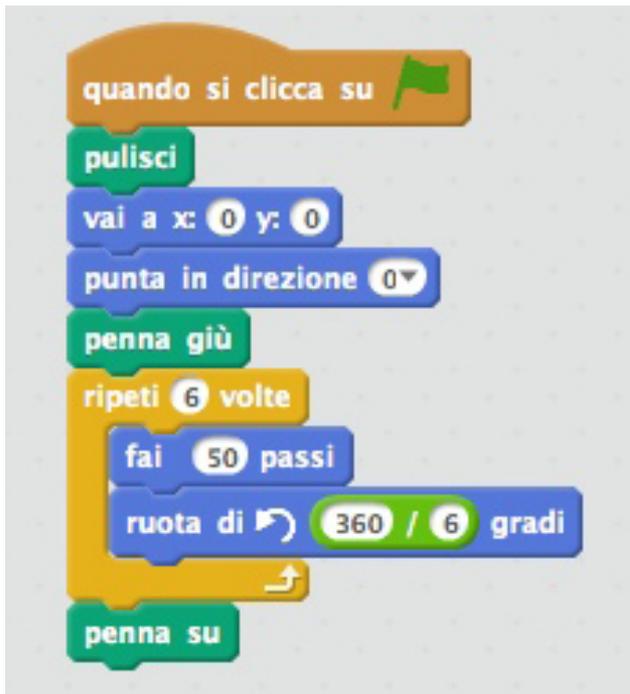
DISEGNO

- Per scrivere e disegnare sullo *Stage* è possibile utilizzare il blocchi PENNA.
- Basta utilizzare PENNA GIU' e far muovere lo *sprite* per rappresentare con un disegno il suo spostamento.
- Per smettere di scrivere è necessario utilizzare il blocco PENNA SU.
- Per cancellare il disegno bisogna utilizzare il blocco PULISCI.



ESEMPIO

- il programma per disegnare un esagono:



- Come disegniamo un quadrato?

ESEMPIO

- Simulazione DADO:
 - Quanti lanci facciamo?
 - Quale evento aspettiamo per un nuovo lancio del dado?



TO DO:

- Definire il codice scratch per disegnare un ottagono.