

Tutorial Python con Edublocks 1:

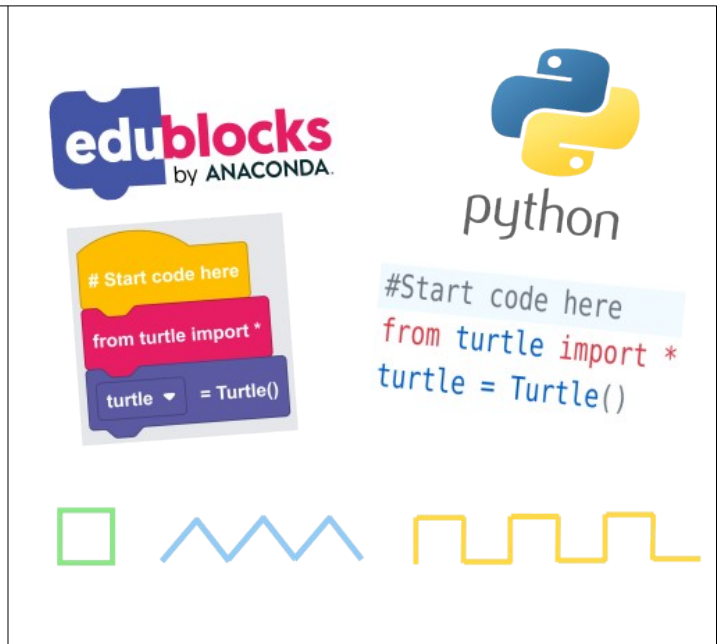
Le prime forme in turtle

Edublocks è un sito per imparare facilmente Python e altri linguaggi di programmazione. In particolare offre un'interfaccia visuale a blocchi comoda per i principianti (un po' come in Scratch, se l'hai già usato)

In questa guida imparerai a:

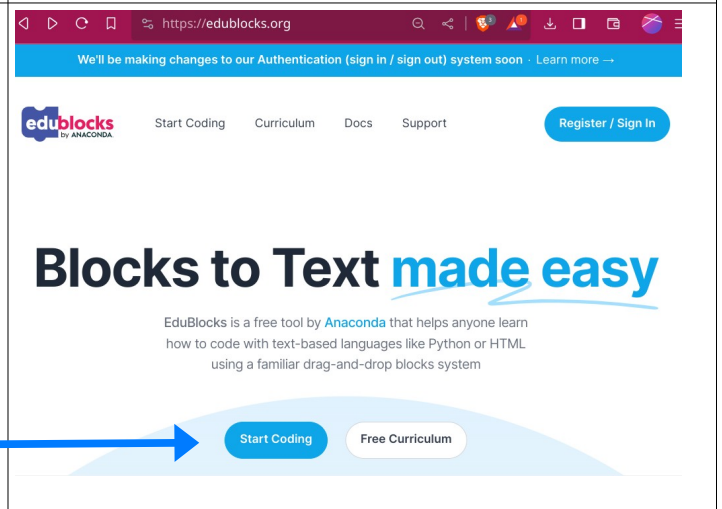
- realizzare il tuo primo programma in Python usando programmazione semplificata a blocchi
- usare il modulo `turtle` di Python per fare disegni e la codifica colori RGB

Nota: al momento (febbraio 2024) Edublocks non supporta tutte le funzionalità del modulo `turtle`



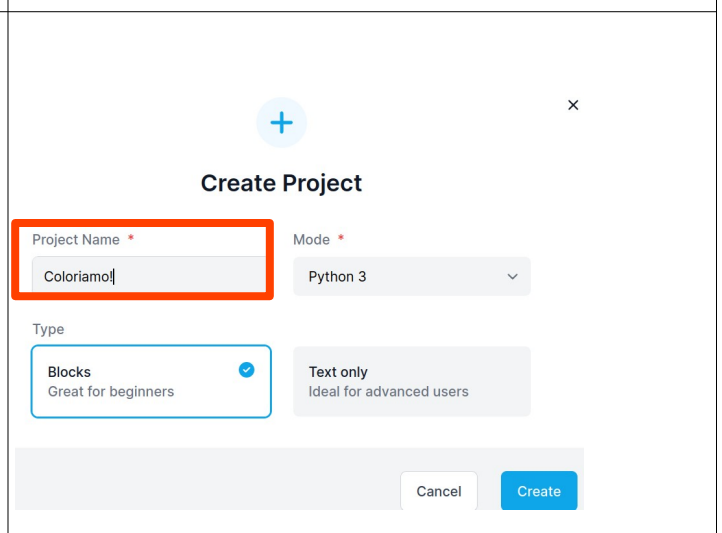
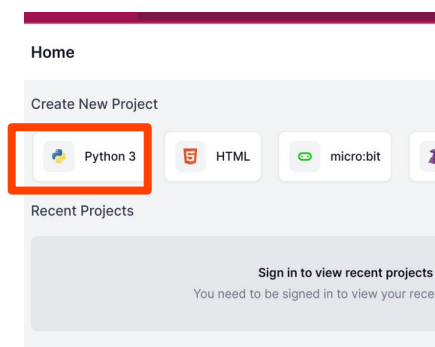
1. Iniziamo

Andiamo sul sito edublocks.org



e scegliamo **Start coding**:

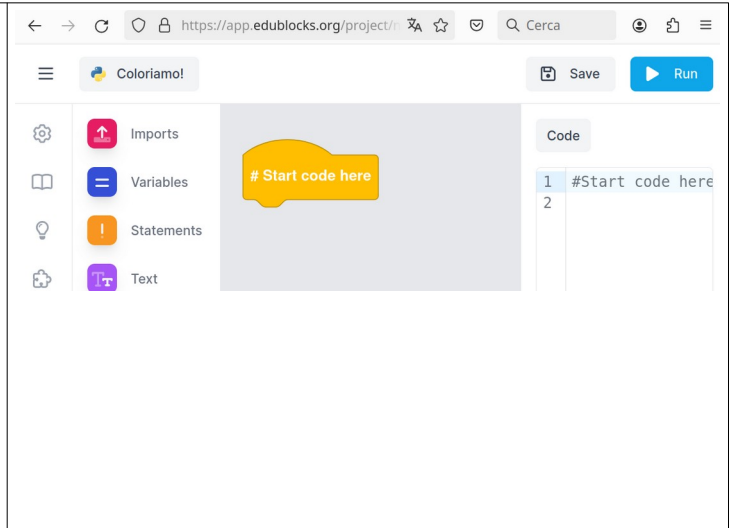
Poi clicchiamo **Python 3** e mettiamo il nome del progetto



Apparirà qualcosa del genere

(questa schermata è anche raggiungibile andando [a questo link](#))

ATTENZIONE: in questo tutorial usiamo SOLO la programmazione a blocchi visuali, quindi NON scrivere a destra sotto Code



2.1 Che cos'è libreria turtle ?

Con il linguaggio Python si possono fare mille cose, noi proveremo a creare qualche disegno con una *libreria* che si chiama `turtle`.

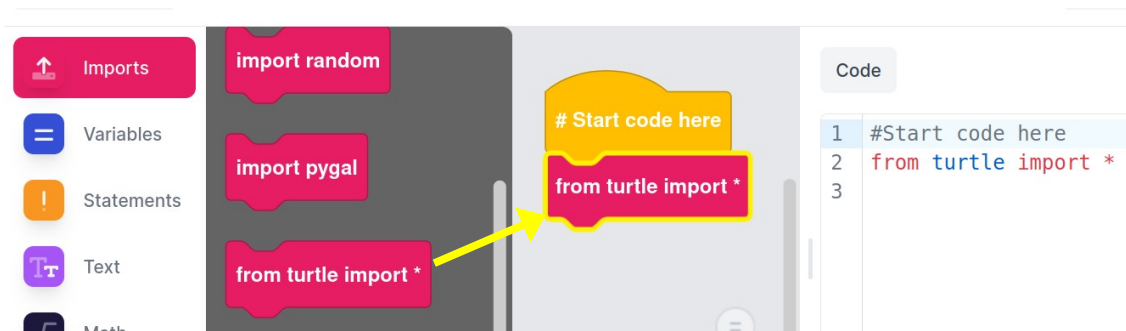
Le *librerie* in Python non sono degli scaffali con libri, bensì delle collezioni di codice che estendono Python per permetterti di fare più cose.

In particolare la libreria `turtle` ci consente di fare disegni. Iniziamo a dire a Python che vogliamo *importare* la libreria `turtle`.

2.2 Importiamola

Clicca la categoria Imports e trascina il blocco `from turtle import *`

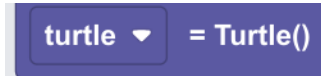
Nota come il corrispondente comando nel linguaggio Python appare nella finestra a destra Code:



3. Creiamo la tartaruga

Tutto in Python è un oggetto, quindi per proseguire ci servirà creare un oggetto tartaruga con l'assegnazione:

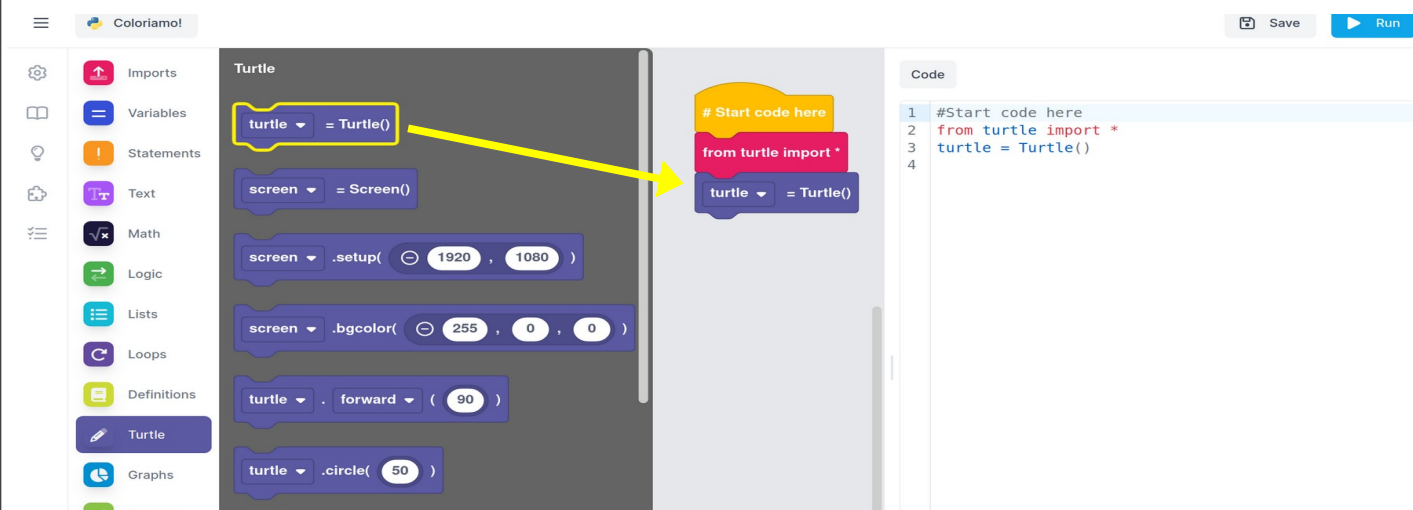
```
turtle = Turtle()
```



Che fa due cose:

1. crea un oggetto di *tipo* Turtle (nota la T maiuscola)
2. memorizza l'oggetto appena creato nella variabile turtle (ma avremmo potuto inventarci un qualunque nome per la variabile)

Troviamo il comando sempre nella categoria Turtle:



4.1 Muoviamo la tartaruga

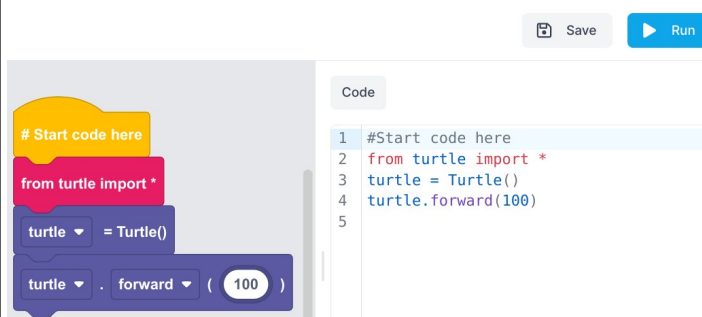
Adesso che abbiamo un oggetto tartaruga, come possiamo comandarla?

Bisogna scrivere il nome della variabile turtle seguito da un punto . e poi dal comando da impartire alla tartaruga, per esempio per andare avanti si può scrivere


```
turtle.forward(100)
```

mettendo tra parentesi tonde il numero di passi da fare.

Cerca il blocco forward nella categoria turtle



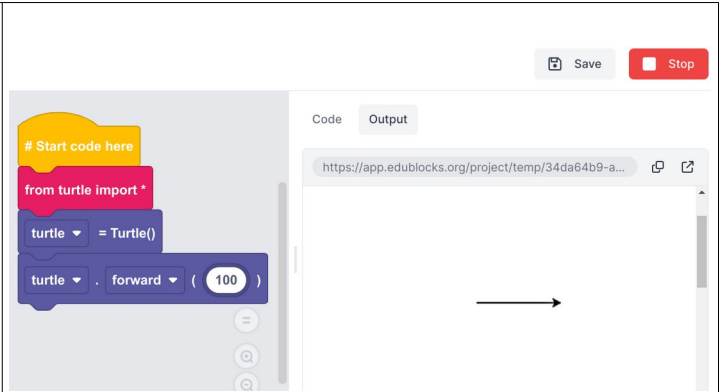
4.2 Esegui

Prova a cliccare  in alto a destra, dovresti vedere la tartaruga (che appare come una punta di freccia) che si è spostata di 100 passi

Ogni passo corrisponde a un punto dello schermo (detto *pixel*)

Dovresti vedere qualcosa del genere:

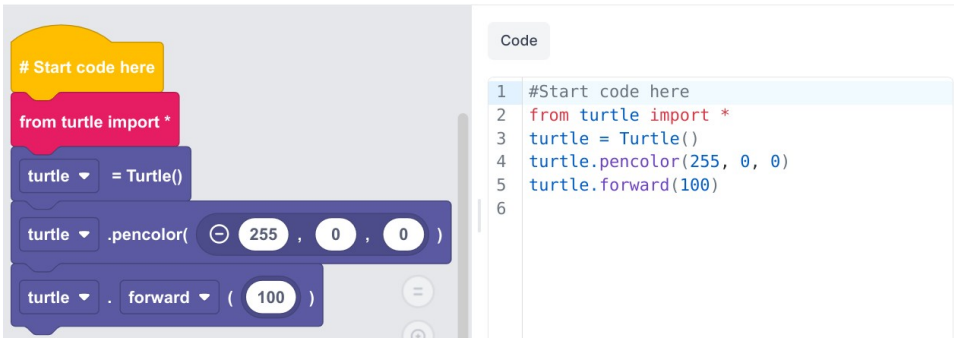
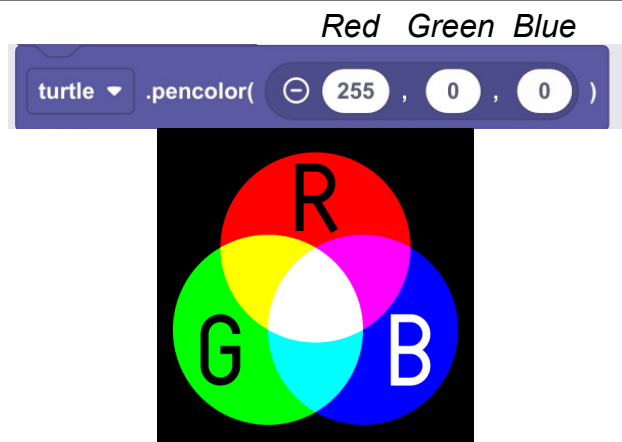
Per rivedere il codice Python, clicca 




5.1 Impostiamo un colore


Proviamo a cambiare il colore della linea tracciata con il comando `turtle.pencolor`. Il colore viene scomposto nelle cosiddette componenti *RGB*: *Red*, *Green*, *Blue*, che vengono miscelate per creare il colore finale con questo sistema.

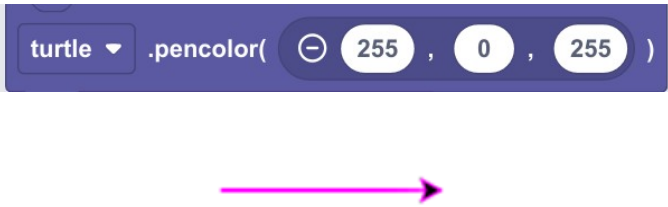
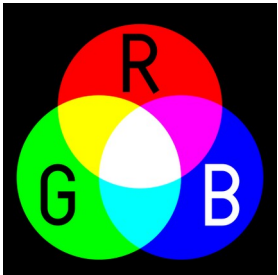
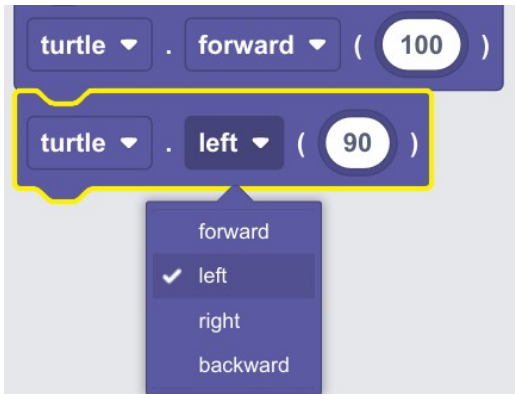
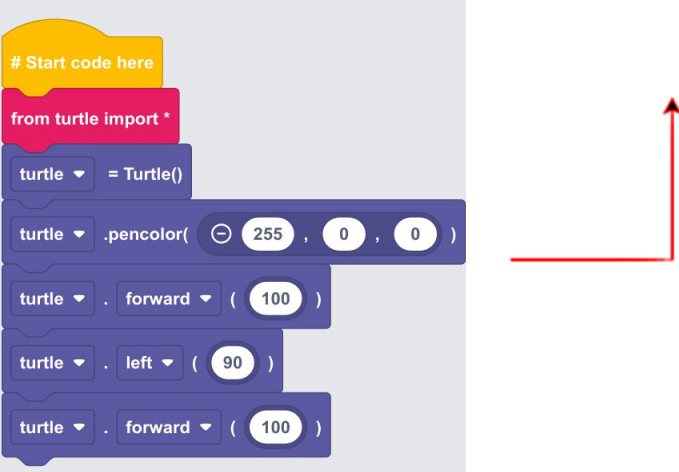

Ogni componente di colore è rappresentata da un numero che può andare da 0 a 255. Nel codice di esempio è puramente rossa



```
Code
1 #Start code here
2 from turtle import *
3 turtle = Turtle()
4 turtle.pencolor(255, 0, 0)
5 turtle.forward(100)
6
```

Cliccando  dovresti vedere questo:



<p>5.2 Prova a cambiare i numeri</p> <p>Agendo sulle componenti, si può comporre qualunque colore:</p> <p>Per esempio, questo dovrebbe generare il viola perché ha una forte componente rossa e una forte componente blu</p>	<p style="text-align: right;">Red Green</p> <p>Blue</p> 
<p>SFIDA: Prova a mostrare altri colori, per es. il giallo o l'azzurro!</p>	
<p>6. Girare</p> <p>La tartaruga può girare di alcuni gradi con i comandi <code>left</code> e <code>right</code>.</p> <p>Clicca su <code>forward</code> e scegli dal menu.</p>	
<p>Se poi aggiungi un altro <code>forward</code> e clicchi Run, dovresti vedere questo risultato:</p>	
<p>7 SFIDA: prova a disegnare un quadrato!</p>	

8.1 Un quadrato abbreviato

Per fare il quadrato di prima, probabilmente hai messo diversi blocchi uguali.

Per fortuna, Python ci consente di ridurre drasticamente il numero di blocchi usando il potente comando

```
for i in range(4):
```

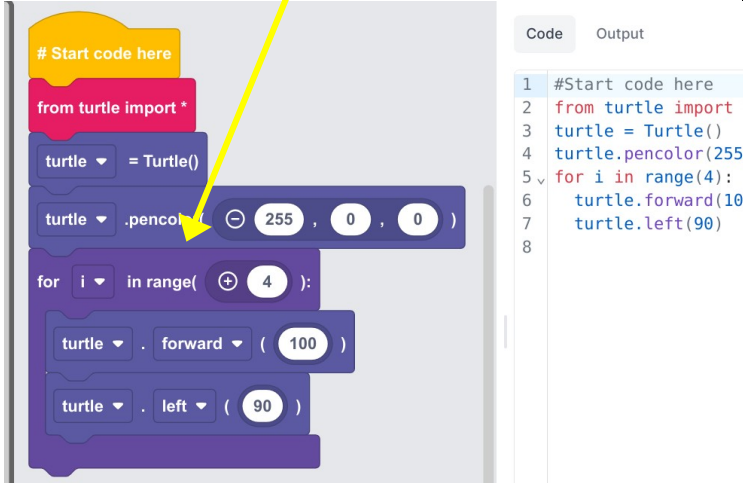
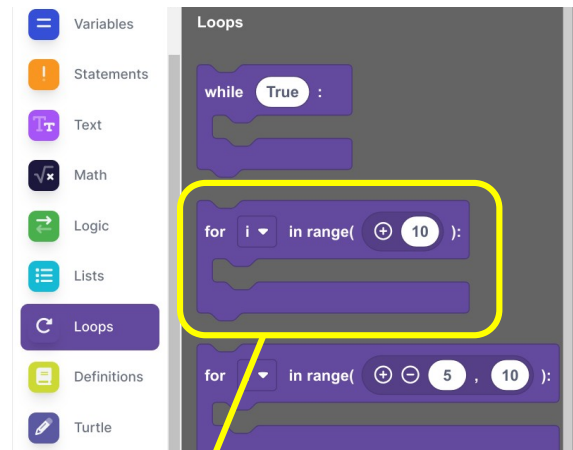
Che puoi trovare nella categoria `Loops`

ATTENZIONE:

Abbiamo cambiato il 10 in 4

ATTENZIONE:

abbiamo messo il `for` attorno ai due comandi `forward` e `left`



8.2 Provi a cliccare Run,

dovresti vedere un quadrato:



9. Salvare un progetto

Non vorrai mica perdere il lavoro fatto, vero?

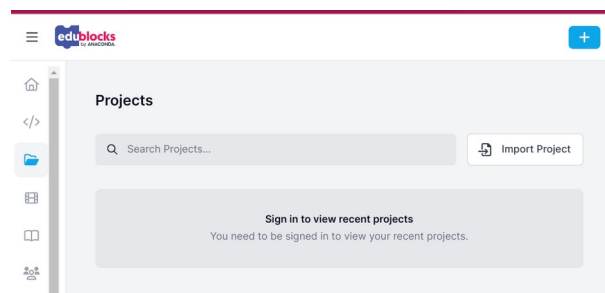
Per salvare il progetto, clicca `Save` in alto a destra



Caricare un progetto

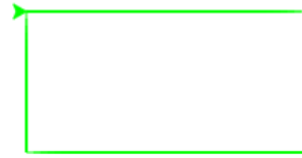
Se un domani vorrai ricaricarlo, ti basterà aprire il browser, andare alla pagina `projects` e cliccare `Import Project`:

app.edublocks.org/projects



10. Sfide

Riusciresti a disegnare.... un rettangolo???



E delle ondine ?

SUGGERIMENTO: meglio usare un `for`



E dei merletti?



Ti è piaciuto ? Trovi questo e altri tutorial gratuiti
alla sezione *Guide* del sito
coderdojotrento.it

Questo tutorial è stato realizzato grazie al sostegno di

CoderDolomiti  coderdolomiti.it