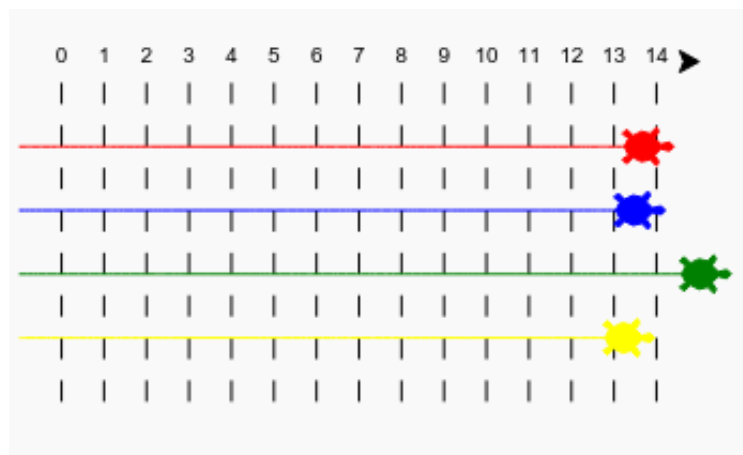


All Code Clubs must be registered. By registering your club we can measure our impact, and we can continue to provide free resources that help children learn to code. You can register your club at codeclubworld.org.

Uvod

U ovom projektu koristit ćeš petlje kako bi napravio utrku kornjača i nacrtao trkalište.



Activity Checklist



Test your Project



Save your Project

Follow these **INSTRUCTIONS** one by one

Click on the green flag to **TEST** your code

Make sure to **SAVE** your work now

Korak 1: Trkalište

Napravit ćeš igru u kojoj se utrkuju kornjače, ali prvo im treba trkalište.

✓ Zadatci

- Otvori prazan Python predložak u Trinketu: jump.to/cc/python-new. Ako čitaš ovo online, možeš koristiti i ugrađenu verziju ovog trinketa koja se nalazi ispod. ☐

- Kako bi nacrtao liniju sa 'kornjačom', dodaj sljedeći kôd: ☐

```
from turtle import *  
forward(100)
```



- Nacrtajmo sada s 'kornjačom' oznake trkališta za utrku. ☐

Naredba `write` ispisuje tekst na ekran.

Pokušaj:

```
from turtle import *  
write(0)  
forward(100)  
write(5)
```



- Sada moraš upisati brojeve između uglatih zagrada kako bi napravio oznake: ☐

```

write(0)
forward(20)
write(1)
forward(20)
write(2)
forward(20)
write(3)
forward(20)
write(4)
forward(20)
write(5)
forward(20)

```



- Primjećuješ li da se tvoj kôd poprilično ponavlja? Mijenjaju se ☐ jedino brojevi koje treba ispisati.

U Pythonu postoji bolji način da ovo napraviš. Možeš koristiti `for` petlju.

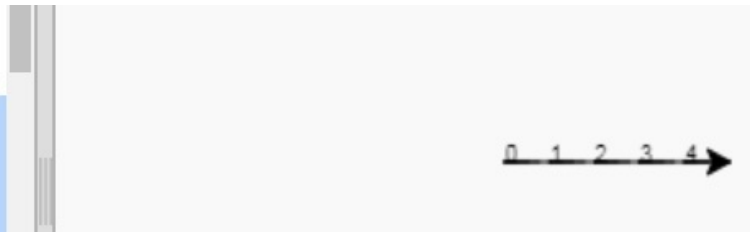
Izmijeni svoj kôd sa `for` petljom:

```

from turtle import *

for korak in range(5):
    write(korak)
    forward(20)

```

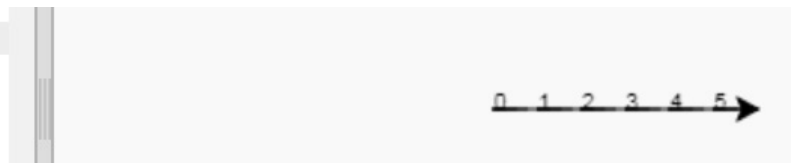


- Hmm, program ispisuje samo brojeve do 4. U Pythonu, ☐ `range(5)` znači da će program ispisati 5 brojeva, od 0 do 4. Ako želiš da ispiše i broj 5, moraš koristiti `range(6)`:

```

for korak in range(6):
    write(korak)
    forward(20)

```



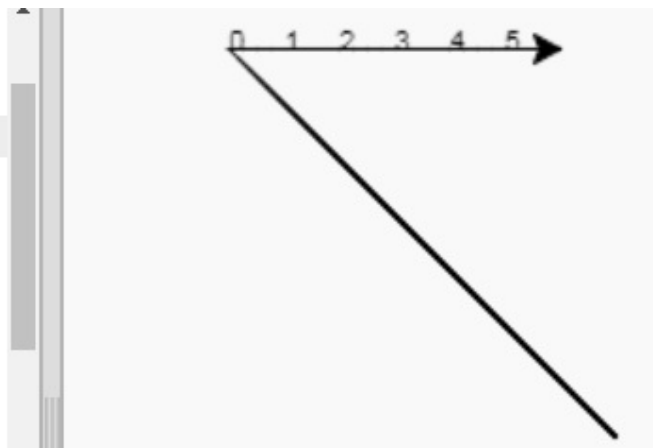
- Sada možemo nacrtati neke oznake trkališta. 'Kornjača' kreće ☐ od koordinata (0,0) koje se nalaze na sredini ekrana.

Pomakni 'kornjaču' u gornji lijevi kut:

```
from turtle import *
```

```
goto(-140, 140)
```

```
for korak in range(6):  
    write(korak)  
    forward(20)
```



- Ah, prvo ćeš morati podignuti olovku!



```
penup()
```

```
goto(-140, 140)
```

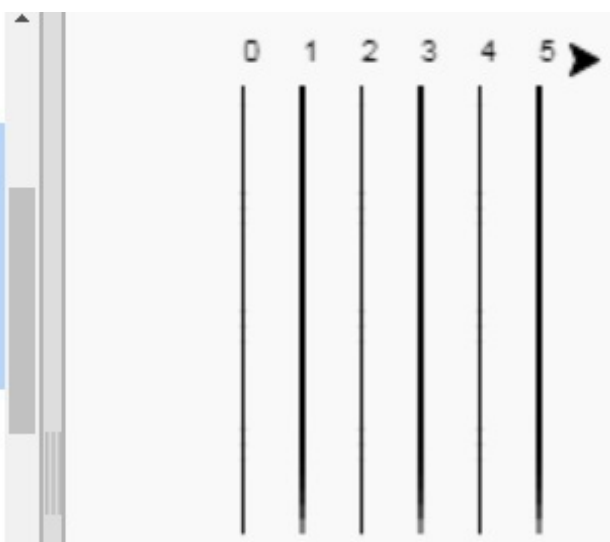
```
for korak in range(6):  
    write(korak)  
    forward(20)
```



- Umjesto da nacrtamo trkalište pomoću horizontalnih linija, nacrtajmo ga pomoću vertikalnih:



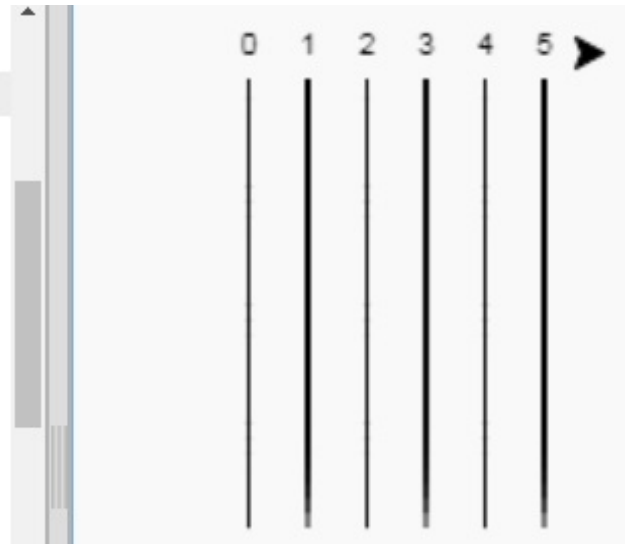
```
for korak in range(6):  
    write(korak)  
    right(90)  
    forward(10)  
    pendown()  
    forward(150)  
    penup()  
    backward(160)  
    left(90)  
    forward(20)
```



- Izgledat će urednije ako centriráš brojeve:



```
for korak in range(6):
    write(korak, align = 'center')
    right(90)
    forward(10)
    pendown()
    forward(150)
    penup()
    backward(160)
    left(90)
    forward(20)
```



- Također, možeš ubrzati 'kornjaču' kako bi crtala brže:



```
from turtle import *

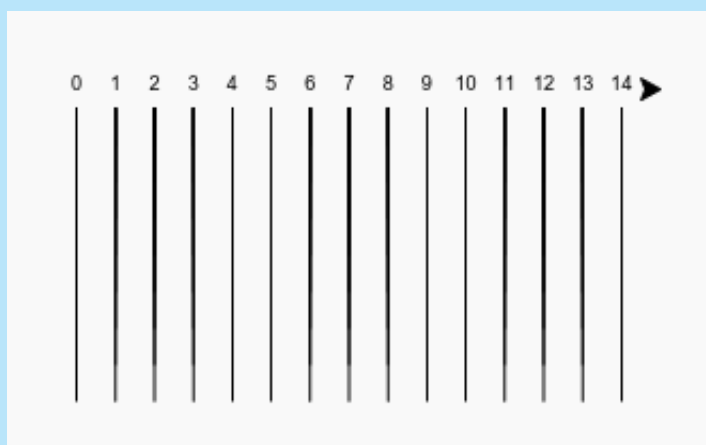
speed(10)
penup()
goto(-140, 140)
```



Spremi projekt

Izazov: Dodaj još linija

Možeš li izmijeniti kôd tako da linije idu preko cijelog ekrana?



Možeš koristiti `speed(0)` ako želiš ubrzati 'kornjaču'.

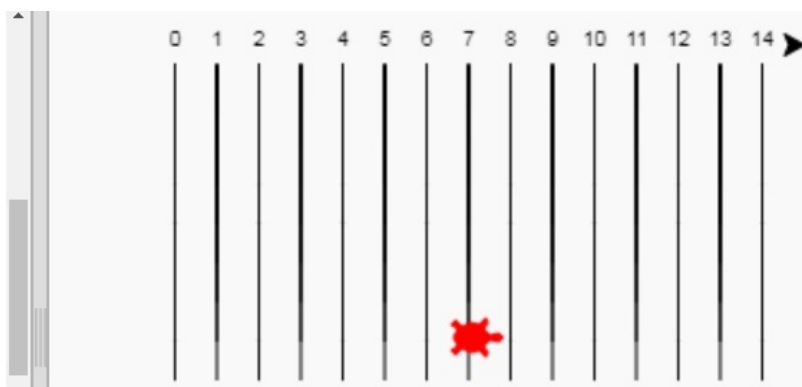
Korak 2: Kornjače za utrku

A sada zabavni dio. Dodajmo nekoliko kornjača koje će se utrkiivati. Bilo bi prilično dosadno kada bi kornjače svaki put radile istu stvar pa će se zato u svakom krugu pomicati za nasumični broj koraka. Pobjednik je ona kornjača koja se najdalje pomakne u 100 krugova.

✓ Zadatci

- Kada koristiš naredbe poput `forward(20)`, pomičeš samo jednu kornjaču. Ali, možeš napraviti više kornjača. Dodaj sljedeći kôd na kraju svog programa:

```
crvena = Turtle()  
crvena.color('red')  
crvena.shape('turtle')
```

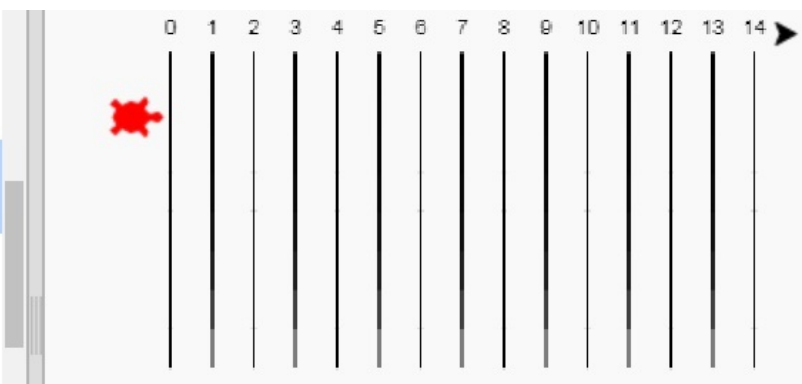


Prva linija kôda stvara kornjaču koja se zove 'crvena'. Ostale linije određuju boju i oblik kornjače. Sada stvarno izgleda kao kornjača!

- Pošaljimo kornjaču na startnu liniju:

```
crvena = Turtle()  
crvena.color('red')  
crvena.shape('turtle')
```

```
crvena.penup()  
crvena.goto(-160, 100)  
crvena.pendown()
```



- Sada moraš napraviti utrku pomičući se nasumičan broj koraka po stazi. Koristi funkciju `randint` iz Pythonove `random`

biblioteke. Dodaj `import` liniju na početak programa:

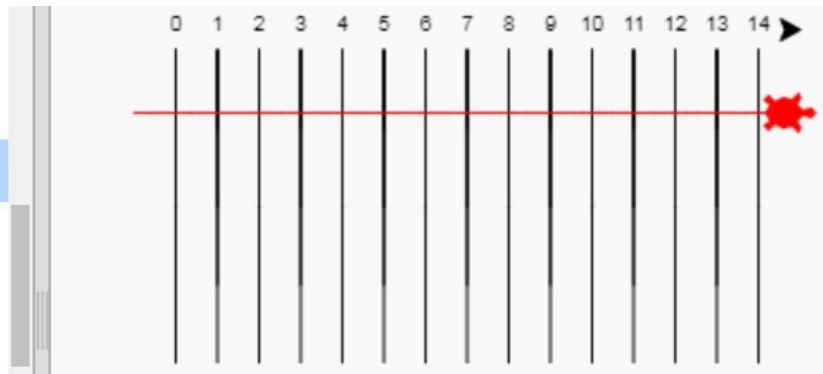
```
from turtle import *  
from random import randint
```

- Funkcija `randint` vraća nasumičan integer (cijeli broj) koji se nalazi između odabranih vrijednosti. Kornjača će se pomaknuti 1, 2, 3, 4 ili 5 koraka unaprijed u svakom krugu.



```
crvena.penup()  
crvena.goto(-160, 100)  
crvena.pendown()
```

```
for red in range(100):  
    crvena.forward(randint(1,5))
```

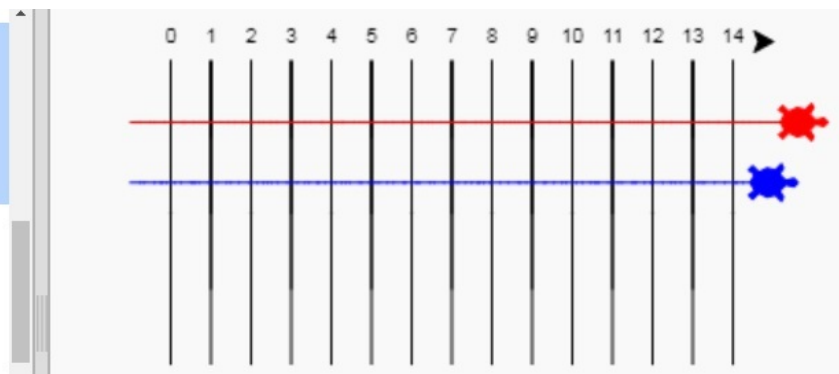


- Za utrku nam nije dovoljna samo jedna kornjača! Dodajmo još jednu:



```
plava = Turtle()  
plava.color('blue')  
plava.shape('turtle')  
  
plava.penup()  
plava.goto(-160, 70)  
plava.pendown()
```

```
for red in range(100):  
    crvena.forward(randint(1,5))  
    plava.forward(randint(1,5))
```



Primijeti da se kôd za pomicanje plave kornjače mora nalaziti unutar iste `for` petlje kao i kôd za pomicanje crvene kornjače. Tako će se obje kornjače pomicati u svakom krugu.

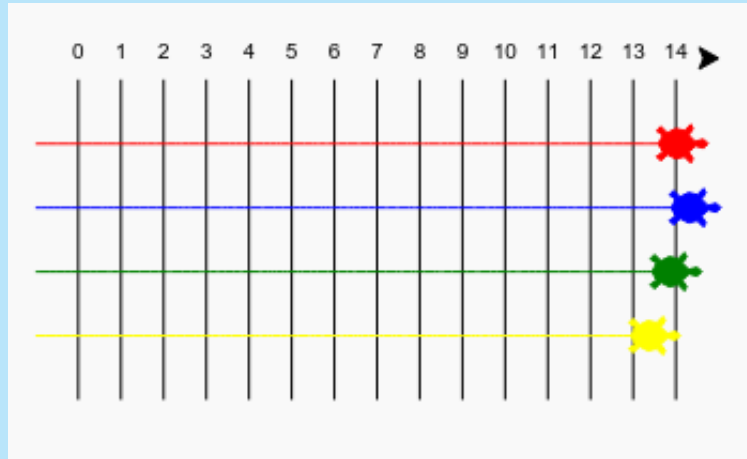


Spremi projekt

Izazov: Vrijeme je za utrku!

Sada si spreman za utrku. Odaberi kornjaču i njenog protivnika i pogledajmo tko će pobijediti.

Možeš li dodati još kornjača kako bi se mogao utrkiivati sa više prijatelja?



Uključene su sljedeće boje: 'orange', 'purple', 'violet', 'tomato', 'turquoise', 'magenta' and 'brown' - ili možeš otići na jump.to/cc/colour-picker i odabrati boju koju želiš!



Spremi projekt

Izazov: Napravi zaokret

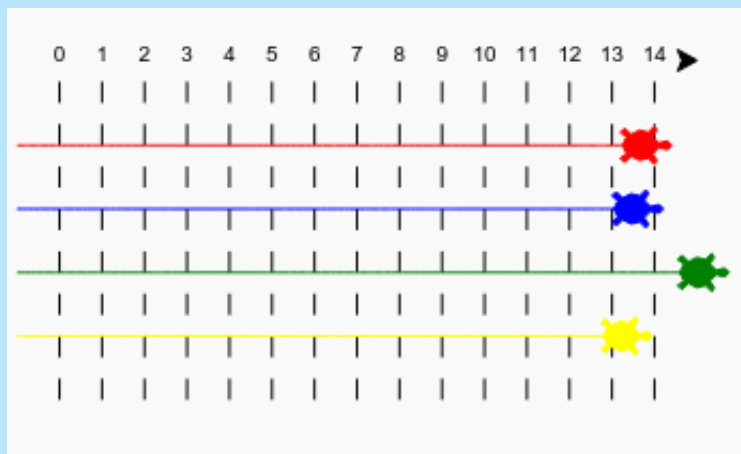
Možeš li koristiti petlju `for red in range():` kako bi svaka kornjača napravila zaokret od 360 stupnjeva nakon što dođe do startne linije? Pobrini se da su kornjače okrenute u pravom smjeru na početku utrke!

`crvena.right(36)` okrenuti će crvenu kornjaču za 36 stupnjeva udesno.

Pomoć: Puni krug iznosi 360 stupnjeva. Kornjača se može okrenuti 10 stupnjeva udesno 36 puta. Ili 5 stupnjeva u lijevo 72 puta. Ili ...

Izazov: Isprekidane linije

Možeš li, uz pomoć petlje, napraviti isprekidane linije trkališta umjesto punih crta?



Pomoć: Pronađi kôd kojim se crtaju ravne crte. Pokušaj koristiti naredbe: `for`, `forward()`, `penup()` i `pendown()`



Spremi projekt