

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](https://codeclubworld.org).

## Uvod

U ovom projektu učitavat ćeš podatke iz datoteke i kreirati karte za igru robotima. Zatim ćeš s prijateljem moći igrati Borbe robota.



**Activity Checklist**

Follow these **INSTRUCTIONS** one by one



**Test your Project**

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



**Save your Project**

Make sure to **SAVE** your work now

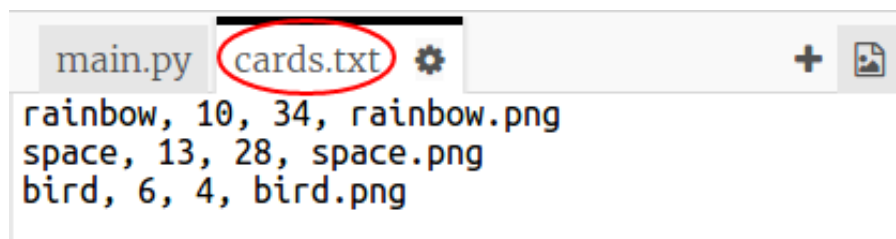
## Korak 1: Učitaj podatke o robotu iz datoteke

Često je korisno moći učitati podatke iz datoteke. Na taj način ne trebaš mijenjati svoj kôd već možeš izmijeniti podatke unutar datoteke.

### ✓ Zadatci

- Otvori ovaj trinket: [jump.to/cc/trumps-go](https://jump.to/cc/trumps-go). Ako čitaš ovo online, možeš koristiti i ugrađenu verziju ovog trinketa koja se nalazi ispod. ☐
- U tvom početnom projektu nalazi se datoteka `cards.txt` koja sadrži podatke o robotima. ☐

Klikni na `cards.txt` za pregled podataka:



Svaka linija sadrži podatke o jednom robotu. Podatci unutar linija odvojeni su zarezima.

Svaka linija sadrži sljedeće podatke:


ime, razinu inteligencije, izdrživost baterije, naziv slikovne datoteke

- Učitajmo podatke iz datoteke kako bi ih mogli koristiti. ☐


Prvo otvori datoteku `cards.txt` u svom programu:

```
datoteka = open('cards.txt', 'r')
```

- Sada možeš učitati podatke iz datoteke: ☐


```
main.py cards.txt + 
#!/bin/python3

datoteka = open('cards.txt', 'r')
print(datoteka.read())
```


Powered by  trinket  
rainbow, 10, 34, rainbow.png  
space, 13, 28, space.png  
bird, 6, 4, bird.png

- Uvijek zatvori datoteku nakon što si završio s njenim korištenjem:



```
main.py cards.txt + 
#!/bin/python3


datoteka = open('cards.txt', 'r')
print(datoteka.read())
datoteka.close()
```

Powered by  trinket  
rainbow, 10, 34, rainbow.png  
space, 13, 28, space.png  
bird, 6, 4, bird.png


- Time smo ispisali podatke kao jedan string. Rastavimo ga na pojedinačne dijelove podataka.



Prvo možeš ispisati podatke u obliku liste koja se sastoji od linija:

```
main.py cards.txt + 
#!/bin/python3


datoteka = open('cards.txt', 'r')
print(datoteka.read().splitlines())
datoteka.close()
```

Powered by  trinket  
['rainbow, 10, 34, rainbow.png', 'space, 13, 28, space.png', 'bird, 6, 4, bird.png']

Pažljivo pogledaj ispis. U listi se nalaze tri elementa, a svaki je element jedna linija iz datoteke.

- Sada možeš petljom proći kroz svaku liniju posebno.




```
main.py cards.txt + 
#!/bin/python3

datoteka = open('cards.txt', 'r')

for linija in datoteka.read().splitlines():
    print(linija)

datoteka.close()
```

Powered by  trinket  
rainbow, 10, 34, rainbow.png  
space, 13, 28, space.png  
bird, 6, 4, bird.png

- Umjesto ispisivanja linija, pohrani ih u varijable:



```
datoteka = open('cards.txt', 'r')

for linija in datoteka.read().splitlines():
    ime, baterija, inteligencija, slika = linija.split(',')

datoteka.close()
```

- Želiš da ti ovi podatci budu i kasnije dostupni za pronalaženje vrijednosti određenog robota. Koristit ćemo ime robota kao ključ u rječniku.



Kreiraj rječnik naziva `roboti`:

```
roboti = {}

datoteka = open('cards.txt', 'r')

for linija in datoteka.read().splitlines():
    ime, baterija, inteligencija, slika = linija.split(',')

datoteka.close()
```

- Dodajmo sada unos za svakog robota u rječnik `roboti`.



Ime robota je ključ, a lista podataka za tog robota je vrijednost.

Dodaj označeni kôd:

main.py cards.txt
+

```
#!/bin/python3

roboti = {}

datoteka = open('cards.txt', 'r')

for linija in datoteka.read().splitlines():
    ime, baterija, inteligencija, slika = linija.split(',')
    roboti[ime] = [baterija, inteligencija, slika]

datoteka.close()

print(roboti)
```

Powered by **trinket**

```
{'rainbow': ['10', '34', 'rainbow.png'], 'space': ['13', '28', 'space.png'],
'bird': ['6', '4', 'bird.png']}
```

Možeš maknuti naredbu `print roboti` nakon testiranja kôda.

## Korak 2: Prikaži podatke

Sada možeš prikazati podatke o robotima na zanimljiviji način.

Prikažimo kartu robota sa slikom i podacima o njegovoj inteligenciji i korisnosti.

Kada to napraviš moći ćeš prikazati robote na ovaj način:



Ime: space  
Baterija: 13  
Inteligencija: 28

### ✓ Zadatci

- Pitaj korisnika da odabere robota: ☐

```
main.py cards.txt +  Powered by  trinket
datoteka.close()

while True:
    robot = input('Odaberi robota: ')
```

- Ako se robot nalazi u rječniku, potraži njegove podatke: ☐


```
main.py cards.txt +  Powered by  trinket
robot = input('Odaberi robota: ')


if robot in roboti:
    print(roboti[robot])
```

Testiraj kôd unošenjem imena robota.

- Ako robot ne postoji, prikaži poruku o grešci:



```
main.py cards.txt +  trinket  
if robot in roboti:  
    print(roboti[robot])  
else:  
    print('Robot ne postoji!')
```

Powered by  **trinket**  
Odaberi robota: shiny  
Robot ne postoji!  
Odaberi robota:

Testiraj kôd unošenjem imena robota koji se ne nalazi u rječniku.

- Sada ćeš koristiti Python turtle za prikazivanje podataka robota.



Uvezi turtle biblioteku na početku svog kôda i podesi ekran i turtle:

```
from random import choice  
from turtle import *  
  
ekran = Screen()  
ekran.bgcolor('white')  
penup()  
hideturtle()  
  
roboti = {}
```

- Zatim dodaj kôd kojim će turtle ispisati ime robota:



```
if robot in roboti:  
    print(roboti[robot])  
    stil = ('Arial', 14, 'bold')  
    write('Ime: ' + robot, font=stil, align='center')
```

Ime: space

- Izmjenjuj varijablu `stil` dok ne budeš zadovoljan sa izgledom teksta.



Umjesto stila `Arial` možeš iskušati stilove: `Courier`, `Times` ili `Verdana`.

Promijeni `14` u neki drugi broj za izmjenu veličine fonta.

Možeš promijeniti `bold` u `normal` ili `italic`.


- Umjesto ispisivanja karakteristika robota, pohrani ih u varijablu. ☐

```
if robot in roboti:  
    karak = roboti[robot]  
    stil = ('Arial', 14, 'bold')  
    write('Ime: ' + robot, font=stil, align='center')
```

- Sada možeš karakteristikama robota pristupiti kao elementima liste: ☐
- `karak[0]` označava inteligenciju ☐
- `karak[1]` označava bateriju ☐
- `karak[2]` označava naziv slikovne datoteke ☐

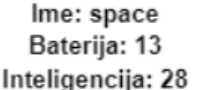
Dodaj kôd koji će prikazivati karakteristike inteligencije i baterije:

```
if robot in roboti:  
    karak = roboti[robot]  
    stil = ('Arial', 14, 'bold')  
    write('Ime: ' + robot, font=stil, align='center')  
    write('Baterija: ' + karak[0], font=stil, align='center')  
    write('Inteligencija: ' + karak[1], font=stil, align='center')
```



- Oh ne! Karakteristike se ispisuju jedna preko druge. Moramo dodati kôd kojim će se turtle pomaknuti: ☐

```
if robot in roboti:  
    karak = roboti[robot]  
    stil = ('Arial', 14, 'bold')  
    setheading(-90)  
    write('Ime: ' + robot, font=stil, align='center')  
    forward(25)  
    write('Baterija: ' + karak[0], font=stil, align='center')  
    forward(25)  
    write('Inteligencija: ' + karak[1], font=stil, align='center')
```



- Konačno, dodajmo sliku robota za upotpunjavanje prikaza. ☐

Moraš dodati liniju za prepoznavanje slike pri učitavanju podataka iz datoteke `cards.txt`:

```
for linija in datoteka.read().splitlines():
    ime, baterija, inteligencija, slika = linija.split(',')
    roboti[ime] = [baterija, inteligencija, slika]
    ekran.register_shape(slika)

datoteka.close()
```

- Dodaj i kôd za postavljanje slike: ☐

```
karak = roboti[robot]
stil = ('Arial', 14, 'bold')
goto(0, 100)
shape(karak[2])
setheading(90)
stamp()
setheading(-90)
forward(70)
write('Ime: ' + robot, font=stil, align='center')
forward(25)
```



- Testiraj kôd unošenjem jednog robota pa drugog i vidjet ćeš da se prikazuju jedan na drugome! ☐

Moraš očistiti ekran prije prikazivanja robota:

```
if robot in roboti:
    karak = roboti[robot]
    stil = ('Arial', 14, 'bold')
    clear()
    goto(0, 100)
    shape(karak[2])
    setheading(90)
    stamp()
```



Spremi projekt



## Izazov: Dodaj još robota

Možeš li u datoteku `cards.txt` dodati podatke za još robota?

Klikni na gumb za prikaz slika kako bi vidio koje slike robota su ti dostupne.

Ti odlučuješ koliko baterije i inteligencije će imati.



Spremi projekt

## Izazov: Dodaj još karakteristika robotima

Možeš li smisliti još karakteristika koje ćeš dodati robotima?

Možeš dodati 'brzinu' ili 'korisnost' ili smisliti nešto drugo.

Trebaš učiniti sljedeće:

- ☐ Dodati podatke u datoteku za svaku novu karakteristiku
- ☐ Dodati novu karakteristiku dijelu kôda koji učitava podatke
- ☐ Ispisati novu karakteristiku pri prikazivanju karte robota

Možeš dodati i boju i prikazati karakteristike svakog robota u drugoj boji.

Pomoć: Koristi `color('red')` za promjenu turtle teksta u crvenu prije pisanja.

Primjer:



Spremi projekt

## Korak 3: Prikaži nasumičnog robota

Dodajmo kôd kojim ćeš dobiti nasumično odabranog robota kada upišeš

Random umjesto njegovog imena.


## Zadatci

- Prvo moraš uvesti choice funkciju iz modula random: ☐

```
from turtle import *  
from random import choice
```

- Možeš koristiti `choice` za odabir nasumičnog imena robota iz liste ključeva u rječniku roboti. ☐

```
robot = input("Odaberi robota: ")  
  
if robot == "random":  
    robot = choice(list(roboti.keys()))  
    print(robot)
```

Powered by  trinket  
Odaberi robota: random  
rainbow  
Odaberi robota:

U Pythonu 3 moraš koristiti `list` za pretvaranje rezultata `ključeva` (keys) u listu.

Savjet: Pripazi na zagrade!



Spremi projekt

## Izazov: Zaigraj Borbe robota sa prijateljem

Podijeli svoj projekt sa prijateljem i zaigrajte Borbe robota. Oboje koristite isti projekt da bude pošteno! Prvi igrač poziva nasumično odabranog robota, a zatim bira kategoriju. Tada i drugi igrač poziva nasumično odabranog robota, a zatim provjeravate tko ima najviši rezultat u odabarnoj kategoriji. Nakon toga se mijenjate.

Igra najbolje funkcioniра ako oboje igrate sa istim špilom karata. Podijeli link svog trinket projekta sa prijateljem kako bi oboje mogli koristiti isti špil.



Spremi projekt