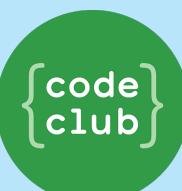


Python  
**2**

# Gdje je svemirska postaja?



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](http://codeclubworld.org).

## Uvod

U ovom projektu koristit ćeš web uslugu kako bi saznao gdje se trenutno nalazi Međunarodna svemirska postaja (ISS) i ucrtao njenu lokaciju na kartu.



### 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: Tko je u svemiru?

Koristit ćeš web uslugu koja uživo pruža informacije o svemiru. Saznajmo prvo tko se trenutno nalazi u svemiru.

## Zadatci

- Web usluga ima svoju adresu (url) baš kao i web stranica, ali umjesto prikazivanja HTML-a, prikazuje podatke.

Otvori <http://api.open-notify.org/astros.json> u internet pregledniku.

Trebao bi vidjeti nešto poput ovoga:

```
{  
  "message": "success",  
  "number": 3,  
  "people": [  
    {  
      "craft": "ISS",  
      "name": "Yuri Malenchenko"  
    },  
    {  
      "craft": "ISS",  
      "name": "Timothy Kopra"  
    },  
    {  
      "craft": "ISS",  
      "name": "Timothy Peake"  
    }  
  ]  
}
```

Podatci se prikazuju uživo pa ćeš vidjeti drugačiji rezultat. Format se zove JSON.

- Pozovimo web uslugu u Pythonu kako bi vidjeli rezultate.

Otvori ovaj trinket: [jumpto.cc/iss-go](https://jumpto.cc/iss-go). Ako čitaš ovo online, možeš koristiti i ugrađenu verziju ovog trinketa koja se nalazi ispod.

- Module `urllib.request` i `json` smo već uvezli za tebe.



Dodaj sljedeći kôd u `main.py` kako bi web adresu koju smo upravo koristili pohranio u varijablu:

```
url = 'http://api.open-notify.org/astros.json'
```

- Pozovimo sada web uslugu:

```
url = 'http://api.open-notify.org/astros.json'  
odgovor = urllib.request.urlopen(url)
```



- Zatim moraš učitati JSON odgovor u Pythonovu strukturu podataka:

```
url = 'http://api.open-notify.org/astros.json'  
odgovor = urllib.request.urlopen(url)  
rezultat = json.loads(odgovor.read())  
print(rezultat)
```



Trebao bi vidjeti nešto poput ovog:

```
{"message": "success", "number": 3, "people": [{"craft": "ISS", "name": "Yuri Malenchenko"}, {"craft": "ISS", "name": "Timothy Kopra"}, {"craft": "ISS", "name": "Timothy Peake"}]}
```

Ovo je Pythonov rječnik koji sadrži 3 ključa: poruku (message), broj (number) i osobe (people).

Vrijednost poruke (message) je ‘success’ što znači da je zahtjev bio uspješan. Odlično.

Imaj na umu da ćeš vidjeti različite rezultate, ovisno o tome tko je trenutno u

svemiru!

- Ispišimo sada informacije u obliku koji će biti jednostavnije pročitati.



Za početak pronađimo broj osoba u svemiru i ispišimo ga:

```
url = 'http://api.open-notify.org/astros.json'
odgovor = urllib.request.urlopen(url)
rezultat = json.loads(odgovor.read())

print('Broj ljudi u svemiru: ', rezultat['number'])
```

rezultat['number'] će u rječnik rezultata ispisati vrijednost povezana sa ključem 'number' (broj). U primjeru je to 3.

- Vrijednost povezana sa ključem 'people' je popis rječnika!



Spremimo tu vrijednost u varijablu kako bi ju mogao koristiti:

```
print('Broj ljudi u svemiru: ', rezultat['number'])

osobe = rezultat['people']
print(osobe)
```

Trebao bi vidjeti nešto poput ovog:

```
[{'craft': 'ISS', 'name': 'Yuri Malenchenko'}, {'craft': 'ISS', 'name': 'Timothy Kopra'}, {'craft': 'ISS', 'name': 'Timothy Peake'}]
```

- Sada moraš ispisati liniju za svakog astronauta.



U Pythonu za to možeš koristiti petlju. Svakim prolazom petlje, p će se spremiti u rječnik za različitog astronauta.

```
print('Broj ljudi u svemiru: ', rezultat['number'])

osobe = rezultat['people']

for p in osobe:
    print(p)
```

- Zatim možeš pogledati vrijednosti za 'name' i 'craft'.

```
print('Broj ljudi u svemiru: ', rezultat['number'])

osobe = rezultat['people']

for p in osobe:
    print(p['name'])
```

Trebao bi vidjeti nešto poput ovog:

```
Broj ljudi u svemiru: 3 Yuri Malenchenko Timothy Kopra Timothy Peake
```

Koristiš podatke koji se dobivaju uživo pa će tvoji rezultati ovisiti o broju ljudi koji su trenutno u svemiru.



Spremi projekt

## Izazov: Prikaži gdje je letjelica

Web uslugom se, uz ime astronauta, može vidjeti i u kojem se objektu taj astronaut nalazi (kao, primjerice, ISS).

Izmijeni svoj kôd tako da ispisuje i objekt u kojem je astronaut.

Primjerice:

```
Broj ljudi u svemiru: 3
Yuri Malenchenko in ISS
Timothy Kopra in ISS
Timothy Peake in ISS
```



Spremi projekt

## Korak 2: Gdje se nalazi Međunarodna svemirska postaja?

Međunarodna svemirska postaja nalazi se u Zemljinoj putanji. Napravi krug oko Zemlja otprilike svakih sat i 30 minuta, a prosječna brzina joj je 7.66 km po sekundi. Brza je!

Pokušajmo pomoću još jedne web usluge saznati gdje se nalazi Međunarodna svemirska postaja.

### Zadataci

- Prvo otvori url web usluge u novoj kartici svog internet preglednika: <http://api.open-notify.org/iss-now.json>



Trebao bi vidjeti nešto poput ovog:

```
{ "iss_position": { "latitude": 8.54938193505081, "longitude":  
73.16560793639105 }, "message": "success", "timestamp": 1461931913 }
```

Rezultat sadrži koordinate mjesta na Zemlji iznad kojeg se trenutno nalazi Međunarodna svemirska postaja.

Geografska dužina kreće se u pravcu istoka ili zapada, od -180 do 180 stupnjeva. 0 je Početni meridijan koji prolazi kroz Greenwich u Londonu, u Velikoj Britaniji.

Geografska širina kreće se u pravcu sjevera ili juga, od 90 do -90 stupnjeva. 0 je Ekvator.

- Sada u Pythonu moraš pozvati istu web uslugu. Dodaj sljedeći kôd na kraju programa kako bi dobio trenutnu lokaciju Međunarodne svemirske postaje:



```
url = 'http://api.open-notify.org/iss-now.json'
odgovor = urllib.request.urlopen(url)
rezultat = json.loads(odgovor.read())
print(rezultat)
```

```
{'message': 'success',
'timestamp': 1511391450,
'iss_position': {'latitude': '-3.9039', 'longitude': '103.4420'}}
```

- Definirat čemo varijable za spremanje geografske dužine i širine, a zatim ih ispisati:

```
url = 'http://api.open-notify.org/iss-now.json'
odgovor = urllib.request.urlopen(url)
rezultat = json.loads(odgovor.read())

lokacija = rezultat['iss_position']
sirina = lokacija['latitude']
duzina = lokacija['longitude']
print('Geografska širina:', sirina)
print('Geografska dužina:', duzina)
```

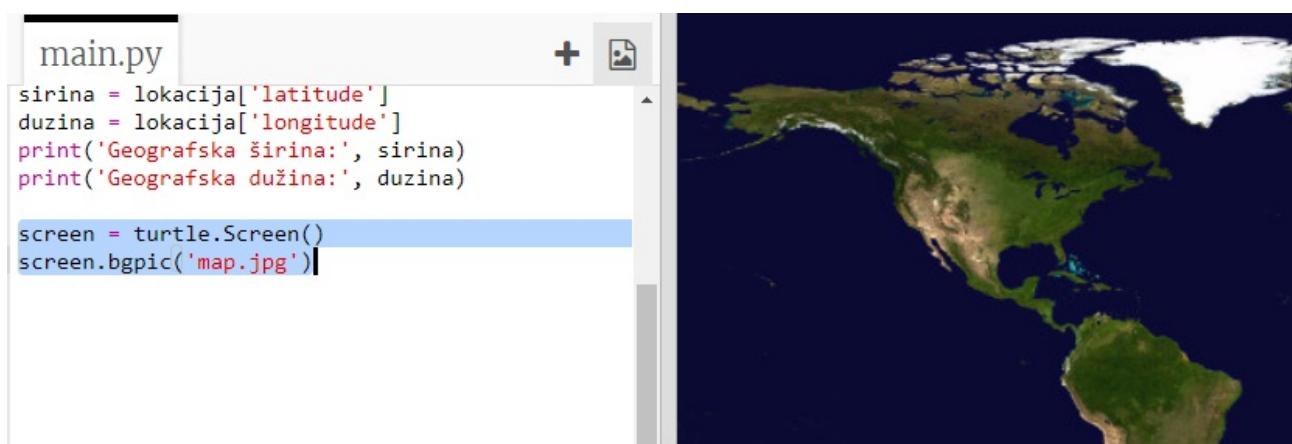
```
Geografska širina: -20.4852
Geografska dužina: 116.1481
```

- Bilo bi korisnije kada bi prikazali lokaciju na karti.

Prvo moramo uvesti turtle biblioteku za slikovni prikaz podataka.

```
import json
import urllib.request
import turtle
```

- Učitajmo mapu svijeta kao pozadinsku sliku. Jedna je već uključena u tvoj trinket.



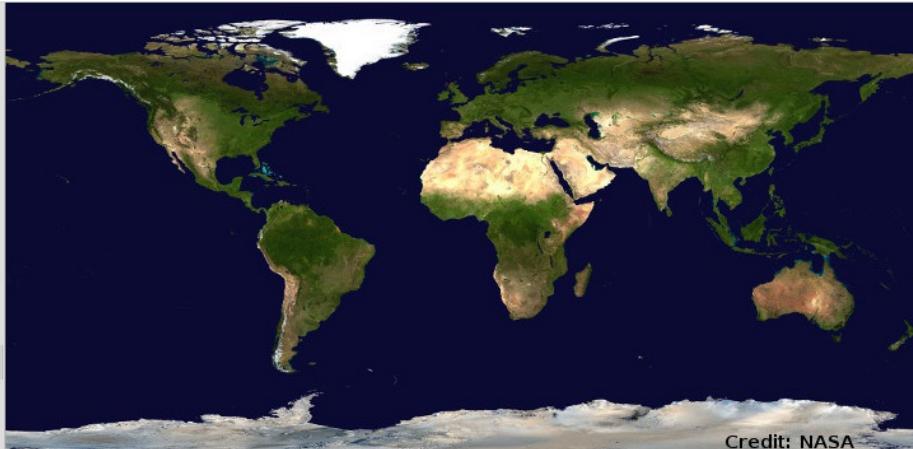
NASA je omogućila ovu divnu mapu i dala dopuštenje za njeno korištenje.

Mapa je centrirana na koordinatama (0, 0), a to je upravo ono što ti treba.

- Moraš podesiti veličinu ekrana na 720x360 tako da odgovara veličini slike.



Dodaj `screen.setup(720, 360)`:



```
main.py
screen = turtle.Screen()
screen.setup(720, 360)
screen.bgpic('map.jpg')
```

- Želiš omogućiti ‘kornjači’ da se pomiče određenim geografskim dužinama i širinama. Da bi ti bilo lakše, podesi ekran tako da odgovara koordinatama koje koristimo:



```
screen = turtle.Screen()
screen.setup(720, 360)
screen.setworldcoordinates(-180, -90, 180, 90)
screen.bgpic('map.jpg')
```

Sada će koordinate odgovarati geografskoj dužini i širini koordinata koje dobijemo sa web usluge.

- Napravimo ‘kornjaču’ za Međunarodnu svemirsku postaju.



```
screen = turtle.Screen()
screen.setup(720, 360)
screen.setworldcoordinates(-180, -90, 180, 90)
screen.bgpic('map.jpg')

screen.register_shape('iss.png')
iss = turtle.Turtle()
iss.shape('iss.png')
iss.setheading(90)
```



Tvoj projekt sadrži slike ‘iss.png’ i ‘iss2.png’. Isprobaj ih obje i odluči koja ti se više sviđa.

- Međunarodna svemirska postaja nalazi se u središtu mape.



Pomaknimo ju na točnu lokaciju na mapi:

```
screen.register_shape('iss.png')
iss = turtle.Turtle()
iss.shape('iss.png')
iss.setheading(90)

iss.penup()
iss.goto(lon, lat)
```

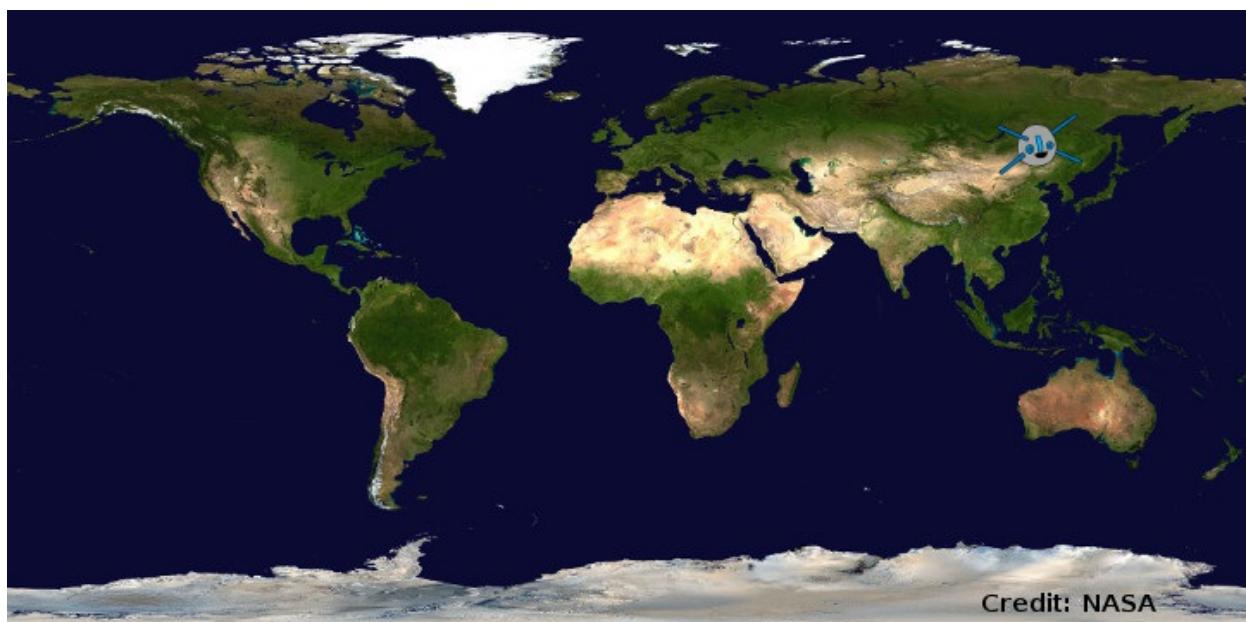
Primijeti da se širina obično prikazuje prva, ali mi moramo prvo upisati dužinu kada crtamo (x,y) koordinate.

- Pokreni i testiraj program.



Međunarodna svemirska postaja trebala bi se pomaknuti na svoju trenutnu lokaciju iznad Zemlje.

Pričekaj nekoliko sekundi, a zatim opet pokreni program da vidiš gdje se Međunarodna svemirska postaja pomaknula.



Latitude: 51.1757114507  
Longitude: 119.515729267



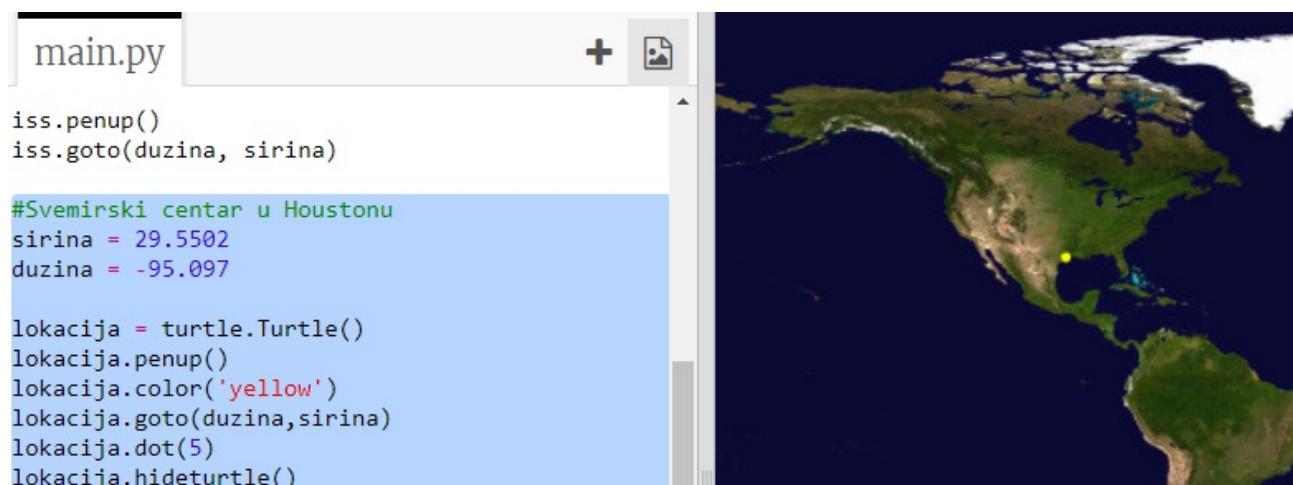
## Korak 3: Kada će Međunarodna svemirska postaja biti iznad nas?

Postoji i web usluga preko koje možeš saznati kada će Međunarodna svemirska postaja biti na određenoj lokaciji.

Saznajmo kada će Međunarodna svemirska postaja biti iznad Svemirskog centra u Houstonu, u SAD-u. Geografska širina Centra je 29.5502, a dužina 95.097.

### Zadataci

- Ucrtajmo prvo točku na mapi na sljedećim koordinatama:



- Saznajmo sada datum i vrijeme kada će se Međunarodna svemirska postaja nalaziti iznad Svemirskog centra.

Kao i prije, web usluzi možemo pristupiti unošenjem url-a u adresnu traku web preglednika: <http://api.open-notify.org/iss-pass.json>

Trebao bi dobiti poruku o grešci:

```
{
  "message": "failure",
  "reason": "Latitude must be specified"
}
```

- Za pristup podatcima preko ove web usluge moramo uključiti

geografsku dužinu i širinu u url koji koristimo.

Ti podatci dodaju se nakon znaka `?` i odvajaju se znakom `&`.

Url-u dodaj `lat` i `lon` podatke kao što je prikazano ovdje: <http://api.open-notify.org/iss-pass.json?lat=29.55&lon=95.1>

```
{  
    "message": "success",  
    "request": {  
        "altitude": 100,  
        "datetime": 1465541028,  
        "latitude": 29.55,  
        "longitude": 95.1,  
        "passes": 5  
    },  
    "response": [  
        {  
            "duration": 630,  
            "risetime": 1465545197  
        },  
        {  
            "duration": 545,  
            "risetime": 1465551037  
        },  
        {  
            "duration": 382,  
            "risetime": 1465568806  
        },  
        {  
            "duration": 625,  
            "risetime": 1465574518  
        }  
    ]  
}
```

U odgovoru se nalazi nekoliko razdoblja kada će Međunarodna svemirska postaja biti iznad Svemirskog centra, ali mi ćemo se usredotočiti samo na prvi termin. Vrijeme je zapisano u standardnom obliku, ali moći ćeš ga u Pythonu pretvoriti u oblik koji se lako čita.

- Pozovimo sada web uslugu u Pythonu. Dodaj sljedeći kôd na kraju svog programa:



```
url = 'http://api.open-notify.org/iss-pass.json'  
url = url + '?lat=' + str(sirina) + '&lon=' + str(duzina)  
odgovor = urllib.request.urlopen(url)  
rezultat = json.loads(odgovor.read())  
print(rezultat)
```

```
{'message': 'success', 'request': {'latitude': 29.5502, 'longitude': -95.097, 'altitude': 100, 'datetime': 1511392512, 'passes': 5}, 'response': [{"duration": 130, 'risetime': 1511394455}, {"duration": 633, 'risetime': 1511399952}, {"duration": 543, 'risetime': 1511405797}, {"duration": 385, 'risetime': 1511423568}, {"duration": 624, 'risetime': 1511429281}]}}
```

- Upotrijebimo sada prvi termin koji smo dobili u odgovoru sa web usluge.



Dodaj sljedeći kôd:

```
url = 'http://api.open-notify.org/iss-pass.json'  
url = url + '?lat=' + str(sirina) + '&lon=' + str(duzina)  
odgovor = urllib.request.urlopen(url)  
rezultat = json.loads(odgovor.read())  
  
iznad = rezultat['response'][1]['risetime']  
print(iznad)
```

Powered by  trinket  
1511399952

Vrijeme prelaska u standardnom formatu

- Vrijeme je zapisano kao vremenska oznaka pa ćemo koristiti Pythonov modul za vrijeme kako bi ga ispisali u čitljivijem formatu i pretvorili u lokalno vrijeme. Neka ‘kornjača’ upiše vrijeme pokraj točke na mapi.
- Dodaj liniju `import time` na početku programa:



```
import json  
import urllib.request  
import turtle  
import time
```

- Funkcija `time.ctime()` pretvorit će vrijeme u čitljiviji format koji možeš zapisati sa ‘kornjačom’.



```
main.py  
  
iznad = rezultat['response'][1]['risetime']  
#print(iznad)  
  
style = ('Arial', 6, 'bold')  
lokacija.write(time.ctime(iznad), font=style)
```



(Možeš ukloniti ili komentirati `print` liniju.)



Spremi projekt

## Izazov: Pronađi još razdoblja kada će Međunarodna svemirska postaja prelaziti iznad nekog mjesto

Možeš koristiti internet stranicu poput <http://wwwlatlong.net/> za pronalaženje geografske širine i dužine mesta koja te zanimaju.

Možeš li pronaći i ucrtati razdoblja kada će Međunarodna svemirska postaja biti iznad još nekih mesta?

- Trebaš promijeniti geografsku širinu i dužinu u url-u web usluge.
- Trebaš ucrtati lokaciju i rezultat na mapi.



Spremi projekt