

# Instaliranje Python-a na Windows

Python for Windows download page



python™

Menu

Python >>> Downloads >>> Windows

## Python Releases for Windows

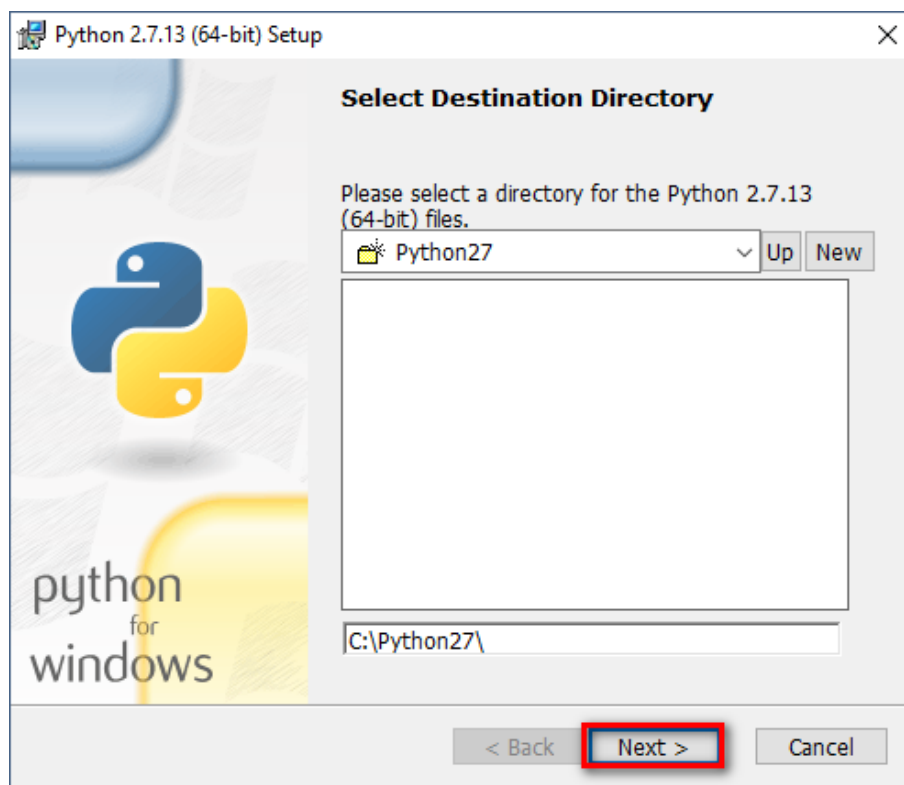
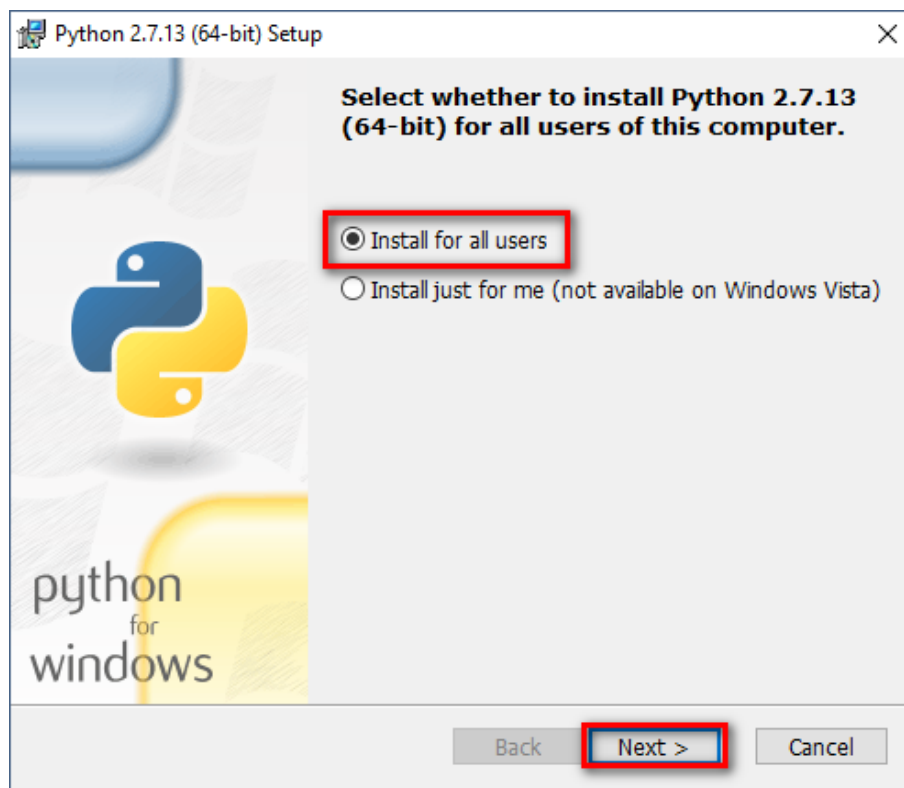
- [Latest Python 2 Release - Python 2.7.13](#)
- [Latest Python 3 Release - Python 3.6.1](#)

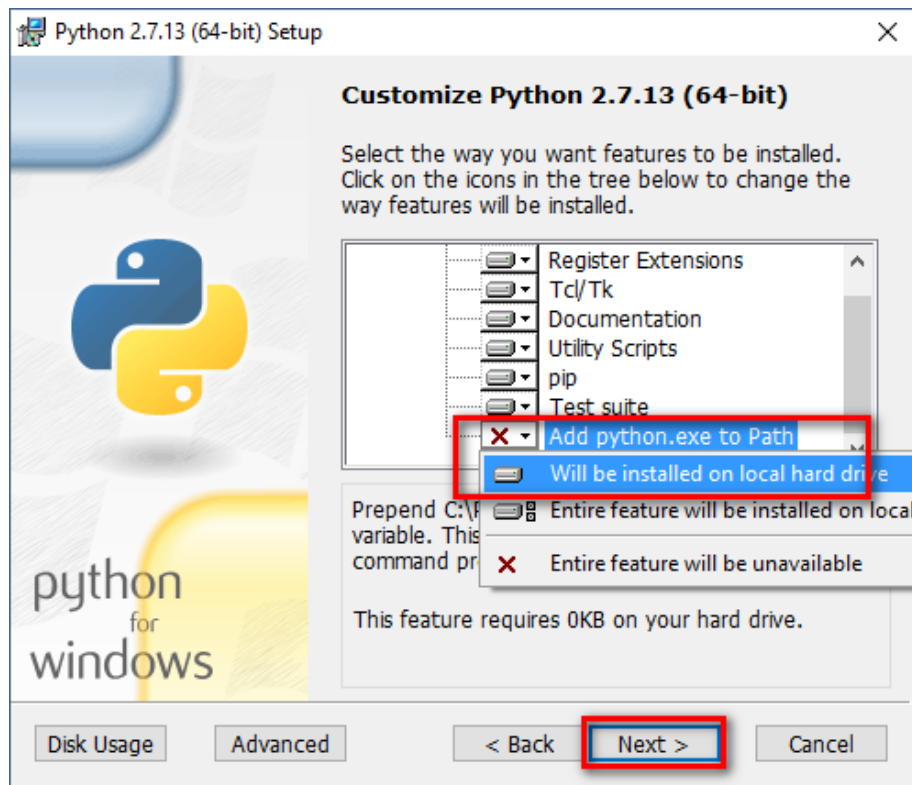
Razlike između verzija 2 i 3.

- [Python 3.6.1 - 2017-03-21](#)
  - Download [Windows x86 web-based installer](#)
  - Download [Windows x86 executable installer](#)
  - Download [Windows x86 embeddable zip file](#)
  - Download [Windows x86-64 web-based installer](#)
  - Download [Windows x86-64 executable installer](#)
  - Download [Windows x86-64 embeddable zip file](#)
  - Download [Windows help file](#)

Razlike između 32 i 64 bita

## Instalacija Python 2

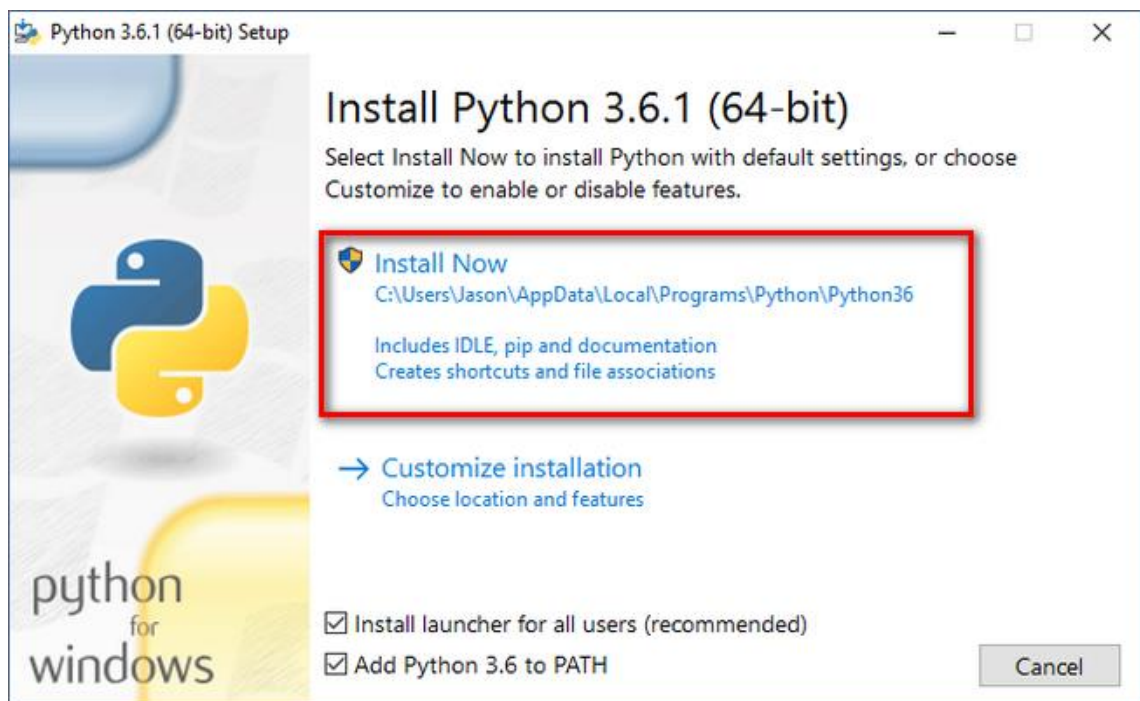




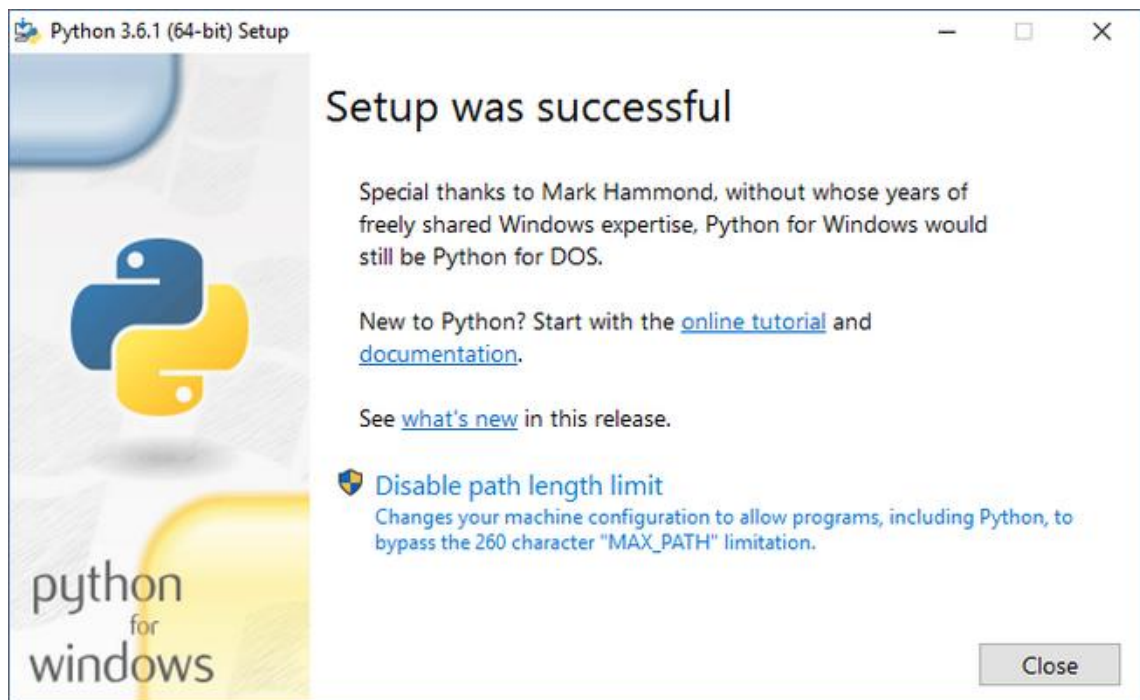
Provera:

```
python -V
```

## Instalacija Python 3



Kako preko 260 max.char u PATH-u na Windows 10.

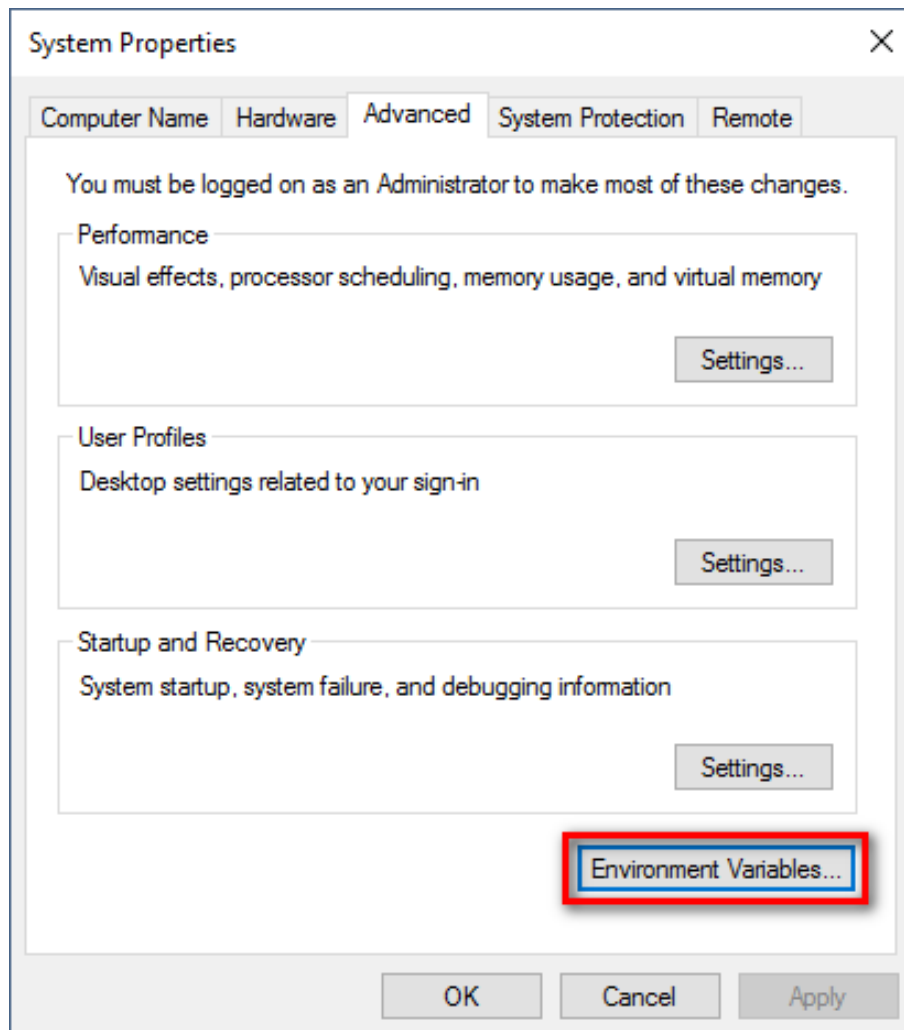


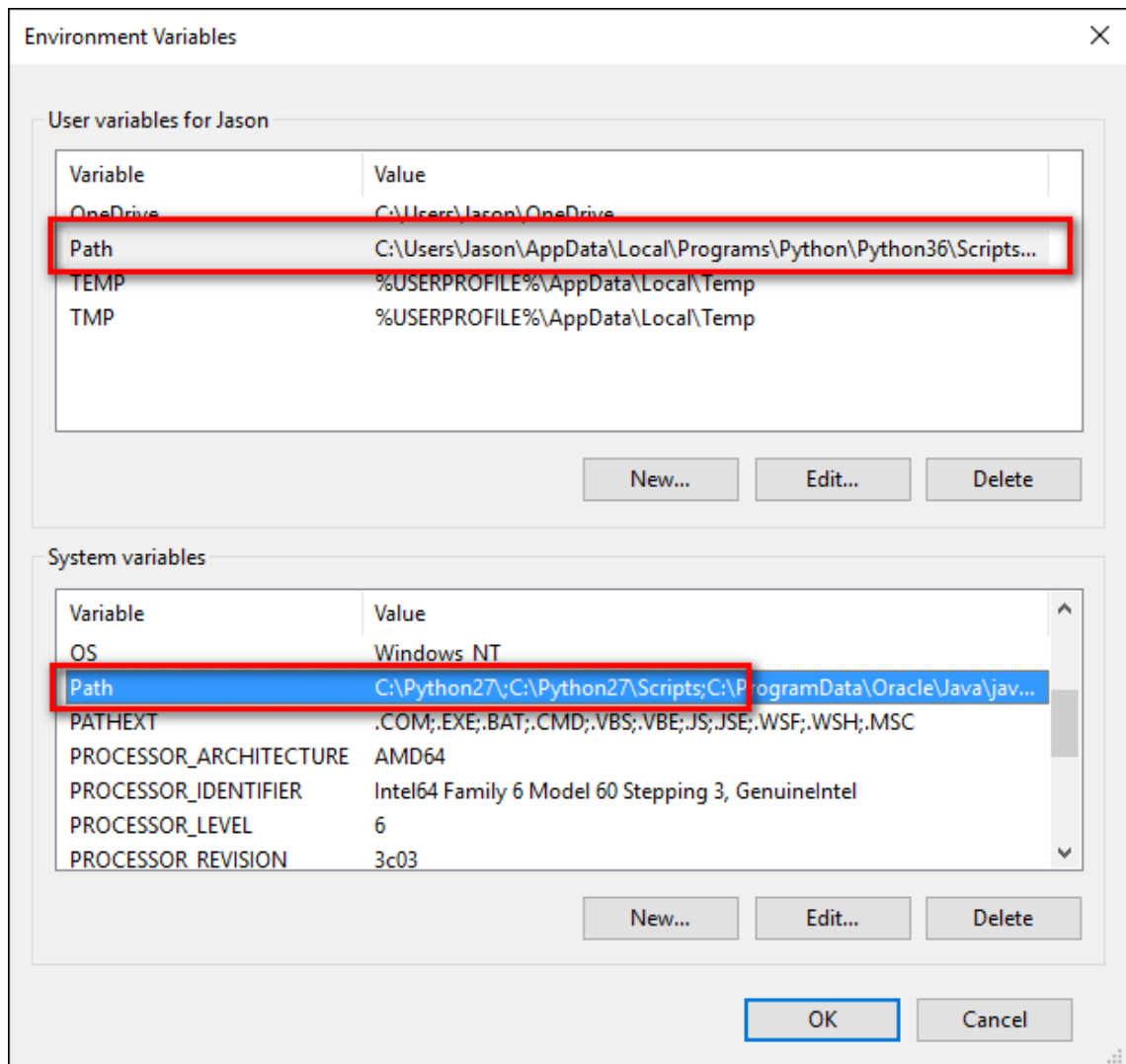
## Podešavanje sistemskih varijabli

Unesite u Search Windows

ENV

i izaberite podešavanje sistemskih varijabli okruženja.





# Virtuelna okolina za python

Kreiramo direktorijum projekta, i pozicioniramo se unutra:

```
mkdir ime_projekta  
cd ime_projekta
```

Kreiramo virtuelnu okolinu sa (slučaj 2 i više interpretera instaliranih na mašini):

```
virtualenv <naziv_dir_po_želji> -p <path_do_željenog_python_interpretera>
```

Ako je samo jedan interpreter instaliran, standardno u direktorijumu projekata:

```
virtualenv venv
```

jer je za Python2 standardan instalacioni direktorijum:

```
c:\Python27
```

a za Python 3:

```
c:\Python3x
```

Virtuelnu okolinu aktiviramo iz direktorijuma projekta sa:

```
venv\Scripts\activate.bat
```

Prompt će se promeniti, biće prefiksovan sa:

```
(venv)
```

Virtuelnu okolinu deaktiviramo sa

```
deactivate
```

# Python: pip Tutorial

Pip je program komadne linije koji se koristi za upravljanje Python paketima. Kućna strana je :

```
https://pypi.python.org/pypi/pip
```

Osnovne komande:

Prikazivanje pomoći:

```
(venv) pip help
```

Usage:

```
pip <command> [options]
```

Commands:

install	Install packages.
uninstall	Uninstall packages.
freeze	Output installed packages in requirements format.
list	List installed packages.
show	Show information about installed packages.
search	Search PyPI for packages.
wheel	Build wheels from your requirements.
zip	DEPRECATED. Zip individual packages.
unzip	DEPRECATED. Unzip individual packages.
bundle	DEPRECATED. Create pybundles.
help	Show help for commands.

General Options:

-h, --help	Show help.
-v, --verbose	Give more output. Option is additive, and can be used up to 3 times.
-V, --version	Show version and exit.
-q, --quiet	Give less output.
--log-file <path>	Path to a verbose non-appending log, that only logs failures. This log is active by default at /Users/outbound/Library/Logs/pip.log.
--log <path>	Path to a verbose appending log. This log is inactive by default.
--proxy <proxy>	Specify a proxy in the form [user:passwd@]proxy.server:port.
--timeout <sec>	Set the socket timeout (default 15 seconds).
--exists-action <action>	Default action when a path already exists: (s)witch, (i)gnore, (w)ipe, (b)ackup.
--cert <path>	Path to alternate CA bundle.

Instalacija jednog paketa:

```
(venv) pip install <package name>==1.0
```

Prikazivanje šta je i gde instalirano za dati paket:

```
(venv) pip show --files <package name>
```

```
Name: <package name>
```



```
Version: 1.0
Location: /my/env/lib/pythonx.x/site-packages
Files:
  ../somepackage/__init__.py
[...]
```

Prikaz paketa koje treba osvežiti:

```
(venv) pip list --outdated
<package name> (Current: 1.0 Latest: 2.0)
```

Osvežavanje paketa:

```
(venv) pip install --upgrade <package name>
[...]
Found existing installation: <package name> 1.0
Uninstalling <package name>:
  Successfully uninstalled <package name>
Running setup.py install for <package name>
Successfully installed <package name>
```

Uklanjanje jednog paketa:

```
(venv) pip uninstall <package name>
Uninstalling <package name>:
  /my/env/lib/pythonx.x/site-packages/somepackage
Proceed (y/n)? y
Successfully uninstalled <package name>
```

## Instalacija Jam.py

Sa pip-om:

```
(venv) pip install jam.py
```

Klasično: Prenesi lokalno sa githuba :

```
https://github.com/jam-py/jam-py
```

Ili kloniraj

```
git clone https://github.com/jam-py/jam-py
```

Idi u master direktorijum a zatim:

```
(venv) python setup.py install
```

## Instalacija drajvera za PostgreSQL

```
(venv) pip install psycopg2
```

## Kreiranje projekta

Izaberite direktijum (kreirajte) gde će projekat/podprojekat biti.

```
ime_projekta/  
    ime_podprojekta/  
        venv/
```

Pozicionirajte se unutra. Na windows mašini pokrenite:

```
(venv) python venv\Scripts\jam-project.py
```

Biće kreirano stablo sa pratećim subdirektorijumima i fajlovima:

```
/  
    css/  
        project.css  
    js/  
    img/  
    reports/  
    static/  
    admin.sqlite  
    server.py  
    index.html
```

Sada sa

```
(venv) python server.py
```

pokrenite web server. Dobićete poruku da možete pronaći vaš upravo kreirani projekt na :

```
http://127.0.0.1:8080/
```

Mogli ste da izaberete neki nestandarni port u prethodnoj komandi:

```
(venv) python server.py 8081
```

Sada otvorite browser na:

```
http://127.0.0.1:8080/builder.html
```

i počnite sa izgradnjom aplikacije.