

# How to set up python development env

*Hieu Nguyen*

*2/22/2019*

## Contents

<b>1</b>	<b>Envs Set up procedures for mac:</b>	<b>2</b>
<b>2</b>	<b>Environment Variables</b>	<b>2</b>
<b>3</b>	<b>Use Python within Rmarkdown</b>	<b>3</b>

## 1 Envs Set up procedures for mac:

- *conda create -name my\_env python=3.x*
- *source activate*
- *source deactivate*
- To view installed packages: *conda list*
- Search outdated packages: *conda search -outdated*
- To update packages: *conda update -all*
- Update the outdated packages: *conda install -U*
- To view which python: *which python*
- To view list of envs: *conda env list*
- To remove an env: *conda remove -name my\_env -all*
- To create a new dir: *mkdir my\_project\_env*
- To Export an env to a YAML file: *conda env > env.yaml*
- display the yaml file on the terminal: *cat env.yaml*
- To create a new env out of the above yaml file: *conda create -f env.yaml*. Make sure that the yaml must be inside the newly create directory

## 2 Environment Variables

used in many different projects to set data base url, hold secret keys or set python path. It allows us to access the data without putting it directly on our project codes for everyone to see. Some people just use global env var that can be shared among all projects BUT you have to be very careful not to overwrite them. It is better to separate them out

- After create a new env, do *conda env list* and *cd env's path*
- Create an activate directory (-p means the dir does not have to curently exist):  
*mkdir -p etc/conda/activate.d*
- Create an deactivate directory (-p means the dir does not have to curently exist):  
*mkdir -p etc/conda/deactivate.d*
- In the activate dir, create a file:  
*touch etc/conda/activate.d/env\_vars.sh*
- In the deactivate dir, create a file:  
*touch etc/conda/deactivate.d/env\_vars.sh*
- In terminal, use sublime text editor to open up the newly created file:  
*subl .*
- In sublime, for the activate/env\_vars.sh:  
*#!/bin/sh*  
*export DATABASE\_URI="postgresql://user:pass@db\_server:5432/test\_db"* This way no one can see the pw

- In sublime, for the deactivate/env\_vars.sh:

```
#!/bin/sh
```

```
*unset DATABASE_URI
```

- Test: try to deactivate and activate the env

Once activated: `echo DATABASE_URI` should show “postgresql://user:pass@db\_server:5432/test\_db”

Once deactivated: `echo DATABASE_URI` should show NOTHING

- To look at all the files in a dir: `ls -la`
- **DO NOT FORGET TO DEACTIVATE ENV WHEN MOVING FROM ONE PROJECT TO ANOTHER**

### 3 Use Python within Rmarkdown

```
library(reticulate)
```

```
# engine.path = '/Users/hq2nguye/anaconda3/bin/python' --> Mac
```

```
import sys
```

```
print("Hello World")
```

```
## Hello World
```

```
print(sys.version_info)
```

```
## sys.version_info(major=2, minor=7, micro=10, releaselevel='final', serial=0)
```