Summer Internship 2013

MOOC PROTOTYPE

EDX-PLATFORM INSTALLATION

ABHINAV PURI

Task :

To integrate and run edx-platform in our systems.

Task Accomplished :

We were able to run edx studio following the below steps :

- **Downloading and beginning** :
    - Download the repository from:
      https://github.com/edx/edx-platform
    - Make a directory (preferably in /home/<username>/ directory) naming 'edx_all' (used in this documentation (one can choose its own name)).
    - Extract the repository tar file in this directory.
      Let's change the name of the extracted folder to "edx-platform"
    - Create three folders in this (edx_all) directory and name them as :
      db
      data
      log
    - To check these folder's rights:
      type (in shell):
      ls -l
    - Give user rights to them(if not :
      chmod uog+rwx db
      chmod uog+rwx data
      chmod uog+rwx log
    - Now, in '/home/<username>/' directory 'edx_all' directory looks like :
      /db
      /data
      /log
      /edx-platform

Proxy people:

(Only for those who are working in proxy environment) :

While following the any of the below steps, if any of the errors like : "HTTP 403 authentication failled", or "connection failled:Cannot authenticate"... etc, occurs then it is an proxy authentication problem, to solve this :

Type in shell :

env | grep proxy

If output is :

http_proxy=http://<proxy>:<port no>

ftp_proxy=ftp://<proxy>:<port no>

ALL_PROXY=socks://<proxy>:<port no>

all_proxy=socks://<proxy>:<port no>

socks_proxy=socks://<proxy>:<port no>

https_proxy=https://<proxy>:<port no>

no_proxy=localhost,

then add authentication to our env :

Type in shell :

export http_proxy=http://<username>:<password>@<proxy>:<port no>

export https_proxy=https://<username>:<password>@<proxy>:<port no>

(Remember : password must not have characters like '@' or ':')

Assumption : Here , we make an assumption that one has python2.7 installed on its linux platform. (if not, type (in shell) : sudo apt-get install python2.7)

- **Virtual Environment** :
  - Edx platform assumes that we have set an virtual environment, where we will run our project. Lets do it by following typing in shell :

sudo apt-get install python-setuptools
sudo apt-get install python-virtualenv

sudo pip install virtualenvwrapper
sudo pip install virtualenv

- *Errors that may occur* :

1. First two commands should not have any problem, but if any problem like "Could not locate package" occurs, then follow the below solution.

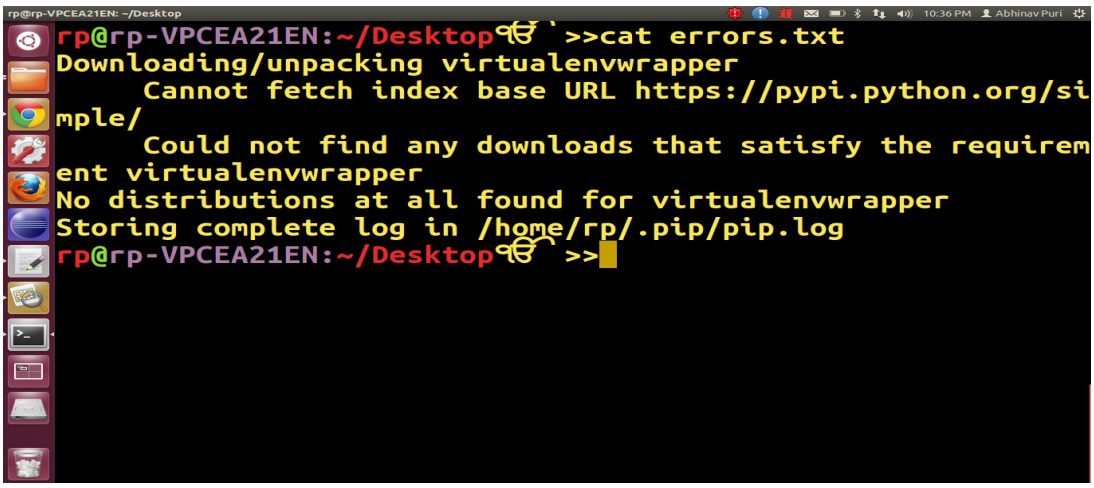
Solution :

If you have not set environment proxy ::
Refer to proxy people topic of this documentation.

If still problem comes, then ::
Sometimes sudo forget environment proxy ,
Type (in shell):
sudo -s
(This command makes you use shell as a root user)
Now, in this root session, again check environment proxy as in proxy people topic and do as given in that topic.

After doing as above,
Type :
apt-get install python-setuptools
apt-get install python-virtualenv
exit (to exit from root session)

2. Third and Fourth command may give you error like :



Solution :
Refer Remark #1 (pip issues).

3. Another type of error that may occur in third commmand :

```
No local packages or download links found for stevedore
    error : could not find suitable distribution for
            Requirement.parse('stevedore')
```

Solution:

Type in shell :

sudo pip install stevedore.

sudo pip install virtualenv

- **Using Virtual Environment** :

  - After previous steps, the following things must get automatically added to your system :

    /home/<username>/.virtualenvs

    /usr/local/bin/virtualenvwrapper.sh

  - Add them to ~/.bashrc as:

    Type (in shell) :

    sudo nano ~/.bashrc

    (this opens .bashrc file)

    Append following lines to this file and save them and exit:

    export WORKON_HOME=$HOME/.virtualenvs

    source /usr/local/bin/virtualenvwrapper.sh

    export PYTHON_DIR=$HOME/.virtualenvs

  - Now, close all the terminals, re-open them for changes to get affected.

    Type in shell:

    mkvirtualenv -a "$WORKON_HOME" edx-platform

- **Installing pre.txt** :

(*Remember* : One should never use sudo commands in virtualenvs)

- As soon as you used mkvirtualenv command, on successfull execution, you will be working on edx-platform virtual environment. To come out of it, Type : deactivate and to go back in it, Type : workon edx-platform.

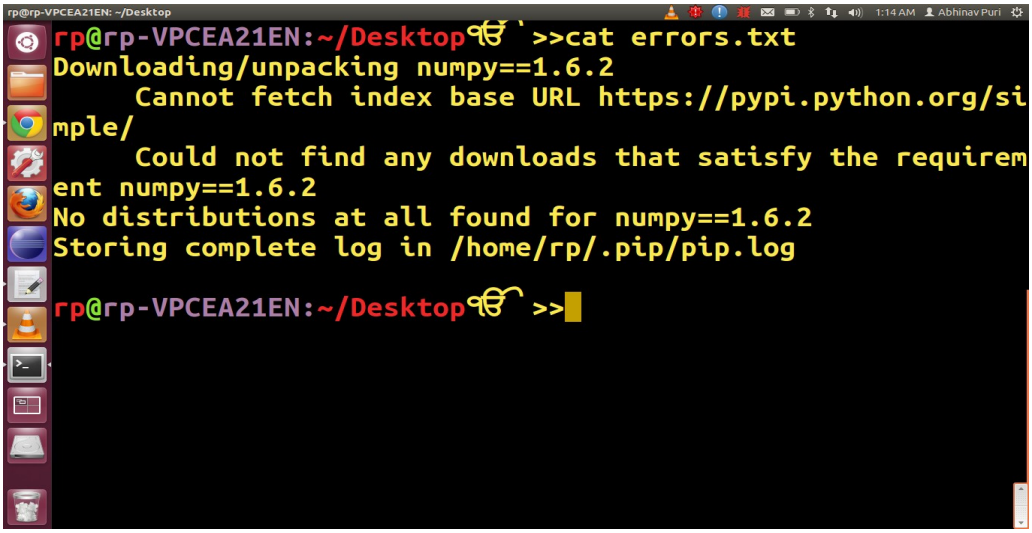
(*You will be in /.virtualenv directory when you are working in edx-platform virtual environment.*)

Type in shell :

workon edx-platform
cd
cd /home/<username>/edx_all/edx-platform
pip install -r requirements/edx/pre.txt
(this install numpy==1.6.2 in your system)

- *Errors that may occur* :

1. Numpy error :



Solution:

If you have not set environment proxy ::

Refer to proxy people topic of this documentation.

If still problem comes, then ::

Add proxy switch to pip, Type :

pip –proxy = http://<username> : <password> @ <proxy>: <port no> install -r requirements/edx/pre.txt

2. Error could be due to not using sudo (since, we can't use sudo in virtualenvs) similiar to :

```
dpkg: error: requested operation requires superuser privilege
W: Could not open file '/usr/lib/term.log' - OpenLog (13: Permission denied)
```

Solution:

In such a case, check out which file cannot be opened, as in above case /usr/lib/term.log can't be opened, for this case, Type :
chown <username>:<username> -R /usr/lib/
(this gives permission to whole directory to be edited).

Now, it will work :
pip --proxy = http://<username> : <password> @ <proxy>: <port no> install -r requirements/edx/pre.txt

- **Installing base.txt** :
  - Type in shell :
    workon edx-platform # Skip this if you are already in it.
    cd
    cd /home/<username>/edx_all/edx-platform
  - Make changes to requirements/edx/github.txt file :

Add '+https' for third party git repositories as shown below :

File looks like (before changes):
Third-party:
-e git://github.com/edx/django-staticfiles.git@6d2504e5c8#egg=django-staticfiles

-e git://github.com/edx/django-pipeline.git#egg=django-pipeline
-e git://github.com/edx/django-wiki.git@e2e84558#egg=django-wiki
-e git :// github.com / dementrock / pystache_custom.git @
776973740bdaad83a3b029f96e415a7d1e8bec2f#egg=pystache_custom-dev
-e git :// github.com / eventbrite / zendesk.git @
d53fe0e81b623f084e91776bcf6369f8b7b63879#egg=zendesk

After changes, file looks like :
Third-party:
-e git+https://github.com/edx/django-staticfiles.git@6d2504e5c8#egg=django-staticfiles
-e git+https://github.com/edx/django-pipeline.git#egg=django-pipeline
-e git+https://github.com/edx/django-wiki.git@e2e84558#egg=django-wiki
-e git+https :// github.com/ dementrock/ pystache_custom.git @
776973740bdaad83a3b029f96e415a7d1e8bec2f#egg=pystache_custom-dev
-e git+https :// github.com/ eventbrite/ zendesk.git @
d53fe0e81b623f084e91776bcf6369f8b7b63879#egg=zendesk

Now again type in shell :

pip install -r requirements/edx/base.txt
(this install many python packages & clone important git repositories in your system)

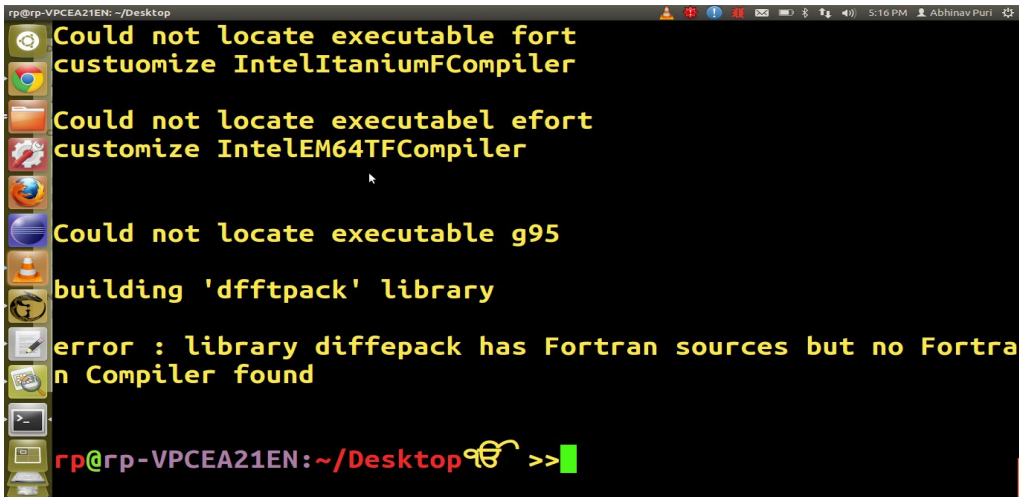
- *Errors that may occur* :

1. Cloning error ::

Solution :
Type in shell :

git config --global http.proxy
http://<username>:<password_internet>@<proxy>:<port no>
(Now, again try pip install -r requirements/edx/base.txt, it will work.)

2. gFortran error ::



Solution:

Type is shell :

apt-get install gfortran.

(donot use sudo, if permission denied issue comes, refer to installing pre.txt second error solved.)

Now, again go for : *pip install -r requirements/edx/base.txt* (it will work.)

3. lmxl error ::

```
checking for libxml libraries >= 2.6.27...
configure:
error:
Could not find libxml2 anywhere, check ftp://xmlsoft.org/.
```

Solution :

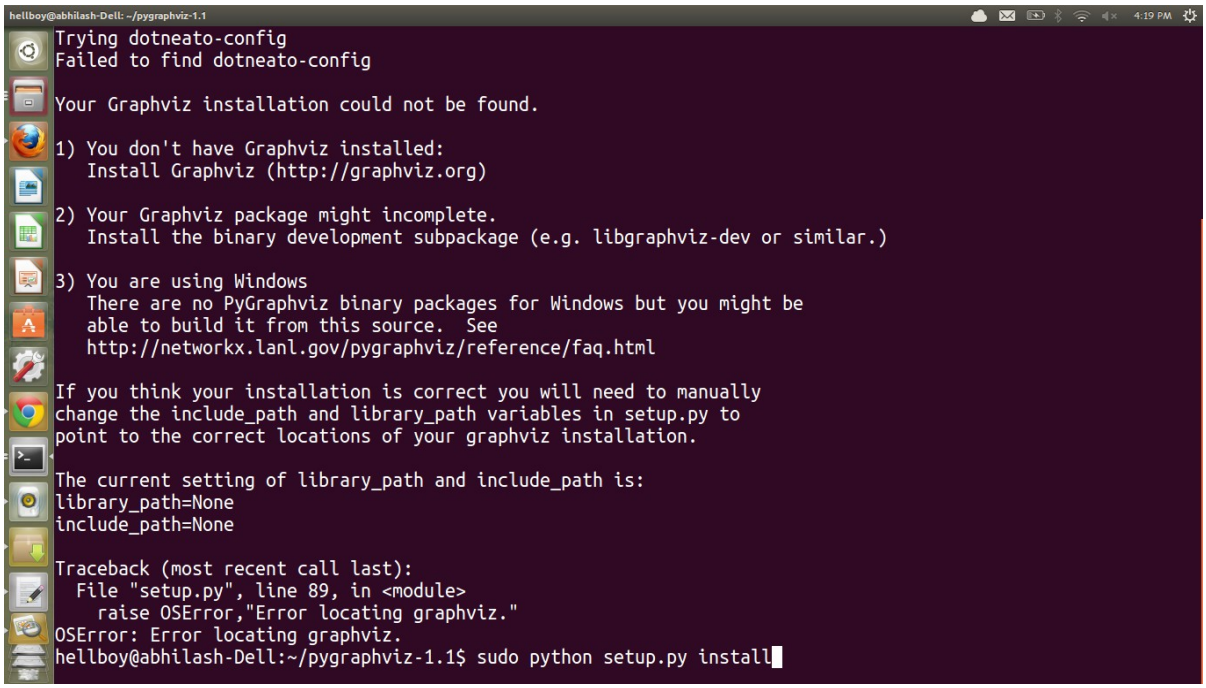
Type in shell :

apt-get install libxml2-dev libxslt-dev

(donot use sudo, if permission denied issue comes, refer to installing pre.txt second error solved.)

Now, again go for : *pip install -r requirements/edx/base.txt* (it will work.)

4. PyGraphviz error ::



Solution :

Type in shell :

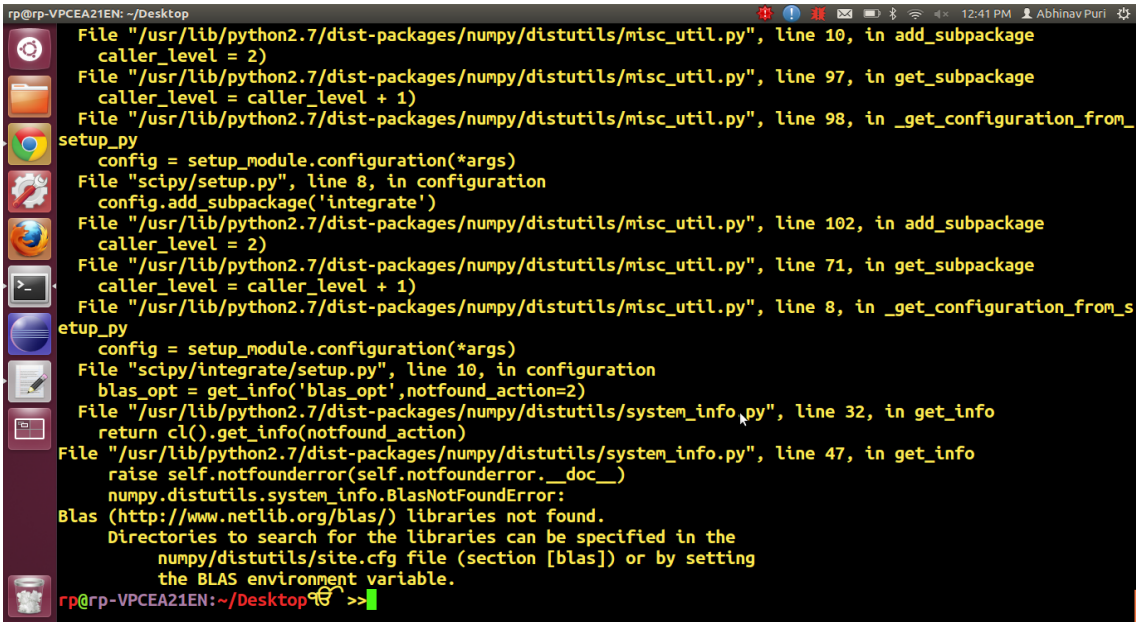
apt-get install libgraphviz-dev

apt-get install graphviz

(donot use sudo, if permission denied issue comes, refer to installing pre.txt second error solved.)

Now, again go for : *pip install -r requirements/edx/base.txt* (it will work.)

5. Scipy error ::



Solution :

(make some 'scr' folder (again 'src' is just a name used here)
mkdir -p ~/src/
cd ~/src/
wget http://www.netlib.org/blas/blas.tgz
tar xzf blas.tgz
cd BLAS
edit the make.inc file by setting OPTS = -O2 -fPIC and NOOPT = -O0 -fPIC
*ar r libfblas.a *.o*
ranlib libfblas.a
*rm -rf *.o*
export BLAS=~/src/BLAS/libfblas.a
(This solves BLAS issues)
dir -p ~/src
cd ~/src/
wget http://www.netlib.org/lapack/lapack.tgz
tar xzf lapack.tgzcd lapack-/*
cp INSTALL/make.inc.gfortran make.inc
make lapacklib
make clean
export LAPACK=~/src/LAPACK/libflapack.a
(This solves LAPACK issues)

Now, again go for : *pip install -r requirements/edx/base.txt* (it will work.)

6. Pip issues ::

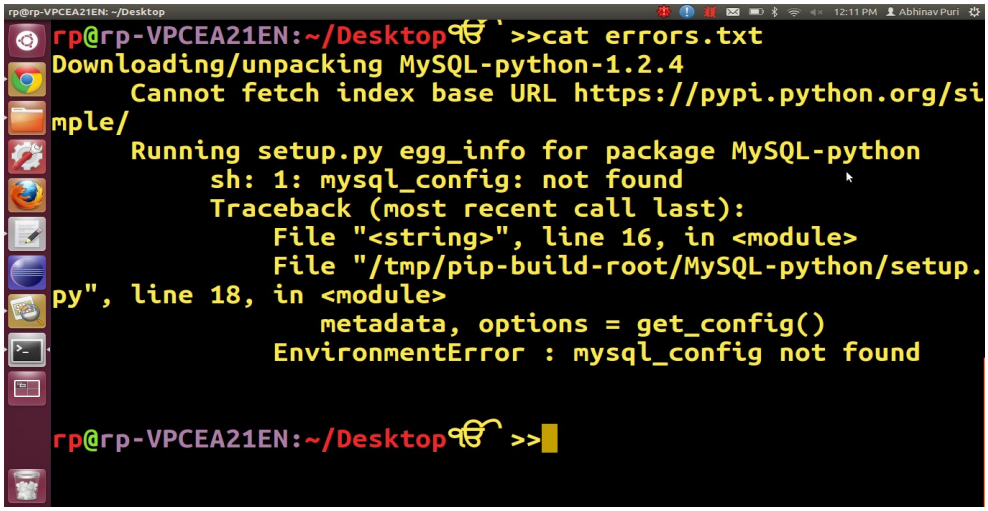
refer to Remark #1 (pip issues)

- **Installing post.txt** :

  - Type in shell :
    workon edx-platform # Skip this if you are already in it.
    cd
    cd /home/<username>/edx_all/edx-platform
    pip install -r requirements/edx/post.txt
    (this install MySQL-python==1.2.4 in your system)

- *Errors that may occur* :

  1. MySQL- python error ::



  Solution :
  Type :
  sudo apt-get install libmysqlclient-dev
  (Now, again run *pip install -r requirements/edx/post.txt*, it will work.)

- **The Bundler** :

  - Type in shell :
    workon edx-platform # Skip this if you are already in it.
    cd
    cd /home/<username>/edx_all/edx-platform
  - Now, ruby installation is required :
    apt-get install ruby1.9.3
    apt-get install rails
    (any error in this installation, refer to 2nd error of pre.txt)
    gem install bundler
    bundle install
    (this install gems from Gemfiles in your system)

- *Errors that may occur* :

1. gem installation error ::

```
GET http://rubygems.org/
                    403 Forbidden
                      or
 bad response Proxy Authentication Required 407  Forbidden
```

Solution :

Type in shell:

export HTTP_PROXY=http://user:pass@proxy:port

gem install --http-proxy=$HTTP_PROXY <gem name>

2. Bundle installation error ::

```
ruby: bad interpreter : No such file or directory
```

Solution :

Type in shell :

sudo nano ~/.bashrc

Append following lines :

export PATH="$PATH:/usr/bin:/usr/local/bin/"

(Now, again try bundle install, it will work.)

- **The NPM** :

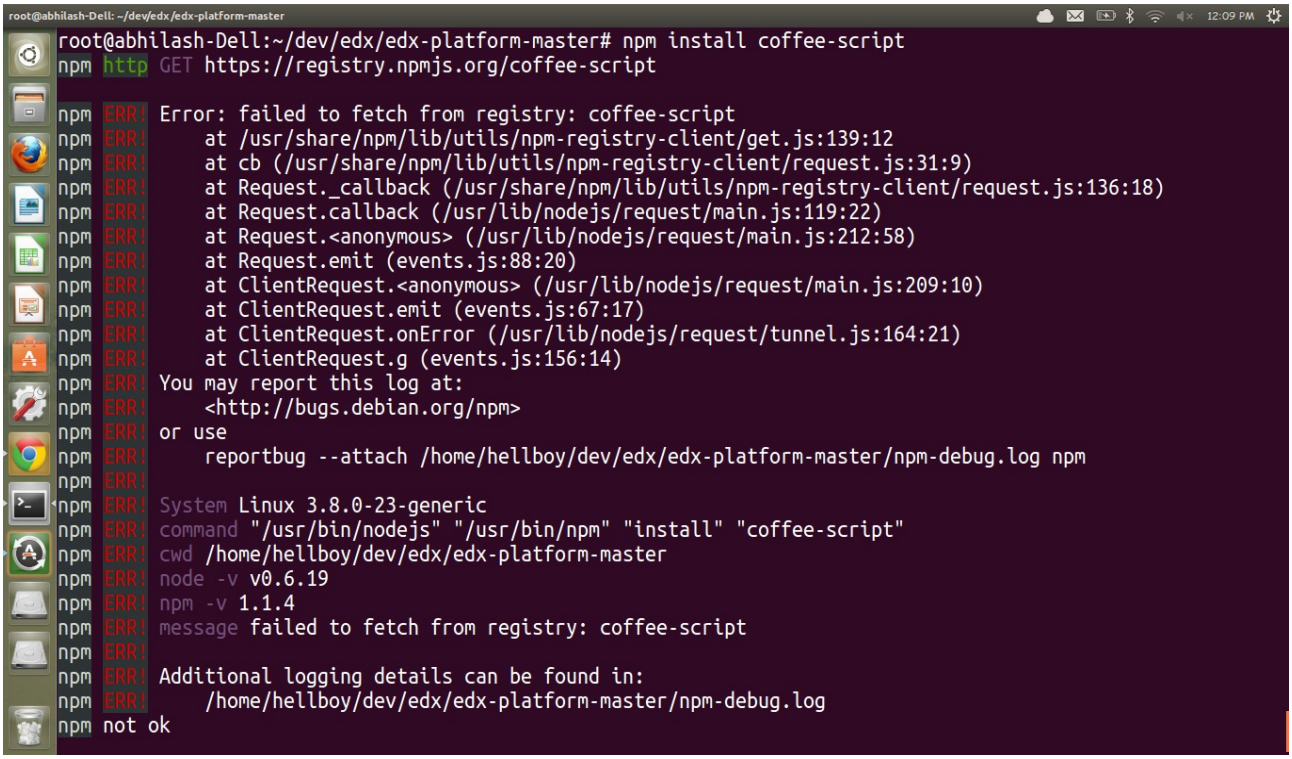
  - Type in shell :

```
workon edx-platform      # Skip this if you are already in it.
cd
cd /home/<username>/edx_all/edx-platform
npm install
```

(this install from package.json file in your system)

- *Errors that may occur* :

1. NPM error :



Solution :
Type in shell :
npm config set strict-ssl false
npm config set registry "http://registry.npmjs.org/"
npm --proxy http://<username>:<password>@<porxy>:<port no> install

- **Mongo DB** :

- Download debian package for mongodb for its official site :
http://www.mongodb.org/downloads

- Type following command :
sudo mkdir -p /data/db
(this gives the database space for mongodb)

- In dashboard, search for the Startup applications.
Open this application, press add and locate the mongod executable file in the bin folder of the extracted mongodb tar file.

(The above step opens mongod and creates the database server each time one opens ubuntu.)

- check for the running server :
  open in your browser :
  http://localhost:27017
  output :

" You are trying to access MongoDB on the native driver port. For http diagnostic access, add 1000 to the port number "

- **<u>Running the script</u>** :

  - Since, we have already install prerequisites for our edx-platform, we can skip running some parts of script 'create-dev-env.sh' present in 'edx-platform/scripts/create-dev-env.sh'.
    There is 100th line in this script :
    "### START
    ..."
  - Just comment (or delete) lines starting from this 100th line till line 447th which is :
    "# building correct version of distribute from source
    ..."

  - Also comment (or delete) lines starting from this 482th line till line 496th which are :

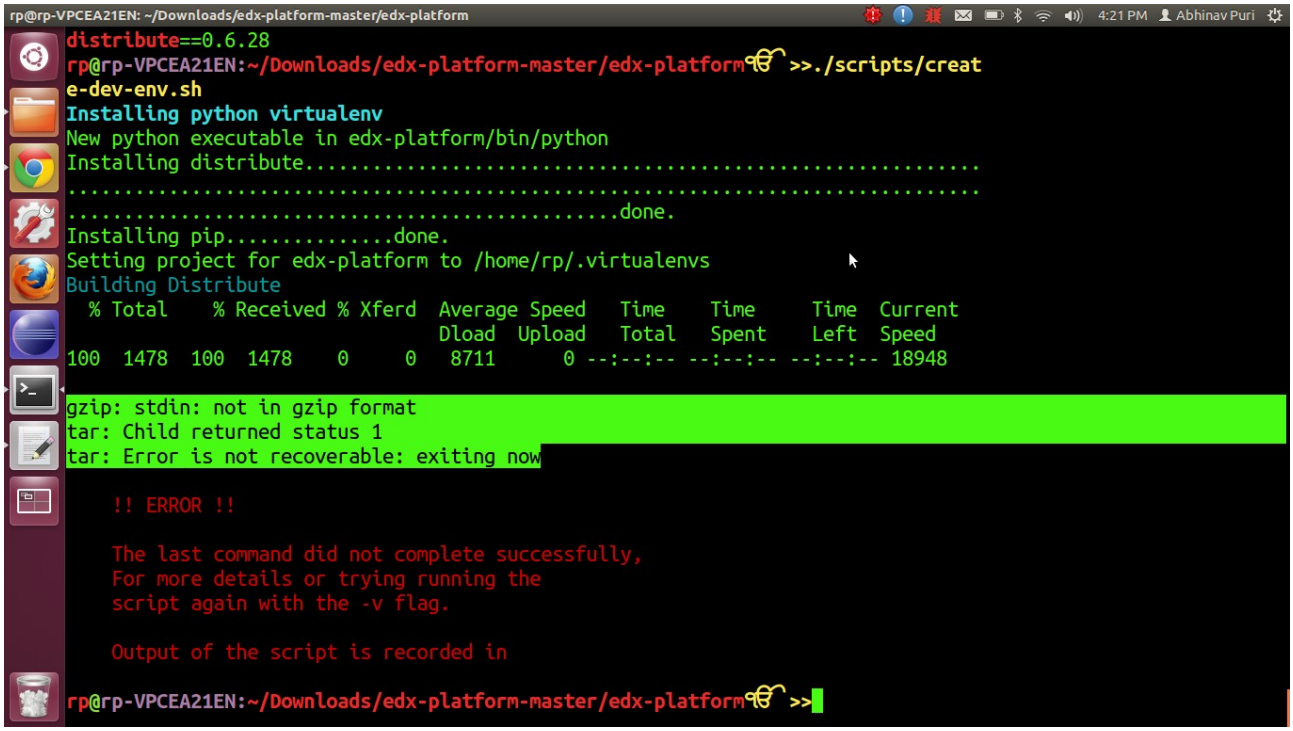
```
output "Installing edX pre-requirements"
pip install -r $BASE/edx-platform/requirements/edx/pre.txt
output "Installing edX requirements"
# Install prereqs
cd $BASE/edx-platform
rvm use "$RUBY_VER@edx-platform"
rake install_prereqs

# Final dependecy
output "Finishing Touches"
cd $BASE
pip install argcomplete
cd $BASE/edx-platform
bundle install
```

- After commenting, save this script and then run it.
  Type in shell (in directory : 'edx-platform/scripts/'):
  ./create-dev-env.sh

(This should run completely and integrate the whole edx-platform in your system.)

- *Errors that may occur* :

1. Virtual Environment distribution package error ::



Solution :
refer to Remark #2 (curl issues).

- **Run the project** :
  - To run Studio, run:

    $ rake cms

  - To run the LMS, run:

    $ rake lms[cms.dev]

Studio runs on port 8001, while LMS runs on port 8000, so you can run both of these commands simultaneously, using two different terminal windows. To view Studio, visit 127.0.0.1:8001 in your web browser; to view the LMS, visit 127.0.0.1:8000.

There's also an older version of the LMS that saves its information in XML files in the data directory, instead of in Mongo. To run this older version, run:

$ rake lms

Reference Remarks :

- Remark #1 (pip issues) :

  If you have not set environment proxy ::
  Refer to proxy people topic of this documentation.

  If still problem comes, then ::
  (NOT TO BE USED WHEN IN VIRUTAL ENVIRONMENTS)
  Sometimes sudo forget environment proxy ,
  Type (in shell):
  sudo -s
  (This command makes you use shell as a root user)
  Now, in this root session, again check environment proxy as in proxy people topic and do as given in that topic.
  After doing as above, type your pip command. (It should work.)

  If still problem comes, then ::
  Add proxy switch to pip, Type :
  sudo pip –proxy=http://<username>:<password>@<proxy>:<port no> install <package_name>

  If still problem prevails ::
  Download <package_name> from pypi.org and then extract the downloaded tar file.
  In terminal reach to the extracted folder's destination and type :
  sudo python setup.py install.
  (It has to work.)

- Remark #2 (curl issues) :

  If curl is not able to download properly :

  sudo nano ~/.curlrc

  Add the line :

  proxy=http://<username>:<password>@<proxy>:<port no>
