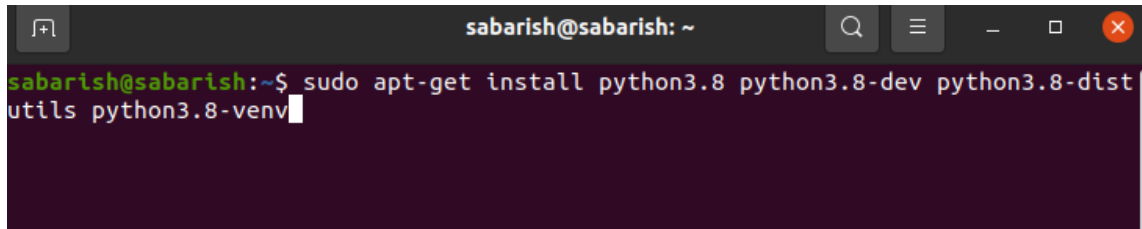


## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

### STEP 1: PYTHON INSTALLATION

\* Python 3.8 should be installed initially. To install python 3.8 open terminal and in cmd

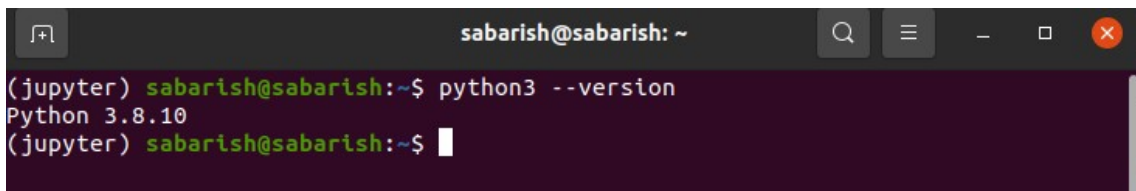
~\$ **sudo apt-get install python3.8 python3.8-dev python3.8-distutils python3.8-venv**



```
sabarish@sabarish: ~  
sabarish@sabarish:~$ sudo apt-get install python3.8 python3.8-dev python3.8-distutils python3.8-venv
```

\* To check whether it is installed or not

~\$ **python3 --version**

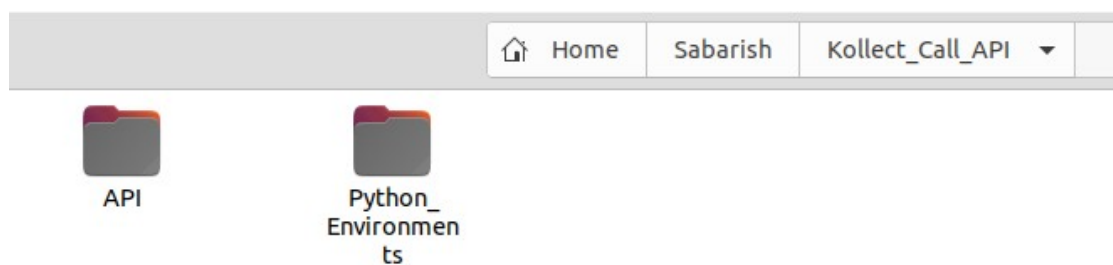


```
(jupyter) sabarish@sabarish:~$ python3 --version  
Python 3.8.10  
(jupyter) sabarish@sabarish:~$
```

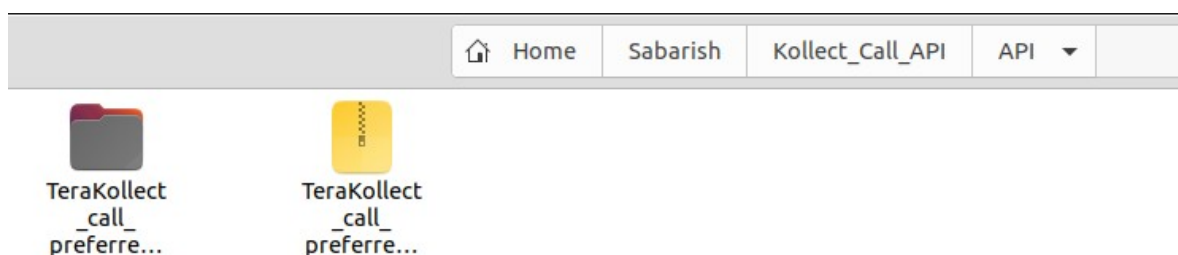
### STEP 2: FOLDER STRUCTURE

\* To create the folder structure go to the respective path you've allocated for installation and create a following directory

\* API                      \*Python\_Environment



In API folder paste the source file and extract the file

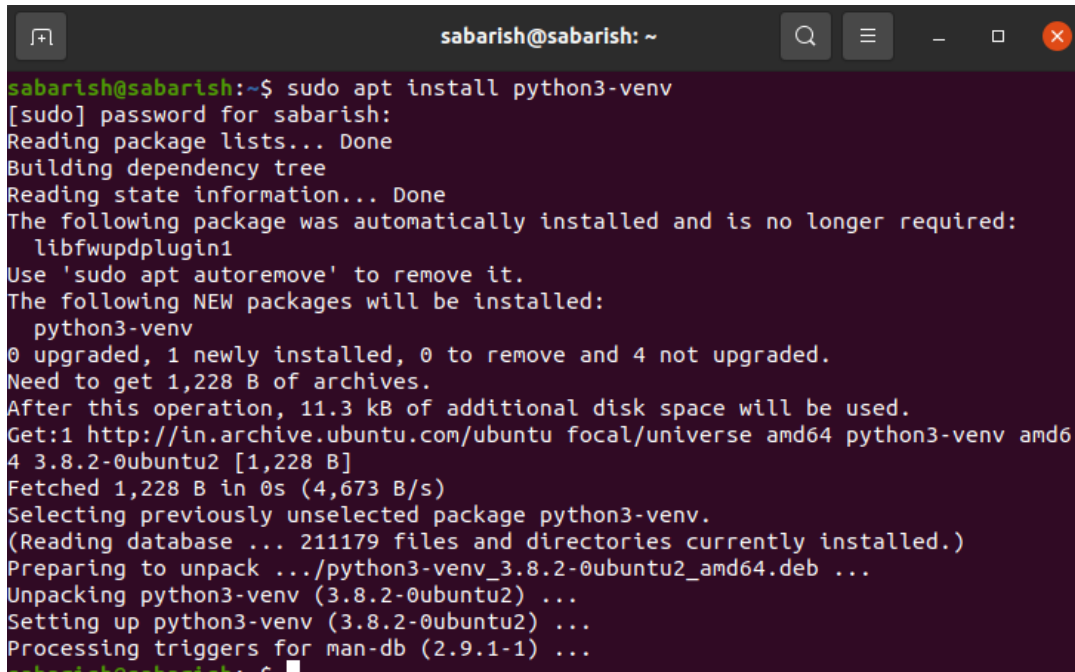


## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

### STEP 3: ENVIRONMENT SETUP

\* Create the virtual environment and install necessary libraries to the respective environment. To create the virtual environment, first to install virtual environment setup

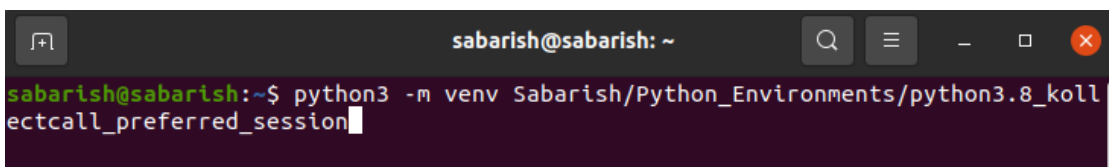
~\$ **sudo apt install python3-venv**

A terminal window titled 'sabarish@sabarish: ~' showing the command 'sudo apt install python3-venv'. The output indicates that the package was successfully installed. It shows the package lists, dependency tree, and state information. It notes that 'libfwupdplugin1' was automatically installed and is no longer required. It lists the new packages to be installed: 'python3-venv'. It shows that 0 packages were upgraded, 1 was newly installed, 0 were to be removed, and 4 were not upgraded. It states that 1,228 B of archives are needed, and after this operation, 11.3 kB of additional disk space will be used. It shows the download progress from 'http://in.archive.ubuntu.com/ubuntu focal/universe amd64 python3-venv amd64 3.8.2-0ubuntu2 [1,228 B]'. It shows the fetched size and speed. It shows the selection of the previously unselected package 'python3-venv'. It shows the preparation to unpack the package. It shows the unpacking of 'python3-venv (3.8.2-0ubuntu2)'. It shows the setting up of 'python3-venv (3.8.2-0ubuntu2)'. It shows the processing of triggers for 'man-db (2.9.1-1)'. The prompt returns to 'sabarish@sabarish:~\$'.

and followed by to create the virtual environment in Python\_Environment folder we've created

~\$ **python3 -m venv Sabarish/Python\_Environments/python3.8\_kollectcall\_preferred\_session**

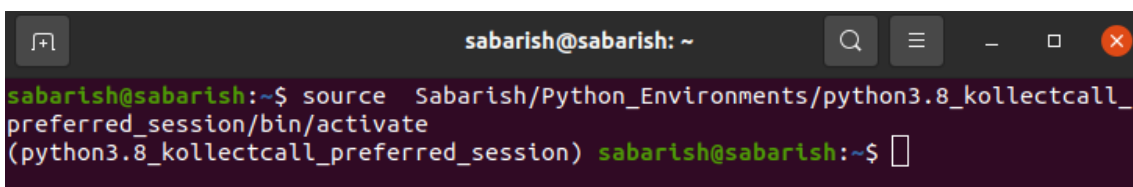
here Sabarish/Python\_Environments is the path for installing python environment and python3.8\_kollectcall\_preferred\_session is the environment name

A terminal window titled 'sabarish@sabarish: ~' showing the command 'python3 -m venv Sabarish/Python\_Environments/python3.8\_kollectcall\_preferred\_session'. The command is entered and the prompt returns to 'sabarish@sabarish:~\$'.

once the environment installed you may find the folder in the respective path you've mentioned

\* To activate the created python environment

~\$ **source Sabarish/Python\_Environments/python3.8\_kollectcall\_preferred\_session/bin/activate**

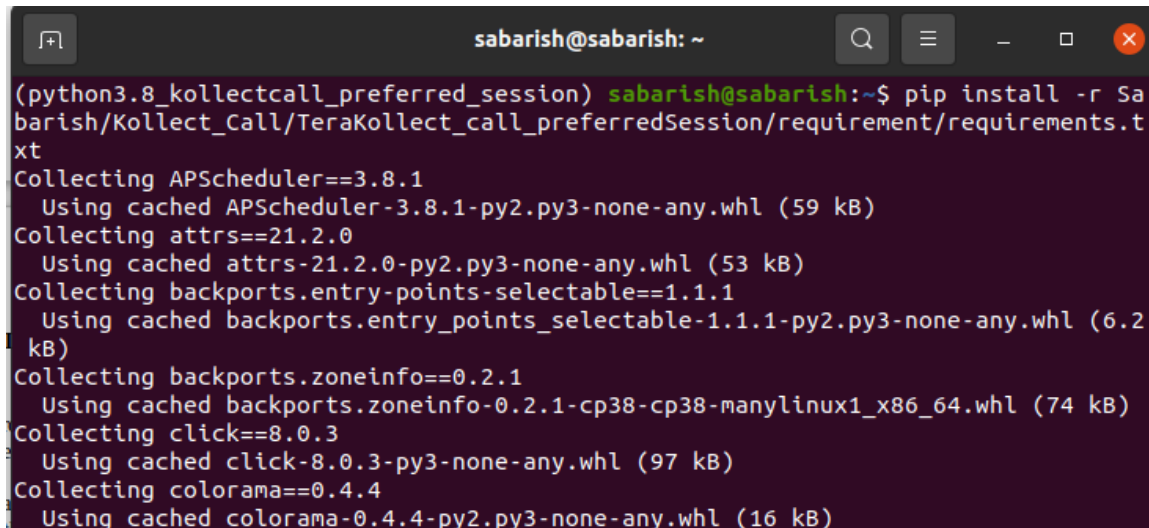
A terminal window titled 'sabarish@sabarish: ~' showing the command 'source Sabarish/Python\_Environments/python3.8\_kollectcall\_preferred\_session/bin/activate'. The output shows the prompt changing to '(python3.8\_kollectcall\_preferred\_session) sabarish@sabarish:~\$'.

## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

here environment path followed by environment name and then /bin/activate will activate your environment. you may find the environment name in terminal

\* Now install the necessary libraries created environment. first navigate to project directory we've extracted and in that find requirement folder & in that requirement.txt have the all the necessary libraries

~\$ **pip install -r Sabarish/Kollect\_Call\_API/API/TeraKollect\_call\_preferredSession/requirement/requirements.txt**



```
sabarish@sabarish: ~  
(python3.8_kollectcall_preferred_session) sabarish@sabarish:~$ pip install -r Sa  
barish/Kollect_Call/TeraKollect_call_preferredSession/requirement/requirement.t  
xt  
Collecting APScheduler==3.8.1  
  Using cached APScheduler-3.8.1-py2.py3-none-any.whl (59 kB)  
Collecting attrs==21.2.0  
  Using cached attrs-21.2.0-py2.py3-none-any.whl (53 kB)  
Collecting backports.entry-points-selectable==1.1.1  
  Using cached backports.entry_points_selectable-1.1.1-py2.py3-none-any.whl (6.2  
kB)  
Collecting backports.zoneinfo==0.2.1  
  Using cached backports.zoneinfo-0.2.1-cp38-cp38-manylinux1_x86_64.whl (74 kB)  
Collecting click==8.0.3  
  Using cached click-8.0.3-py3-none-any.whl (97 kB)  
Collecting colorama==0.4.4  
  Using cached colorama-0.4.4-py2.py3-none-any.whl (16 kB)
```

it will take some time based on the libraries we've used

### STEP 4: DATABASE PREREQUISITES

\* Fetching data from the customer\$information table and make sure below fields are exist in that table

~\$ **SELECT ID, GENDER, CUSTOMER\_RISK\_STATUS, INDUSTRY, RESIDENCE, CUSTOMER\_STATUS, PROMISE\_SUCCESS, TIME\_STAMP, REPAY\_CURRENCY, STAGE from customer\$information**

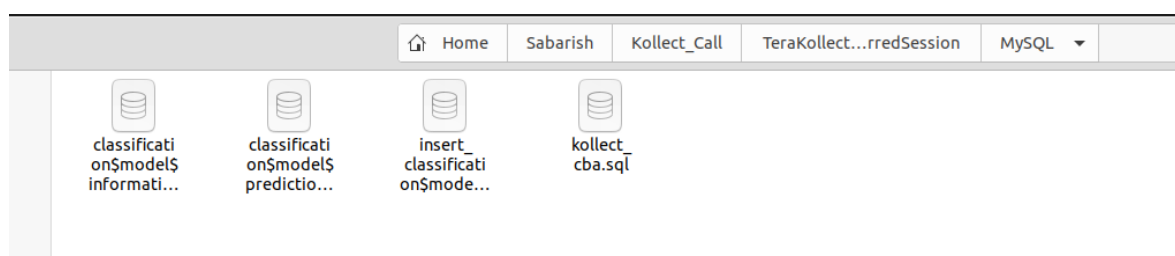
\* In further need two table's to store the model information and input/output details

1) **classification\$model\$information**

2) **classification\$model\$prediction**

you may find the DDL script in project file under MYSQL folder and do execute the script.

\* one last to execute the insert scripts in the **insert\_classification\$model\$information.sql** under MYSQL folder in project directory



## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

### STEP 5: APPLICATION PREREQUISITES

\* we are almost done, on last final step to setup is to set up database credentials in config file. To do that in source file go to the config folder and open **config.sh** using notepad. In that change the values as per the requirement

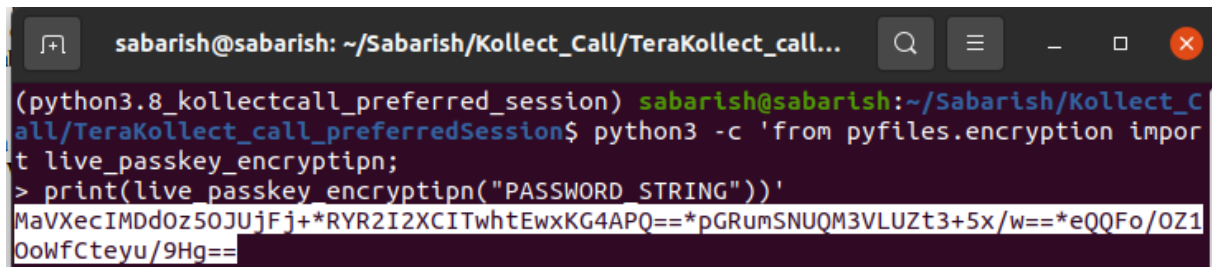
To encrypt the MYSQL password do the below

\* Activate the environment as per above commands

\* Navigate to project directory using above guidance and do the follow

```
~$ source config/config.sh
```

```
~$ python3 -c 'from pyfiles.encryption import live_passkey_encryptpn;
print(live_passkey_encryptpn("PASSWORD_STRING"))'
```



```
sabarish@sabarish: ~/Sabarish/Kollect_Call/TeraKollect_call...
(python3.8_kollectcall_preferred_session) sabarish@sabarish:~/Sabarish/Kollect_Call/TeraKollect_call_preferredSession$ python3 -c 'from pyfiles.encryption import live_passkey_encryptpn;
> print(live_passkey_encryptpn("PASSWORD_STRING"))'
MaVXecIMDdOz50JUjFj+*RYR2I2XCITwhtEwxKG4APQ==*pGRumSNUQM3VLUZt3+5x/w==*eQQFo/OZ1
OoWfCteyu/9Hg==
```

Replace password string with the respective password and enter you will get the encrypted string . Replace the encrypted string with LIVE\_MYSQL\_PASSWORD string

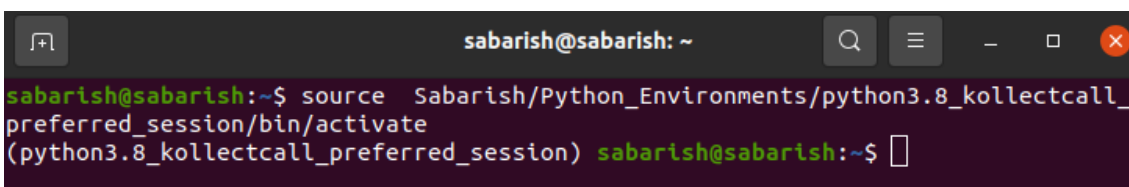
```
export LIVE_MYSQL_USER="root"
export LIVE_MYSQL_PASSWORD="5WWLgw==*1ahuUFcWTl2sRKbQ==" # Password encrypted
export LIVE_MYSQL_HOST="127.0.0.1"
export LIVE_MYSQL_PORT=3306
export LIVE_MYSQL_DB="kollect_cba"
```

```
## UAT/LIVE ENVIRONMENT MYSQL CONFIG
export LIVE_MYSQL_USER="root"
export LIVE_MYSQL_PASSWORD="+JRs6wwUJA==*uuYEPmy6/9eshVbHbQ/U4A==*D7XI1h41MMduhM635WwLgw==*1aha5rPdMuUFcWTl2sRKbQ==" # Password encrypted
export LIVE_MYSQL_HOST="127.0.0.1"
export LIVE_MYSQL_PORT=3306
export LIVE_MYSQL_DB="kollect_cba"
```

### STEP 6 : RUN OUR APPLICATION

\* To run our application first to active the python environment we've created initially. To active the environment navigate the respective directory and do activate

```
~$ source Sabarish/Python_Environments/python3.8_kollectcall_preferred_session/bin/activate
```



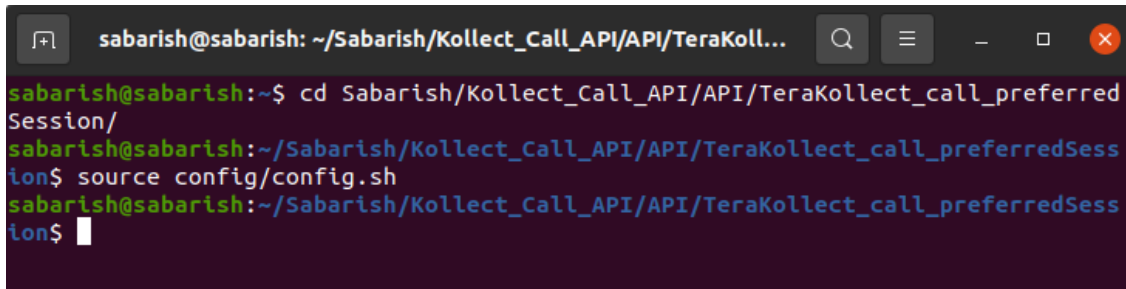
```
sabarish@sabarish: ~
sabarish@sabarish:~$ source Sabarish/Python_Environments/python3.8_kollectcall_preferred_session/bin/activate
(python3.8_kollectcall_preferred_session) sabarish@sabarish:~$
```

\* Next, to active the configuration files which has the DB related information. To active the config file navigate to project directory and execute below commands

## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

~\$ cd Sabarish/Kollect\_Call\_API/API/TeraKollect\_call\_preferredSession/

~\$ source config/config.sh

A terminal window with a dark background. The title bar shows 'sabarish@sabarish: ~/Sabarish/Kollect\_Call\_API/API/TeraKoll...'. The terminal content shows the user navigating to the directory and sourcing the configuration file.

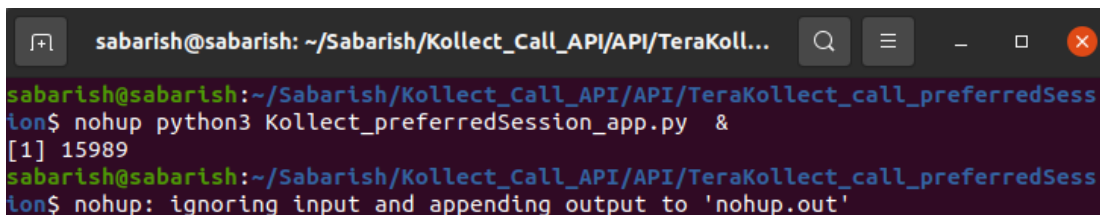
```
sabarish@sabarish:~$ cd Sabarish/Kollect_Call_API/API/TeraKollect_call_preferredSession/
sabarish@sabarish:~/Sabarish/Kollect_Call_API/API/TeraKollect_call_preferredSession$ source config/config.sh
sabarish@sabarish:~/Sabarish/Kollect_Call_API/API/TeraKollect_call_preferredSession$
```

\* To execute the application as background process

~\$ nohup python3 Kollect\_preferredSession\_app.py &  
or

~\$ nohup python Kollect\_preferredSession\_app.py &

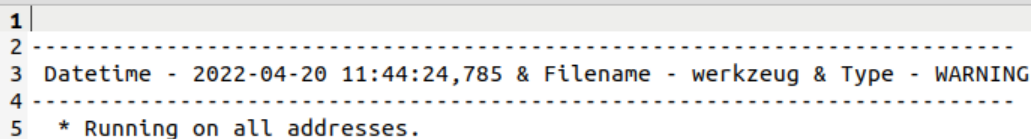
and close the terminal further

A terminal window showing the execution of the nohup command to run the application in the background.

```
sabarish@sabarish:~/Sabarish/Kollect_Call_API/API/TeraKollect_call_preferredSession$ nohup python3 Kollect_preferredSession_app.py &
[1] 15989
sabarish@sabarish:~/Sabarish/Kollect_Call_API/API/TeraKollect_call_preferredSession$ nohup: ignoring input and appending output to 'nohup.out'
```

### STEP 7: VALIDATE APPLICATION RUNNING OR NOT

\* once application is up, the application creates the log files in source file directory under folder log named **errorlog.log**. You may open using notepad and see the latest time you've started the application.

A screenshot of a log file with line numbers 1 through 5. The log contains a warning message about running on all addresses.

```
1 |
2 | -----
3 | Datetime - 2022-04-20 11:44:24,785 & Filename - werkzeug & Type - WARNING
4 | -----
5 | * Running on all addresses.
```

\* In another way we may pass the respective url and input details in postman to get our respective output

METHOD TYPE : post

URL: [http://127.0.0.1:2012/kollectCall\\_preferredSession\\_xgbClassifier\\_modelPredictions](http://127.0.0.1:2012/kollectCall_preferredSession_xgbClassifier_modelPredictions)

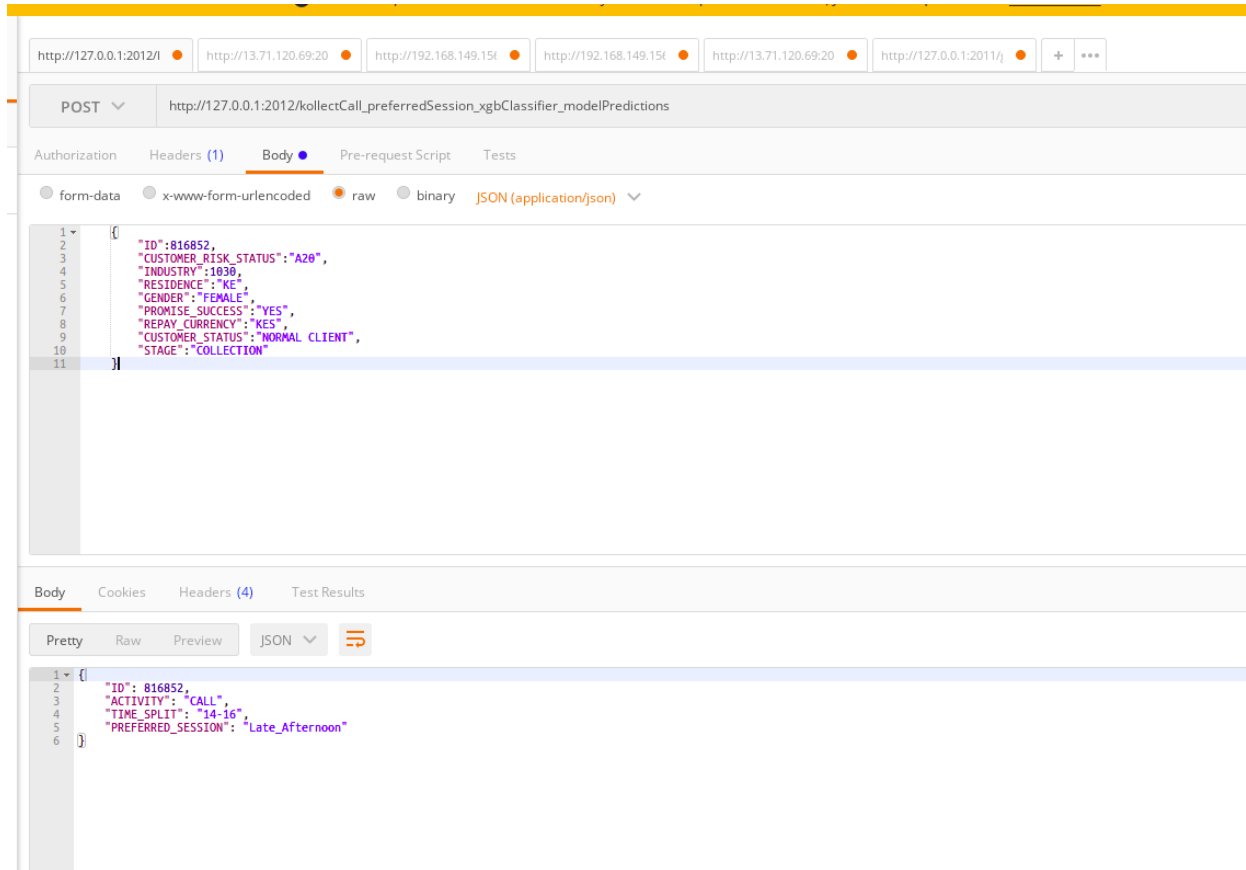
INPUT: {

```
"ID":816852,
"CUSTOMER_RISK_STATUS":"A20",
"INDUSTRY":1030,
"RESIDENCE":"KE",
"GENDER":"FEMALE",
"PROMISE_SUCCESS":"YES",
"REPAY_CURRENCY":"KES",
"CUSTOMER_STATUS":"NORMAL CLIENT",
"STAGE":"COLLECTION"
```

## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

```
}
```

and you'll get the respective output as mentioned



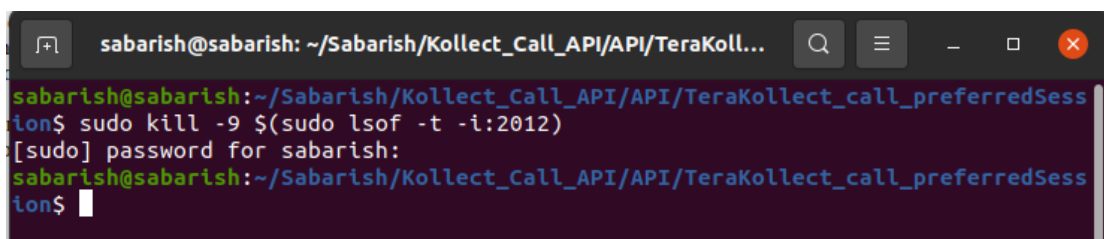
### \* FURTHER VERSION'S DEPLOYMENT

In the above we've seen how to deploy the fresh version. here quickly brief how to deploy the next and further releases

- \* Extract the latest file to the destination folder
- \* Activate the respective python environment using above commands
- \* navigate to the respective project directory in terminal

\* First step to terminate the process we've executed already in past. We're running our application in port no 2012 and going to kill our process using port number. To do that in cmd

```
~$ sudo kill -9 $(sudo lsof -t -i:2012)
```



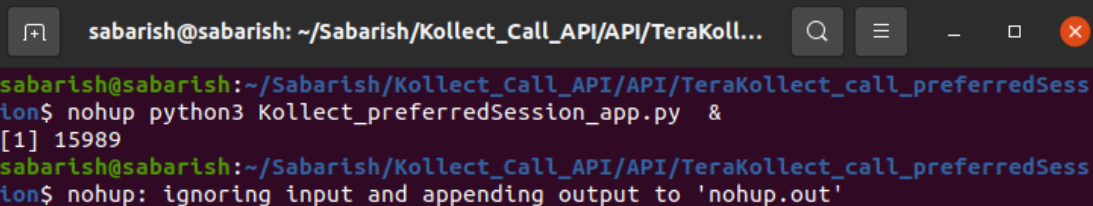
## TERA KOLLECT ML PACK DEPLOYMENT INSTRUCTIONS – LINUX (UBUNTU)

and to Run our application again

```
~$ nohup python3 Kollekt_preferredSession_app.py &  
or
```

```
~$ nohup python Kollekt_preferredSession_app.py &
```

and close the terminal further



```
sabarish@sabarish: ~/Sabarish/Kollekt_Call_API/API/TeraKoll...  
sabarish@sabarish:~/Sabarish/Kollekt_Call_API/API/TeraKollekt_call_preferredSess  
ion$ nohup python3 Kollekt_preferredSession_app.py &  
[1] 15989  
sabarish@sabarish:~/Sabarish/Kollekt_Call_API/API/TeraKollekt_call_preferredSess  
ion$ nohup: ignoring input and appending output to 'nohup.out'
```

and this need's to be validated further, whether the application running or not using the above methods mentioned