Ansible Training

MANUAL STEP:

Write a playbook to install and configure Jenkins

We are going to keep ansible inventory in same directory of ansible-playbook.

1. Install all required dependencies

Python and its dependencies

```
apt-get install -y python python-setuptools python-dev build-essential python3-pip python-pip python-mysqldb
```

2. Install and Configure Database

Install MySQL database

```
apt-get install -y mysql-server mysql-client
```

3. Start Database Service

- Start the database service

  ```
  service mysql start
  ```

- Create database and database users

```
# mysql -u <username> -p

mysql> CREATE DATABASE employee_db;
mysql> GRANT ALL ON *.* to db_user@'%' IDENTIFIED BY 'Passw0rd';
mysql> USE employee_db;
mysql> CREATE TABLE employees (name VARCHAR(20));
```

- Insert some test data

```
mysql> INSERT INTO employees VALUES ('RAJESH');
```

4. Install and Configure Web Server

Install Python Flask dependency

```
pip install flask
pip install flask-mysql
```

- Copy app.py and save it to /opt/app.py

```
import os
from flask import Flask
from flaskext.mysql import MySQL      # For newer versions of flask-mysql
# from flask.ext.mysql import MySQL   # For older versions of flask-mysql
app = Flask(__name__)

mysql = MySQL()

mysql_database_host = 'MYSQL_DATABASE_HOST' in os.environ and os.environ['MYSQL_DATABASE_HOST'] or  'localhost'

# MySQL configurations
app.config['MYSQL_DATABASE_USER'] = 'db_user'
app.config['MYSQL_DATABASE_PASSWORD'] = 'Passw0rd'
app.config['MYSQL_DATABASE_DB'] = 'employee_db'
app.config['MYSQL_DATABASE_HOST'] = mysql_database_host
mysql.init_app(app)

conn = mysql.connect()

cursor = conn.cursor()

@app.route("/")
def main():
    return "Welcome!"

@app.route('/how are you')
def hello():
    return 'I am good, how about you?'

@app.route('/read from database')
def read():
    cursor.execute("SELECT * FROM employees")
    row = cursor.fetchone()
    result = []
    while row is not None:
      result.append(row[0])
      row = cursor.fetchone()

    return ",".join(result)
```

```
if __name__ == "__main__":
    app.run()
```

- Configure database credentials and parameters

5. Start Web Server

Start web server

```
export FLASK_APP=app.py
```

```
flask run --host=0.0.0.0
```

6. Test

Open a browser and go to URL

```
http://<IP>:5000                          => Welcome
http://<IP>:5000/how%20are%20you          => I am good, how about you?
http://<IP>:5000/read%20from%20database   => RAJESH
```