DEPLOY part of Integrated Learning Project (ILP):

Prerequisite:

1. You should have completed the Build part of ILP and ready with the war file
2. You should have completed the hands-on activity provided for Day11 to Day14.

What do Do?

1. Write Ansible Playbook that would create the required run time environment for your application, fulfilling all dependencies required.
   Ex: If you created the Web Application that required TOMCAT and JDK, your playbook should install JDK, enable the path to access JAVA tools and it should install TOMCAT
   Note: JDK and JAVA tools may be already there on your TOPGEAR machine. Even TOMCAT would have been already installed, hence make sure that you expose your TOMCAT server on a different port.
2. You can manually copy the web application (Ex: Application.war file) into your installation done in the first step.
3. You may extend your Ansible Playbook to copy the web application into your installation.
4. You may use Jenkins to copy the web application into your installation extending your BUILD part by integrating Jenkins to do deployment
5. You may use Jenkins to invoke Ansible Playbook to create the environment so that entire tool chain can be automated.
6. You may go further to enhance your automation to smoothen the application life cycle management.

What is Required?

1. Step 1 and 2 defined above are essentially need to be done, as writing Ansible Playbook is part of deliverable.
2. Step 3 is optional to use Ansible as alternate deployment tool rather than Jenkins.
3. Step 4 and above are completely optional and not required to be submitted as deliverable. These steps are stepping stone for Capstone Project.

What to submit as deliverable?

1. You need to submit your playbook as deliverable.
2. Your playbook must be in the form of Ansible Role.
3. You should provide tree structure of Ansible Role.

Capture the following commands in your script file, and submit along with screenshot. Screenshot may be submitted as png file or may be copied to word document before submitting in GITLAB.

Example:

```
osgdev@TG-DevOps-OS004:~/ansilab$ tree ./roles/newrole/
./roles/newrole/
├── defaults
│   └── main.yml
├── files
```

```
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── README.md
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml
```

4. You should list the content of files under tasks, handlers and vars

Example:

```
osgdev@TG-DevOps-OS004:~/ansilab$ cat ./roles/sample/tasks/main.yaml
- name: To Create a folder
  file:
    path: "{{folder_path}}/NEWSAMPLE4"
    state: directory
    mode: 0755
  notify:
    - To Create a file

osgdev@TG-DevOps-OS004:~/ansilab$ cat ./roles/sample/handlers/main.yaml
- name: To Create a file
  file:
    path: "{{folder_path}}/NEWSAMPLE4/new_file4"
    state: touch

osgdev@TG-DevOps-OS004:~/ansilab$ cat ./roles/sample/vars/main.yaml
folder_path: /home/osgdev/ansilab
```

5. You must submit the screen shot capturing the output of your project:

Make sure the IP address of your machine shown below matches with the IP address appearing in the screenshot (Do not use localhost:4949)

Example: Capture the IP address of your machine:

```
osgdev@TG-DevOps-OS004:~$ ifconfig
ens160    Link encap:Ethernet  HWaddr 00:50:56:a0:70:f2
          inet addr:10.199.0.112  Bcast:10.199.15.255  Mask:255.255.240.0
          inet6 addr: fe80::250:56ff:fea0:70f2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16552146 errors:0 dropped:0 overruns:0 frame:0
          TX packets:231571 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1896872943 (1.8 GB)  TX bytes:77480740 (77.4 MB)
```