

Practices for Lesson 1: Build, Deploy, and Run in Kubernetes

Provision the Practice Environment

Overview

In the first part of these practices, you connect to OCI and provision a Linux Compute instance, Oracle Container for Kubernetes Cluster (OKE) and OCI Registry repository. You will access the Linux compute instance using SSH and VNC, installed on your local computer.

About the Practices

These practices are taken from the Oracle University course **Practical Cloud Native for Java Developers – Build, Deploy and Run in Kubernetes**, in the **Application Development Learning Subscription**. This is a free learning subscription that consists of many OCI courses that will be of interest to you. You can find this and other learning subscriptions at <https://learn.oracle.com>.

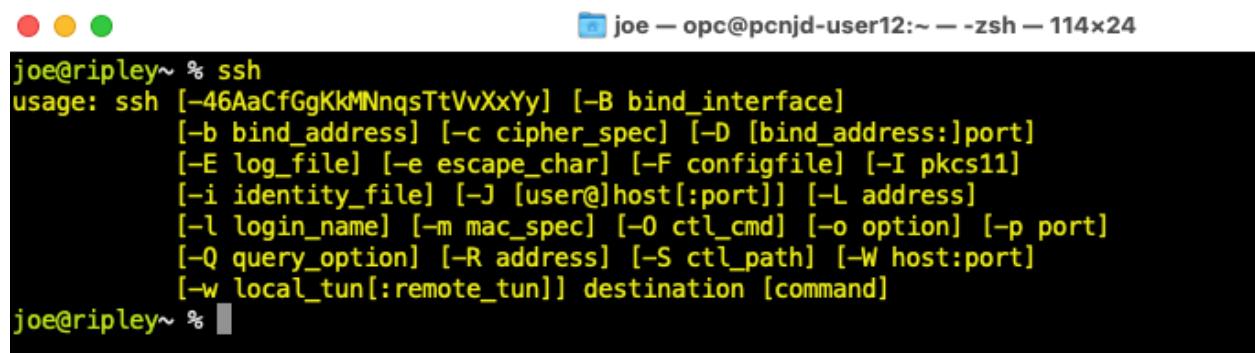
The course this is taken from teaches practical steps for migrating an open-source build, deploy and run in Kubernetes solution to OCI using Oracle Container for Kubernetes (OKE) and Visual Builder Studio for CI/CD.

Practice Requirements

The computer you use for these labs must meet the following requirements.

- In order to access the OCI Compute Linux instance, you need to use SSH. If you have a Macintosh or Linux computer, then you would only need to open a terminal window and type “ssh” to confirm it is installed and working.
- On a Windows PC you will need to enable OpenSSH, if it is not already enabled. You can test this by opening a command prompt and typing ssh.

Linux/UNIX/Mac OS



A screenshot of a terminal window on a Linux/UNIX/Mac OS system. The window title is "joe — opc@pcnjd-user12:~ — zsh — 114x24". The terminal shows the user "joe" at the prompt "joe@ripley~ %". The user types "ssh" and receives the following usage information:

```
joe@ripley~ % ssh
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command]
```

Windows.

1. Confirm OpenSSH client is enabled. Go to: **Settings > Apps > Apps and Features > Optional Features.**

The screenshot shows the Windows Settings interface. At the top, there's a back arrow and the word "Settings". Below that is a button labeled "Optional features". Underneath is a heading "Optional features" with a house icon. A large plus sign button is labeled "Add a feature". Below it is a link "See optional feature history".

Installed features

Find an installed optional feature 🔍

Sort by: Name ▾

	Name	Size	Last modified
	Internet Explorer 11	3.20 MB	12/7/2019
	Math Recognizer	33.3 MB	
	Microsoft Paint	6.68 MB	
	Microsoft Quick Assist	2.89 MB	
	Notepad	632 KB	
	OpenSSH Client	10.1 MB	

Open a Command Window: **Start > Run > cmd**

- You will also need a VNC client. TigerVNC is a free client and is available for download from:
<https://sourceforge.net/projects/tigervnc/files/stable/1.11.0/>
For windows, download the vncviewer: [vncviewer64-1.11.0.exe](https://sourceforge.net/projects/tigervnc/files/stable/1.11.0/vncviewer64-1.11.0.exe)
- If you have an active VPN you will not be able to connect to the remote Linux environment successfully over SSH. You **must** disconnect from the VPN to access the Linux environment to run the labs. If that is not possible for security reasons on your current PC, you will need to find another PC to run the labs on.

It needs only an internet connection, browser and support for SSH, per above. A basic home PC is sufficient, as all computing is performed in the cloud.

Part I: Provision and Set Up OCI Compute Instance

Overview

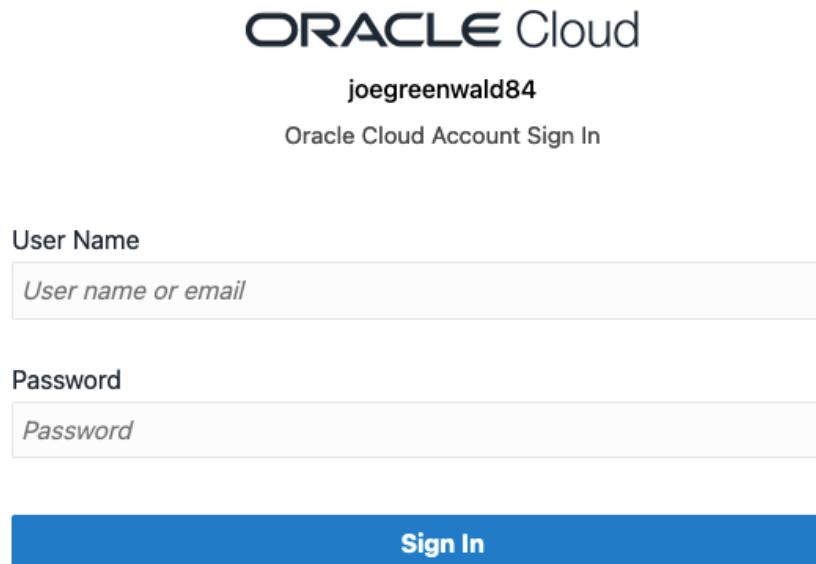
In this practice, you provision the OCI Compute instance, download its key files, run a script to configure the instance and download the lab files.

Assumptions

You have received your lab access details from your instructor or Lab Details tab.

Tasks

1. Sign in to **Oracle Cloud Infrastructure Console**.
 - a. In a browser on your local machine, enter the **Oracle Cloud Infrastructure Console URL** provided to you:
`https://cloud.oracle.com/?tenant=<tenancy name>&provider=OracleIdentityCloudService`
 - b. Enter username and password that is assigned to you. Details are provided in the email you received when you registered for the Free Trial account.



- c. You will now have access to OCI Console.
2. Create your Compute Instance.

- On the left, you will see the Navigation menu; navigate to **Compute** and select **Instances**.

The screenshot shows the Oracle Cloud Compute Instances page. On the left, there's a navigation sidebar with a menu icon (three horizontal lines) and a search bar. Below the search bar are several service links: Home, Compute (which is highlighted with a red box), Storage, Networking, Oracle Database, Databases, and Analytics & AI. To the right of the sidebar is the main content area. At the top of the content area is a header with the Oracle Cloud logo and a search bar. Below the header is a title 'Compute' with a gear icon. Underneath the title is another 'Compute' section with a green star icon, followed by a list of options: Instances (which is also highlighted with a red box), Dedicated Virtual Machine Hosts, Instance Configurations, Instance Pools, Cluster Networks, Autoscaling Configurations, Capacity Reservations, and Custom Images.

- Select your compartment. Your compartment will be the root compartment. Click the **Create Instance** button.

The screenshot shows the Oracle Cloud Instances page. On the left, there's a sidebar with 'Compute' and 'Instances' selected. The main content area has a heading 'Instances in joegreenwald84 (root) Compartment'. Below the heading is a brief description of the Compute service. In the center, there's a 'Create Instance' button (highlighted with a red box). Below the button is a table showing existing instances. The table has columns for Name, State, Public IP, Private IP, and Shape. One instance named 'HOL_test' is listed, showing it's running with the IP 158.101.116.208 and shape VM.Standard. At the bottom of the page, there's a 'List Scope' section with a 'Compartment' dropdown menu. The dropdown shows 'joegreenwald84 (root)' selected (highlighted with a red box).

- c. On the **Create Compute Instance** page enter a name for your instance as: pcnjd.

The screenshot shows the 'Create Compute Instance' page in Oracle Cloud. The 'Name' field is filled with 'pcnjd-user12' and is highlighted with a red box. Below it, the 'Create in compartment' dropdown is set to 'C12'. The 'Placement' section shows three availability domains: AD 1 (selected), AD 2, and AD 3. The 'Image and shape' section is collapsed. The 'Image' section shows 'Oracle Linux 7.9' selected, with a 'Change Image' button.

- d. In the **Image and Shape** section, click **Edit** and then the **Change Image** button to select a new image.

The screenshot shows the 'Image and shape' section. The 'Image' section displays 'Oracle Linux 7.9' with a 'Change Image' button. The 'Change Image' button is highlighted with a red box.

- e. When presented with the **Browse All Images** page, select **Oracle Images** from the **Image source** pulldown.

The screenshot shows the 'Browse All Images' page. The 'Image source' dropdown is open, showing 'Platform Images' and 'Oracle Images'. 'Oracle Images' is selected and highlighted with a blue background. Other options include 'Partner images', 'Custom images', 'Community images', 'Boot volumes', and 'Image OCID'. A note at the bottom states: 'Platform Images are pre-built operating systems for Oracle Cloud Infrastructure.'

- f. Navigate to the second page by clicking the > link in the bottom right corner.

Hint: If the help icon is in the way of the > link, click and drag its six-dot rectangle to move it out of the way.



Browse All Images

An image is a template of a virtual hard drive that determines the operating system and other software for an instance.

Image source

Oracle images

Compartment

C12

ocuocictrng6 (root)/C12

Choose from Oracle enterprise images and solutions enabled for Oracle Cloud Infrastructure.

App Name	Publisher	Price
<input type="checkbox"/> AI (All-in-One) GPU Image for Data Science v3	Oracle	Free
<input type="checkbox"/> Enterprise Manager 13c Workshop	Oracle	Free
<input type="checkbox"/> FoldingATHome GPU Image	Oracle	Free
<input type="checkbox"/> Genome Analysis Toolkit	Oracle	Free
<input type="checkbox"/> JD Edwards EnterpriseOne One-Click Provisioning Server	Oracle	BYOL
<input type="checkbox"/> JD Edwards EnterpriseOne Reference Architecture (Terraform)	Oracle	BYOL
<input type="checkbox"/> JD Edwards EnterpriseOne Trial Edition	Oracle	BYOL
<input type="checkbox"/> Julia AI HPC GPU Image	Oracle	Free
<input type="checkbox"/> Kali Linux	Oracle	Free
<input type="checkbox"/> MySQL Enterprise Edition	Oracle	BYOL
0 Selected		Showing 10 Items < 1 of 6 >

- g. Select **Oracle Cloud Developer Image** and check the box to accept the terms and click the **Select Image** button.

Browse All Images

An image is a template of a virtual hard drive that determines the operating system and other software for an instance.

Image source

Oracle images

Compartment

C12

ocuocictrng6 (root)/C12

Choose from Oracle enterprise images and solutions enabled for Oracle Cloud Infrastructure.

App Name	Publisher	Price	
<input type="checkbox"/> NVIDIA GPU Cloud Machine Image	Oracle	Free	▼
<input type="checkbox"/> NVIDIA Quadro Virtual Workstation - Windows Server 2016 VM	Oracle	Free	▼
<input type="checkbox"/> NVIDIA Quadro Virtual Workstation - Windows Server 2019 VM	Oracle	Free	▼
<input type="checkbox"/> Oracle Audit Vault and Database Firewall	Oracle	BYOL	▼
<input checked="" type="checkbox"/> Oracle Cloud Developer Image	Oracle	Free	▼
<input type="checkbox"/> Oracle Cloud Developer Image (Autonomous Linux)	Oracle	Free	▼
<input type="checkbox"/> Oracle Communications Session Border Controller	Oracle	BYOL	▼
<input type="checkbox"/> Oracle Communications Session Router	Oracle	BYOL	▼
<input type="checkbox"/> Oracle Communications Subscriber-Aware Load Balancer	Oracle	BYOL	▼
<input type="checkbox"/> Oracle Database	Oracle	BYOL	▼

1 Selected

Showing 10 Items < 2 of 6 >



Agreement for Oracle App *Oracle Cloud Developer Image

I have reviewed and accept the [Oracle Terms of Use](#)

Select Image **Cancel**

- h. When you return to the **Create Compute Instance** page, in the **Image and Shape** section, click the **Change Shape** button.

Image and shape

[Collapse](#)

A **shape** is a template that determines the number of CPUs, amount of memory, and other resources allocated to an instance. The image is the operating system that runs on top of the shape.

Image

ORACLE Linux Oracle Cloud Developer Image
Oracle Cloud Developer Image

Change Image

Shape

AMD VM.Standard.E3.Flex
Virtual Machine, 1 core OCPU, 16 GB memory, 1 Gbps network bandwidth

Change Shape

- On the **Browse All Shapes** page, select **Intel** from the **Shape series** choices and select **VM.Standard2.2** for the **Shape Name**. Click the **Select Shape** button when done.

Note: If you get a warning – you can ignore it and select the shape and continue to create the compute instance.

Browse All Shapes

A shape is a template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance. See [Compute Shapes](#) for more information.

Instance type

Virtual Machine

A virtual machine is an independent computing environment that runs on top of physical bare metal hardware.

Bare Metal Machine

A bare metal compute instance gives you dedicated physical server access for highest performance and strong isolation.

Shape series

AMD

AMD Flexible OCPU count.
AMD processors.

Intel

Flexible OCPU count.
Intel processors.

Ampere

Arm-based processor.
AMPERE

Specialty and Previous Generation

Earlier generation AMD and Intel Standard shapes. Always Free, Dense I/O, GPU, and HPC shapes.

Shape Name	OCPU	Memory (GB)	Network Bandwidth (Gbps)	Max. Total VNICs	
<input type="checkbox"/> VM.Optimized3.Flex ⓘ	1	14	4	2	▼
<input type="checkbox"/> VM.Standard2.1	1	15	1	2	▼
<input checked="" type="checkbox"/> VM.Standard2.2	2	30	2	2	^
Local Disk: Block Storage Only					
<input type="checkbox"/> VM.Standard2.4	4	60	4.1	4	▼
<input type="checkbox"/> VM.Standard2.8	8	120	8.2	8	▼
<input type="checkbox"/> VM.Standard2.16	16	240	16.4	16	▼
<input type="checkbox"/> VM.Standard2.24	24	320	24.6	24	▼
1 Selected					Showing 7 It...

Don't see the shape you want? [View your service limits and request an increase.](#)

[Select Shape](#)

[Cancel](#)

- Scroll down to the **Add SSH keys** section.

Notes: There are several options for how SSH keys can be configured.

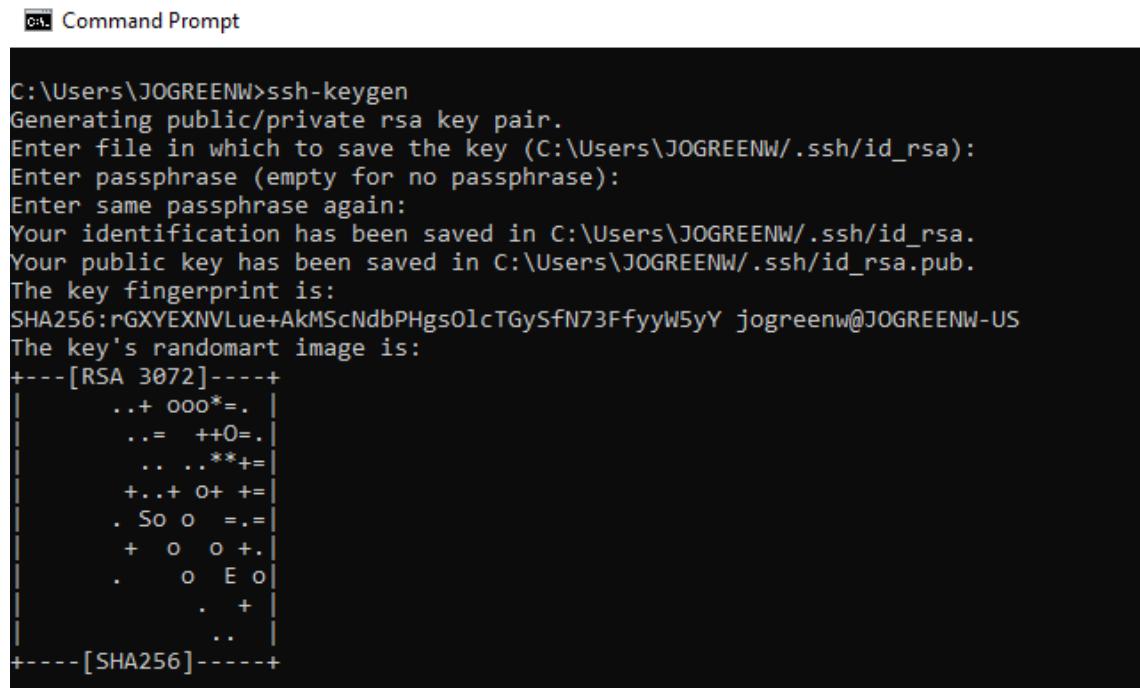
- You may request keys be generated by the Oracle Cloud environment and then download these keys.
- Or you may generate keys locally and either upload or copy-paste them to Oracle Cloud.
- If you already have SSH keys generated you should take care not to overwrite existing keys, and simply upload the existing public key and use that, instead of generating new ones.

- This document describes how to generate new keys and upload them to Oracle Cloud.

- k. Use **ssh-keygen** utility to generate your keypair locally.

Note: Only perform this step if you don't have an SSH key already present on your local machine. Otherwise, proceed to next step.

Open terminal or command prompt window and execute the `ssh-keygen` command. Accept the default file location. Press **enter** to set an empty passphrase.



```
C:\Users\JOGREENW>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\JOGREENW/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\JOGREENW/.ssh/id_rsa.
Your public key has been saved in C:\Users\JOGREENW/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:rGXYEXNVLue+AkMScNdbPHgsOlcTGySfN73FfyyW5yY jogreenw@JOGREENW-US
The key's randomart image is:
+---[RSA 3072]---+
| ..+ ooo*=.. |
| ... = ++0=.. |
| ... . .**+=.. |
| +...+ o+ +=.. |
| . So o -.-.. |
| + o o +.. |
| . o E o .. |
| . + .. |
| .. |
+---[SHA256]---
```

- I. Once the key has been generated, return to the Oracle Cloud Console browser window and use "**Upload public key file (.pub)**" option.

Add SSH keys

Generate an [SSH key pair](#) to connect to the instance using a Secure Shell (SSH) connection, or upload a public key that you already have.

Generate a key pair for me Upload public key files (.pub) Paste public keys No SSH keys

SSH public keys

Drop .pub files here. [Browse](#)

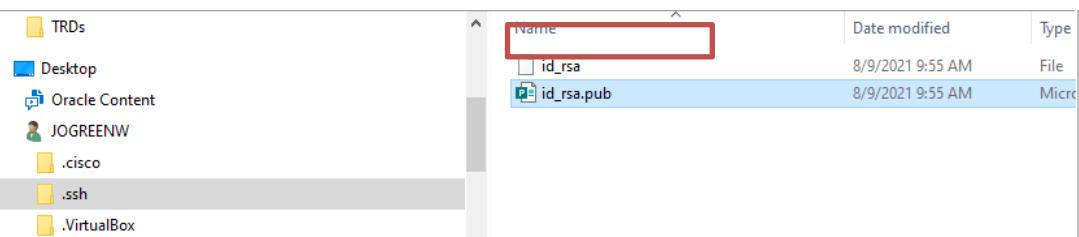
- m. Select `id_rsa.pub` file from the default location where key has been placed.

Note: By default, SSH keys are placed into the `.ssh` folder under the current user's home folder, depending on the platform this would look like:

Linux/UNIX: `/home/<user>/ .ssh`

MacOS: `/Users/<user>/ .ssh`

Windows: `C:\users\<user>\ .ssh`



Note: We do not recommend changing the default file name (`id_rsa`) and location! If you choose to use a custom file name and location for SSH keys:

- Custom SSH key files would have to be qualified every time you use the `ssh` command using `-i` option. For example:

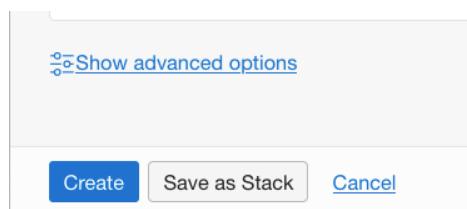
```
ssh -i <full path to the private key> opc@<public ip address>
```

- Custom SSH private key files must have only current user permissions applied. For example, to change permissions for the private key file on a POSIX (Linux, Unix, Mac) platform so it will only allow read access for the current user, use following command:

```
chmod 400 <private key file name>
```

- There is no need to perform these actions if you have followed the generate keys with `ssh-keygen` utility practice step above exactly and accepted default file name and location.

- n. Once key upload is complete, click the **Create** button to create the compute instance.



Note: It will take a couple of minutes to provision the instance. Once the provisioning process is complete you will see the **Public IP Address** information is filled in.

- o. When the **Public IP Address** information is filled in, click the **Copy** link and paste the compute instance's IP address to a local text editor and save it.

The screenshot shows the Oracle Cloud Instance Details page for an instance named 'pcnjd'. The 'Instance Information' tab is selected. In the 'General Information' section, the 'Public IP Address' field contains '129.213.203.137' and has a 'Copy' link next to it, which is highlighted with a red box. The 'Primary VNIC' section also lists the private IP address as '10.0.0.122'.

3. Open an SSH terminal connection to the remote host. You will need the public IP address you just copied for the provisioned remote compute instance.
 - a. Open a terminal or Command Prompt to open an SSH connection to your newly provisioned instance. Enter:

```
ssh opc@<public ip address>
```

Note: Even after the instance has been successfully provisioned, it may take a minute or so before the SSH connection can succeed and you may get unable to connect messages during this time.

- b. When you are prompted to continue connecting, enter `yes`.

Note: You are now logged into the new instance.

Linux/UNIX/Mac OS:

```
joe — opc@pcnjd-user12:~ — ssh opc@129.213.37.23 — 114x24
joe@ripley~ % ssh opc@129.213.37.23
The authenticity of host '129.213.37.23 (129.213.37.23)' can't be established.
ECDSA key fingerprint is SHA256:U/CT/BZxwseqL9660qFa9S2qdHLxySayQiwgWNBLKNY.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '129.213.37.23' (ECDSA) to the list of known hosts.
Enter passphrase for key '/Users/joe/.ssh/id_rsa':
Last login: Thu Aug  5 15:39:59 2021 from cpe-76-93-146-33.san.res.rr.com
[opc@pcnjd-user12 ~]$
```

Windows:

```
opc@pcnjd-user12:~  
  
C:\Users\JOGREENW>ssh opc@129.213.37.23  
The authenticity of host '129.213.37.23 (129.213.37.23)' can't be established.  
ECDSA key fingerprint is SHA256:U/CT/BZxwseqL9660qFa9S2qdHLxysayQiwgWNBLKNY.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '129.213.37.23' (ECDSA) to the list of known hosts.  
Enter passphrase for key 'C:\Users\JOGREENW/.ssh/id_rsa':  
Last login: Thu Aug 5 15:39:52 2021  
[opc@pcnjd-user12 ~]$
```

4. In order to set up your new instance, you need to carry out some housekeeping and set up steps.

- a. Enter `sudo yum -y update --skip-broken`

```
opc@pcnjd-user12:~  
[opc@pcnjd-user12 ~]$ sudo yum -y update --skip-broken
```

Note: This will take about three minutes to update the repositories.

- b. Upgrade OCI. Enter:

```
sudo yum upgrade -y python36-oci-cli
```

- c. Install kubectl. Enter the following command all on one line (Hint: you can copy/paste from this document):

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
```

- d. Once the download is complete, enter, all on one line:

```
sudo install -o root -g root -m 0755 kubectl  
/usr/local/bin/kubectl
```

- e. Test kubectl is installed. Enter:

```
kubectl version
```

*** Note: It is normal to see the server localhost:8080 refused. This will be resolved in a later step.

```
[opc@pcnjd-808557 ~]$ curl -L0 "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload Total Spent   Left Speed
100  154  100  154    0      0  782  0:--:-- --:--:--:--:--:-- 785
100 44.7M  100 44.7M    0      0 438k  0:01:44  0:01:44 --:--:-- 433k
[opc@pcnjd-808557 ~]$ sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
[opc@pcnjd-808557 ~]$ kubectl version
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.1", GitCommit:"632ed300f2c34f6d6d15ca4cef3d3c7073412212", GitTreeState:"clean", BuildDate:"2021-08-19T15:45:37Z", GoVersion:"go1.16.7", Compiler:"gc", Platform:"linux/amd64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
[opc@pcnjd-808557 ~]$
```

f. Ensure docker will run for this user. Enter:

```
sudo usermod -a -G docker $USER
```

g. Create the vncpassword for your user. Enter:

```
vncpasswd
```

Note:

- Give a password of your choice that is easy to remember. “oracle” is a good choice.
- When prompted to enter a view only password enter n.

h. Add this password to your configuration text file that contains your public IP address that you previously saved.

```
[opc@pcnjd-user01 ~]$ vncpasswd
Password:
Verify:
Would you like to enter a view-only password (y/n)? n
A view-only password is not used
[opc@pcnjd-user01 ~]$
```

i. Reboot the machine for the changes to take effect. Enter:

```
sudo reboot
```

```
[opc@hol-test ~]$ sudo usermod -a -G docker $USER
[opc@hol-test ~]$ sudo reboot
```

Your SSH connection will be closed.

j. After about one minute, reconnect using the SSH tunneling command:
ssh -L 5901:localhost:5901 opc@<Public IP Address>

k. Once connected, start a new VNC Server session by entering vncserver.

```

A view-only password is not used
[opc@pcnjd ~]$ sudo reboot
Connection to 129.213.50.9 closed by remote host.
Connection to 129.213.50.9 closed.
[joe@ripley ~ % ssh -L 5901:localhost:5901 opc@129.213.50.9
Enter passphrase for key '/Users/joe/.ssh/id_rsa':
Last login: Tue Aug 24 19:30:12 2021
[[opc@pcnjd ~]$ vncserver
xauth:  file /home/opc/.Xauthority does not exist

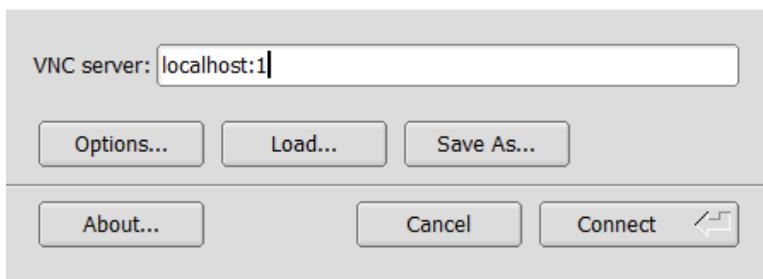
New 'pcnjd:1 (opc)' desktop is pcnjd:1

Creating default startup script /home/opc/.vnc/xstartup
Creating default config /home/opc/.vnc/config
Starting applications specified in /home/opc/.vnc/xstartup
Log file is /home/opc/.vnc/pcnjd:1.log

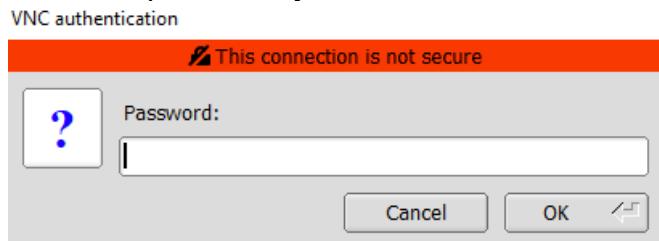
```

5. Connect to your compute instance using your VNC client.
 - a. After successfully tunneling via SSH, start your VNC client and enter localhost:1 or localhost:5901 (it works either way).

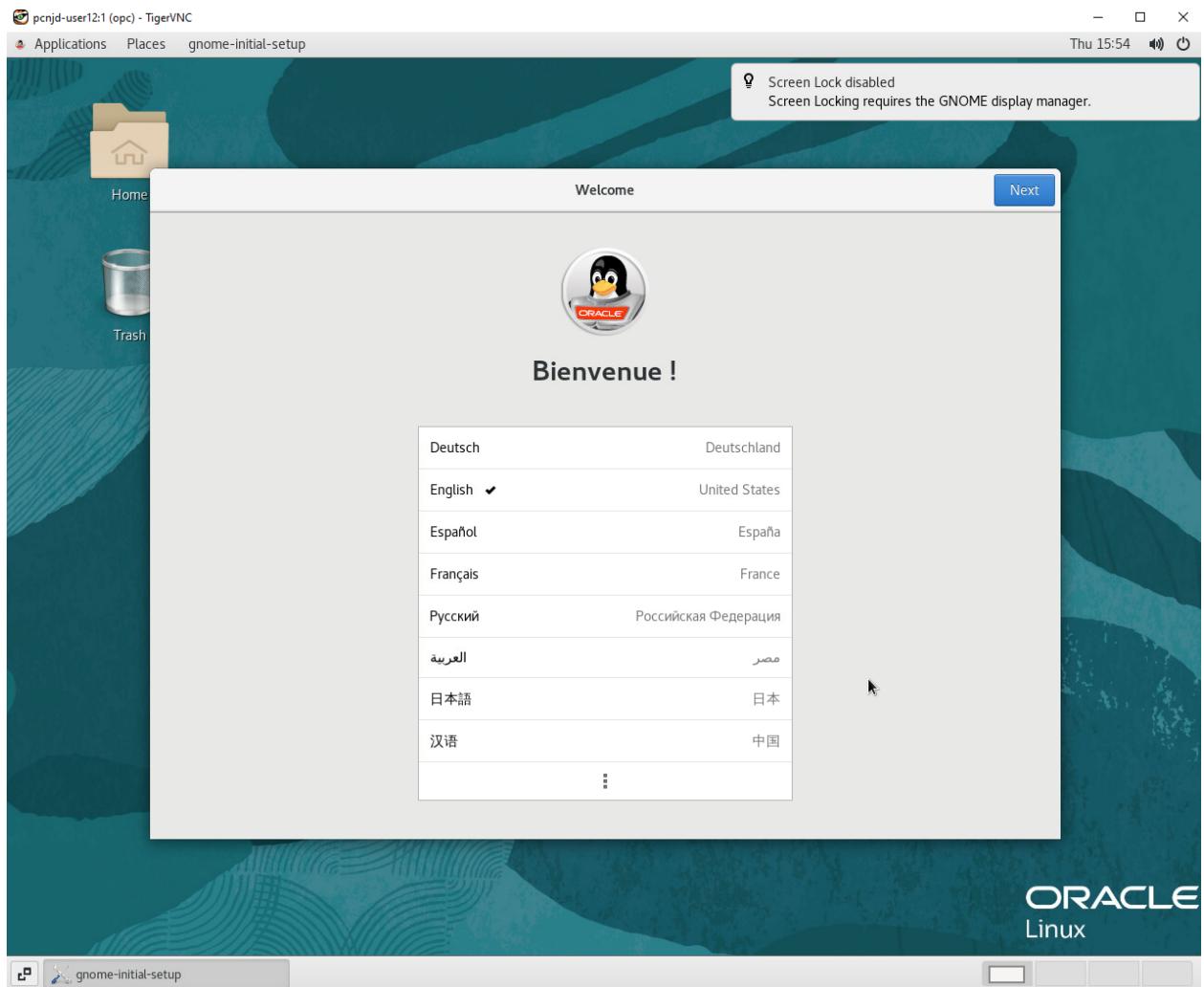
VNC Viewer: Connection Details



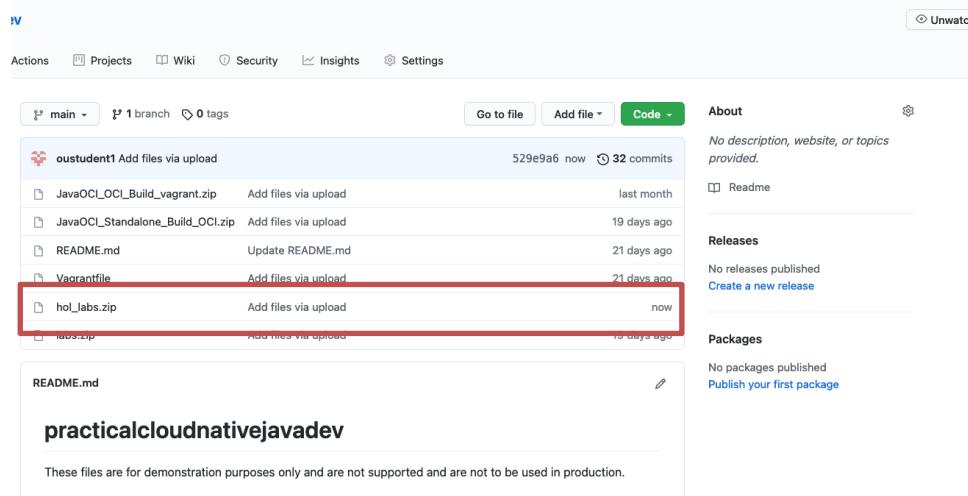
- b. Enter the password you've created earlier with the vncpasswd command.



6. You are presented with the GUI screen for your compute instance.
 - a. On the **Welcome** screens, select your preferred country and language options; use the **Next** button to proceed through this setup wizard.
 - b. Then, close the **Getting Started Gnome Help** window.



7. After you have re-established a VNC connection to the remote host, continue environment setup by downloading and unzipping the Lab files.
- On the remote compute host, open the Firefox browser (**Applications > Favorites > Firefox**) and enter the following URL:
<https://github.com/oustudent1/practicalcloudnativejavadev>

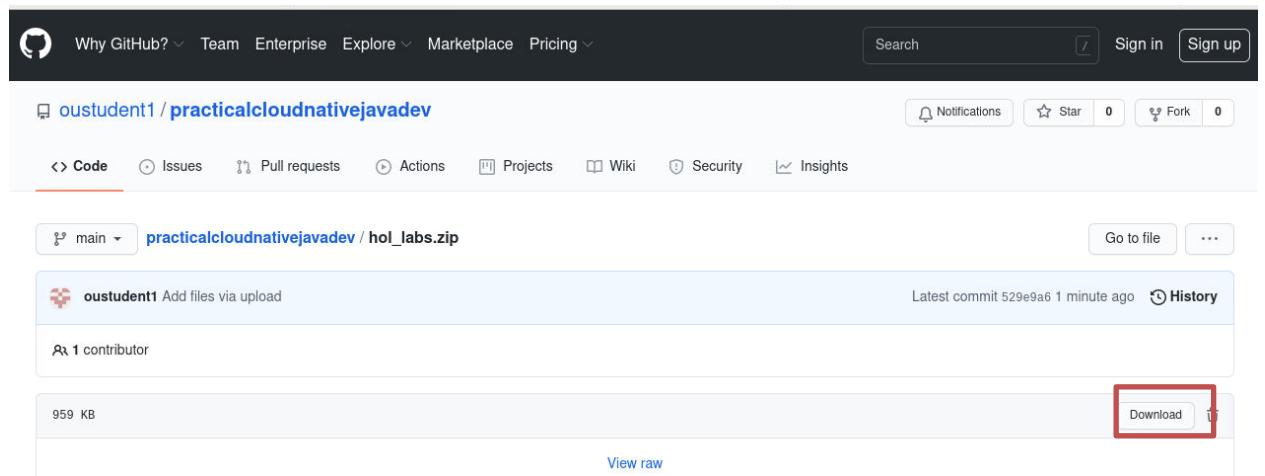


The screenshot shows a GitHub repository page for 'practicalcloudnativejavadev'. The main content area displays a list of files:

- oustudent1 Add files via upload (529e9a6 now) 32 commits
- JavaOCL_OCL_Build_vagrant.zip Add files via upload (last month)
- JavaOCL_Standalone_Build_OCI.zip Add files via upload (19 days ago)
- README.md Update README.md (21 days ago)
- Vagrantfile Add files via upload (21 days ago)
- hol_labs.zip Add files via upload (now)** (highlighted with a red box)
- lab2.zip Add files via upload (19 days ago)

Below the file list, there is a 'README.md' section and a heading 'practicalcloudnativejavadev'. A note at the bottom states: 'These files are for demonstration purposes only and are not supported and are not to be used in production.'

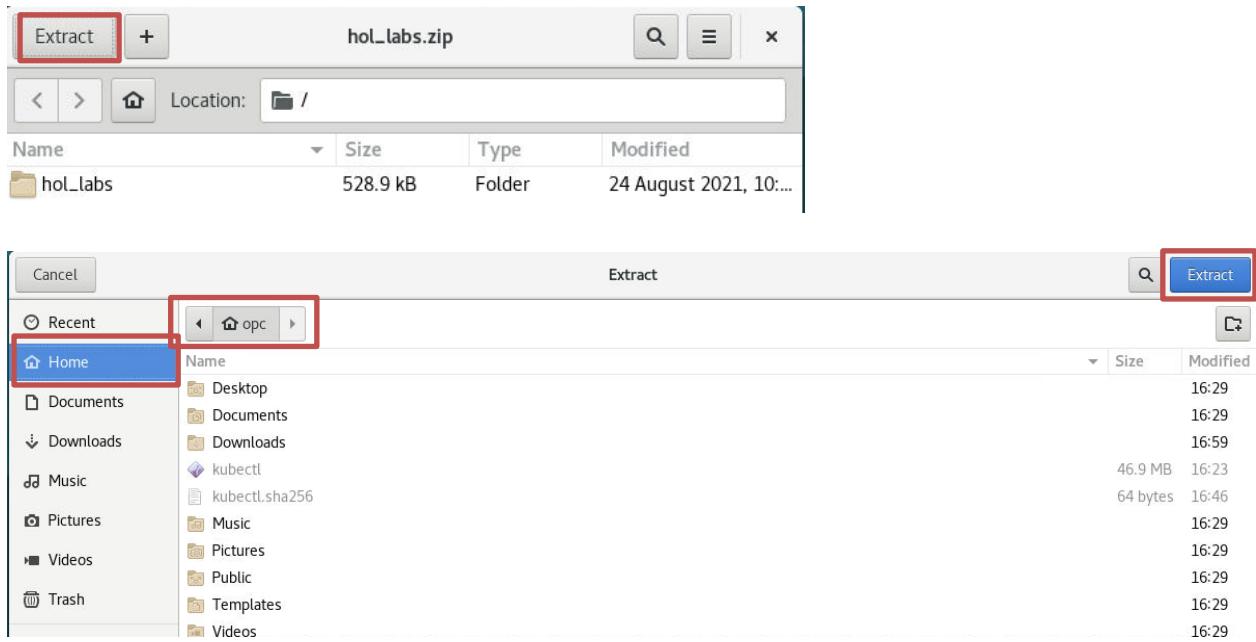
- Click the **hol_labs.zip** link and then click the **Download** button.



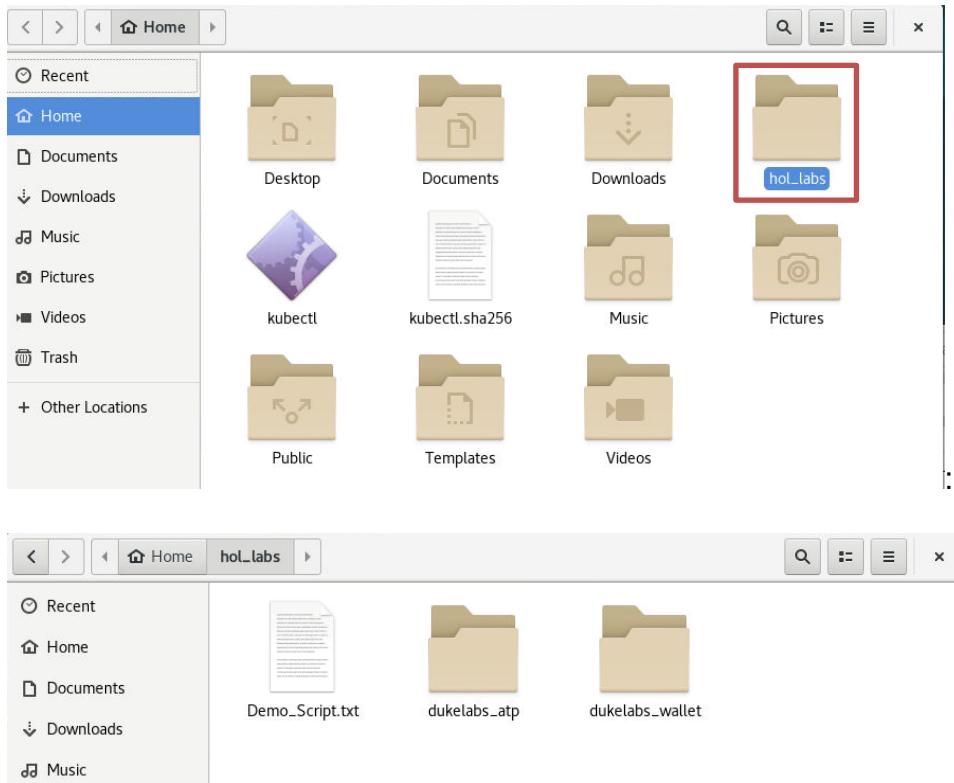
The screenshot shows the GitHub file page for 'hol_labs.zip' in the 'practicalcloudnativejavadev' repository. The page includes the following elements:

- Repository navigation: Why GitHub? Team Enterprise Explore Marketplace Pricing
- Search bar and user links: Search, Sign in, Sign up
- File navigation: Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights
- File details: main branch, practicalcloudnativejavadev / hol_labs.zip, Go to file, ...
- File metadata: oustudent1 Add files via upload, Latest commit 529e9a6 1 minute ago, History
- Contributor information: 1 contributor
- File size: 959 KB
- Action buttons: View raw, Download (highlighted with a red box), ...

- c. Extract the labs folder to **Home / opc**. When prompted, select **Show the Files**.



- d. Double-click the **hol_labs** folder and verify the contents



This completes this practice.

Part II: Provision Oracle Container for Kubernetes (OKE)

Overview

In this practice, you will provision the Oracle Container for Kubernetes (OKE) and capture the OCI Registry (OCIR) namespace details. You also create an Auth Token to enable access to OCI Registry.

Tasks

8. Access Oracle Cloud Console.
 - a. From within your remote Linux compute instance environment, if not already open, launch the Firefox browser (**Applications > Favorites > Firefox**).
 - b. Log into the Oracle Cloud using the user and password you used in Part I of this practice. Save the password when prompted. You may also want to bookmark this page (CTL-D).
-

The screenshot shows the Oracle Cloud Account Sign In page. At the top, it displays "ORACLE Cloud" and the user name "joegreenwald84". Below that is the "Oracle Cloud Account Sign In" button. The main form has two fields: "User Name" with placeholder text "User name or email" and a green "Edit" icon, and "Password" with placeholder text "Password". At the bottom is a large blue "Sign In" button. Below the button, a link says "Need help signing in? [Click here](#)".

9. Setup Kubernetes Cluster.

- From the Navigation Menu, select **Developer Services** and then **Kubernetes Clusters (OKE)**.

The screenshot shows the Oracle Cloud interface with the 'Developer Services' section selected. The 'Containers & Artifacts' category is also highlighted with a red box. The 'Kubernetes Clusters (OKE)' option is specifically highlighted with a red box.

- Home
- Compute
- Storage
- Networking
- Oracle Database
- Databases
- Analytics & AI
- Developer Services** (highlighted with a red box)
- Identity & Security
- Observability & Management

Containers & Artifacts

- Kubernetes Clusters (OKE)** (highlighted with a red box)
- Container Registry
- Artifact Registry

Functions

- Applications

APEX Application Development

- APEX Instances

API Management

- Gateways
- APIs

- You are presented with the **Containers & Artifacts** page. Make sure you are in the same compartment as was used in Part I of this practice.

The screenshot shows the 'Containers & Artifacts' page with the 'Clusters' section. A red box highlights the 'Compartment' dropdown menu, which shows 'joegreenwald84 (root)' selected.

Clusters in joegreenwald84 (root) Compartment

Name	Status	Node Pools <small>(i)</small>	VCN	Version	Created
No clusters exist. Create one to get started.					

Showing 0 Items < 1 of 1 >

- Once the correct compartment is selected, click the **Create Cluster** button.

d. Click the **Launch Workflow** button to start the **Quick Create** workflow.

Create Cluster Help

Quick Create

Custom Create

Select the Quick Create option to create a new cluster, along with creating new network resources. New network resources include one regional subnet for the Kubernetes API endpoint, one regional subnet for worker nodes, and another regional subnet for load balancers.

New resources include:

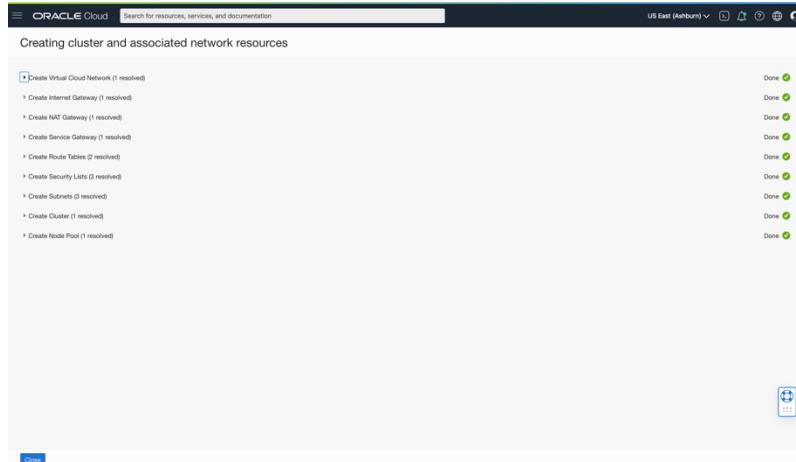
- Virtual Cloud Network (VCN)
- Internet Gateway (IG)
- NAT Gateway (NAT)
- Service Gateway (SGW)
- Kubernetes cluster
- Kubernetes worker nodes(s) and node pool

[Launch Workflow](#) [Cancel](#)

- e. Enter **cluster1** for the Name. Select **VM.Standard2.1** for the **Shape** and change the **Number of nodes** to **1** and click the **Next** button.

The screenshot shows the 'Quick Create Cluster' wizard in Oracle Cloud. The first step, 'Create Cluster', is selected. The 'Cluster' input field contains 'cluster1'. The 'Shape' input field is set to 'VM.Standard2.1'. The 'Number of nodes' input field contains '1'. Other fields like 'Compartment', 'Kubernetes Version', and 'Kubernetes API Endpoint' are also visible.

- f. Click the **Create Cluster** button. Click the **Close** button when provisioning of the OKE cluster is finished.



10. Create an **Auth Token** to be able to log in to the OCI Registry.

- Click the **Profile** icon and select **User Settings**.

The screenshot shows the Oracle Cloud Identity User Details page for a user named joe.greenwald@oracle.com. The profile picture is a green circle with a white letter 'U'. The top navigation bar includes the Oracle Cloud logo, a search bar, and account information (US East (Ashburn)). On the right side, there is a sidebar with options like Profile, Service User Console, and User Settings (which is highlighted with a red box). Below the sidebar is a Sign Out button.

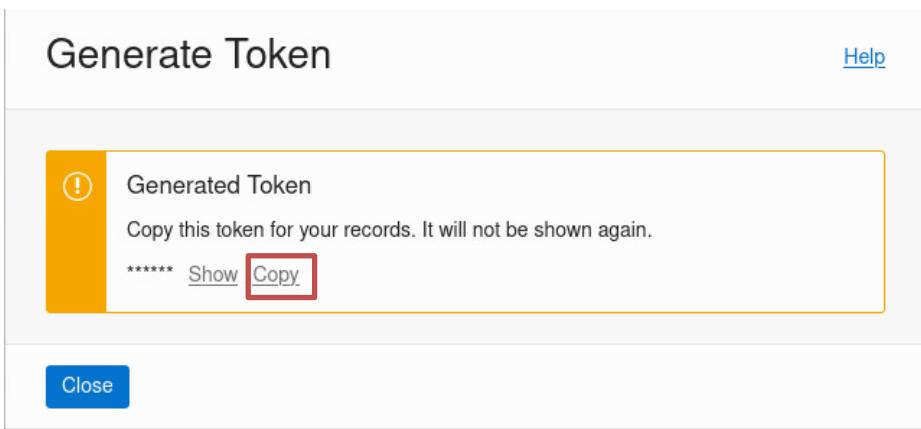
- Scroll down the page and select the **Auth Tokens** link and click the **Generate Token** button.

The screenshot shows the Oracle Cloud Identity Auth Tokens page. On the left, there is a sidebar with links: Groups, API Keys, **Auth Tokens** (which is highlighted with a red box), Customer Secret Keys, OAuth 2.0 Client Credentials, and SMTP Credentials. The main area is titled "Auth Tokens" and contains a table with one row. The table has columns for "Description" and "Created". The "Description" column contains "Lab Auth Token". The "Created" column contains "No items found." At the bottom right of the table is a blue "Generate Token" button with a copy icon. A message "No Token to display" is also visible.

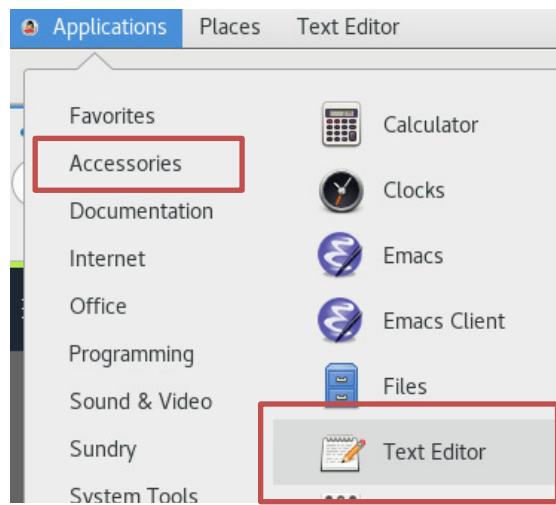
- Enter Lab Auth Token for the name and click the **Generate Token** button.

The screenshot shows a "Generate Token" dialog box. It has a "Description" field containing "Lab Auth Token" (which is highlighted with a blue border). Below the field are "Generate Token" and "Cancel" buttons. At the bottom, there are two status indicators: "Auth tokens: Yes" and "OAuth 2.0 Client Credentials: Yes". There is also a link "View Configuration file" with a help icon.

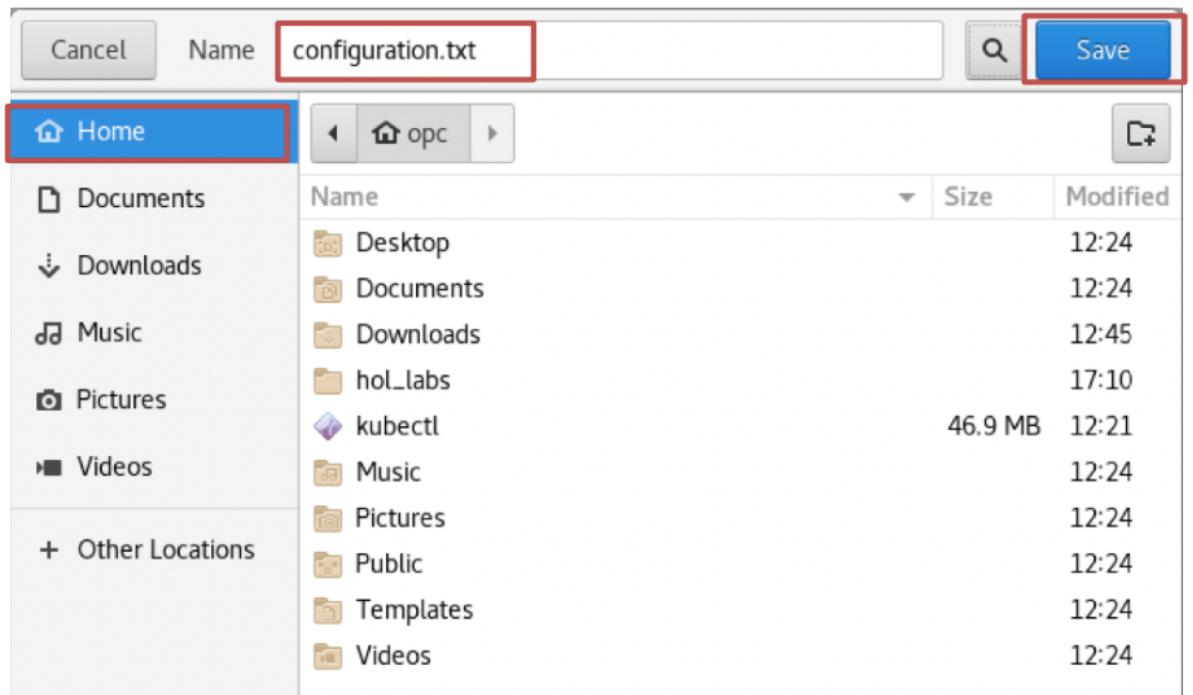
- Click the **Copy** link.



- e. Open a text editor: **Applications > Accessories > Text Editor**.



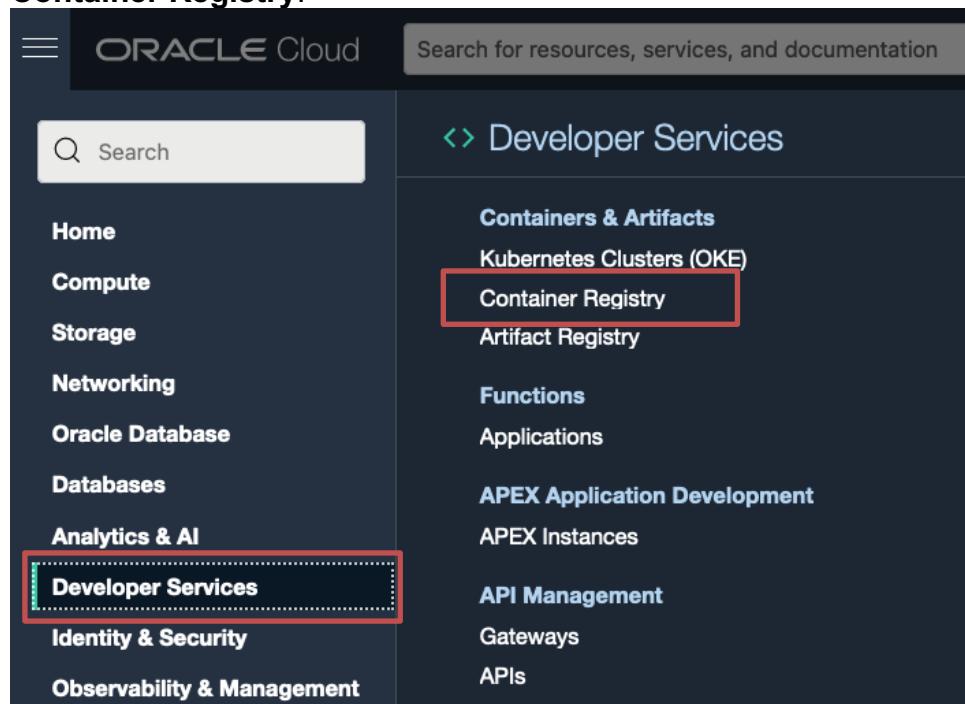
- f. Paste the Auth Token you just copied into the editor and save the file as configuration.txt to /home/opc.



g. Close the **Generate Token** dialog.

11. Find the namespace for the OCI Registry.

- From the Navigation Menu, select **Developer Services** and then **Container Registry**.



- b. Copy and paste the namespace shown into the configuration text file you just saved the Auth Token to.

The screenshot shows the Oracle Cloud Container Registry interface. On the left, there's a sidebar with 'Containers & Artifacts' and sub-options like 'Kubernetes Clusters (OKE)', 'Container Registry' (which is selected and highlighted in blue), and 'Artifact Registry'. Below that are 'List Scope' and 'Image filter' sections. The main area is titled 'Container Registry in joegreenwald84 (root) Compartment'. It shows a 'Create repository' button and a list of repositories under 'joegreenwald84 (root)'. One repository, 'idkxosvste4z', is highlighted with a red box. To its right are statistics: 'Repositories 0', 'Images 0', 'Layers 0', and 'Utilized 0 B'. At the top of the page, there's a search bar and a navigation bar with 'US East (Ashburn)' and other icons.

12. Look up and save your OCI User and Tenancy OCIDs and Region Code.

- a. For **User OCID**, navigate in the OCI Console to the **User Profile** and click the username.

The screenshot shows the Oracle Cloud User Profile page. At the top, there's a navigation bar with 'US East (Ashburn)', a user icon, and other account-related icons. Below it is a 'Profile' section containing the full username 'oracleidentitycloudservice/joe.greenwald@oracle.com', which is highlighted with a red box. Further down are 'Tenancy: joegreenwald84' and a 'Service User Console' link. Underneath is a 'User Settings' section and a 'Sign Out' link.

- b. From the **User Details** page, copy the **full username** and paste it into your configuration text file where you pasted your Auth Token and OCIR namespace.

The screenshot shows the Oracle Cloud Identity User Details page. At the top, there's a navigation bar with 'ORACLE Cloud' and a search bar. Below that, the URL 'oracleidentitycloudservice/joe.greenwald@oracle.com' is displayed. A large green circular icon with a white 'U' is on the left, labeled 'ACTIVE'. On the right, there are tabs for 'User Information' and 'Tags'. Under 'User Information', the OCID '...6ekd7q' is shown with a 'Copy' button, which is highlighted with a red box. Other details include 'Federated: Yes', 'Created: Mon, Aug 23, 2021, 20:15:25 UTC', and 'My Oracle Support account: -'.

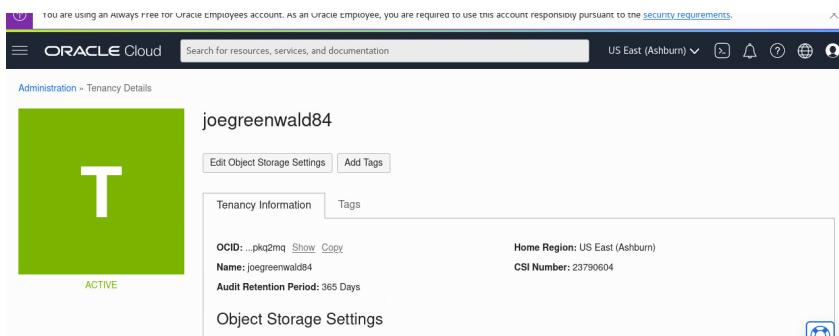
- c. Click the **Copy** link to copy OCID from the **User Details** page.
- d. Paste it into your configuration text file where you pasted your Auth Token and OCIR namespace.

This screenshot is identical to the one above, showing the Oracle Cloud Identity User Details page for the same user. The OCID '...6ekd7q' is again highlighted with a red box in the 'User Information' section.

- e. Get the **Tenancy OCID** and **Region** from the **Tenancy Details** page by navigating to the **User Profile** and selecting the **Tenancy: <your tenancy>** link.

The screenshot shows the Oracle Cloud User Profile page. At the top, there are navigation links for 'US East (Ashburn)', a user icon, and other account-related icons. Below that is a 'Profile' section with the user's email 'oracleidentitycloudservice/joe.greenwald@oracle.com'. A red box highlights the 'Tenancy: joegreenwald84' link. Further down, there are sections for 'Service User Console', 'User Settings', and 'Sign Out'.

- f. Click the **Copy** link and copy the OCID value and paste it into your configuration text file where you pasted your Auth Token, OCIR namespace and user values. Save the file.



The screenshot shows the Oracle Cloud Administration interface. At the top, there's a banner stating, "You are using an Always Free or Oracle employee account. As an Oracle employee, you are required to use this account responsibly pursuant to the [security requirements](#)." Below the banner is the Oracle Cloud logo and a search bar. The main content area is titled "Administration > Tenancy Details". On the left, there's a large green square icon with a white letter "T" and the word "ACTIVE" below it. The tenant name "joegreenwald84" is displayed prominently. There are two buttons at the top: "Edit Object Storage Settings" and "Add Tags". Below these are two tabs: "Tenancy Information" (selected) and "Tags". Under "Tenancy Information", the OCID is listed as "...pkq2mq" with "Show" and "Copy" links. Other details shown include "Name: joegreenwald84", "Home Region: US East (Ashburn)", "CSI Number: 23790604", and "Audit Retention Period: 365 Days". At the bottom, there's a "Object Storage Settings" section and a small "Help" icon.

- g. **Region code** is the code for the OCI Registry region. **iad** is the region code of **us-ashburn-1**. If your region is not us-ashburn-1, then you can find the list of all Region Codes here:
[#Available](https://docs.cloud.oracle.com/iaas/Content/Registry/Concepts/registryprerequisites.htm)

For example:

US East (Ashburn)	<ul style="list-style-type: none">• <code>https://us-ashburn-1.ocir.io</code>• <code>https://iad.ocir.io</code>
US West (Phoenix)	<ul style="list-style-type: none">• <code>https://us-phoenix-1.ocir.io</code>• <code>https://phx.ocir.io</code>
US West (San Jose)	<ul style="list-style-type: none">• <code>https://us-sanjose-1.ocir.io</code>• <code>https://sjc.ocir.io</code>

- h. Save your configuration text file.

This completes this practice.