INDEX

Practical No 1

Aim: Define a simple services like Converting Rs into Dollar and Call it from different platform like JAVA and .NET

Solution:

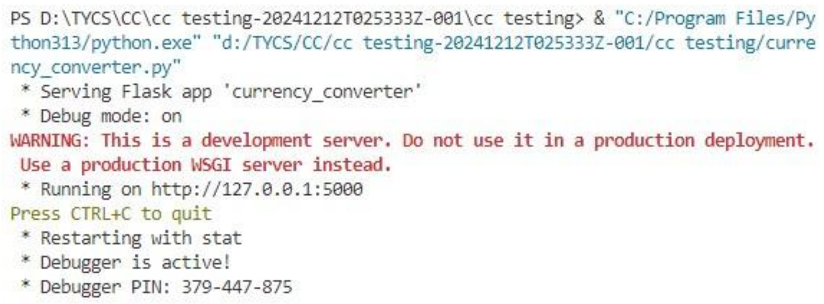
currency_converter.py

```
from flask import Flask, request, jsonify
app = Flask(__name__)
# Conversion rate: 1 INR = 0.012 USD (example rate)
conversion_rate = 0.012
@app.route('/convert', methods=['GET'])
def convert_currency():
    inr = request.args.get('inr')
    if inr:
        try:
            inr_value = float(inr)
            usd_value = inr_value * conversion_rate return
            jsonify({"INR": inr_value, "USD": usd_value})
        except ValueError:
            return jsonify({"error": "Invalid INR amount"}), 400
    return jsonify({"error": "INR amount missing"}), 400
if __name__ == '__main__':
    app.run(debug=True)
```

CurrencyConverterClient.java

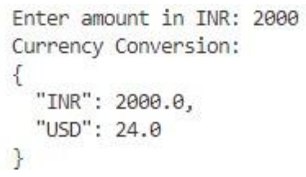
```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.Scanner;
public class CurrencyConverterClient {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter amount in INR: "); String
        inrAmount = scanner.nextLine(); // User input
        try {
            String urlString = "http://127.0.0.1:5000/convert?inr=" + inrAmount;
            URL url = new URL(urlString);
            HttpURLConnection conn = (HttpURLConnection) url.openConnection();
            conn.setRequestMethod("GET");
            conn.setRequestProperty("Accept", "application/json");
            if (conn.getResponseCode() != 200) { throw new
               RuntimeException("Failed : HTTP error code : "
                     + conn.getResponseCode());
            }
            BufferedReader br = new BufferedReader(new InputStreamReader(
               (conn.getInputStream())));
            String output;
            System.out.println("Currency Conversion:");
            while ((output = br.readLine()) != null) {
            System.out.println(output);
            }
            conn.disconnect();
         } catch (Exception e) {
            e.printStackTrace();
         }
      }
   }
```

Output:



```
PS D:\TYCS\CC\cc testing-20241212T025333Z-001\cc testing> & "C:/Program Files/Py
thon313/python.exe" "d:/TYCS/CC/cc testing-20241212T025333Z-001/cc testing/curre
ncy_converter.py"
 * Serving Flask app 'currency_converter'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment.
 Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 379-447-875
```

Now take new terminal



```
Enter amount in INR: 2000
Currency Conversion:
{
  "INR": 2000.0,
  "USD": 24.0
}
```

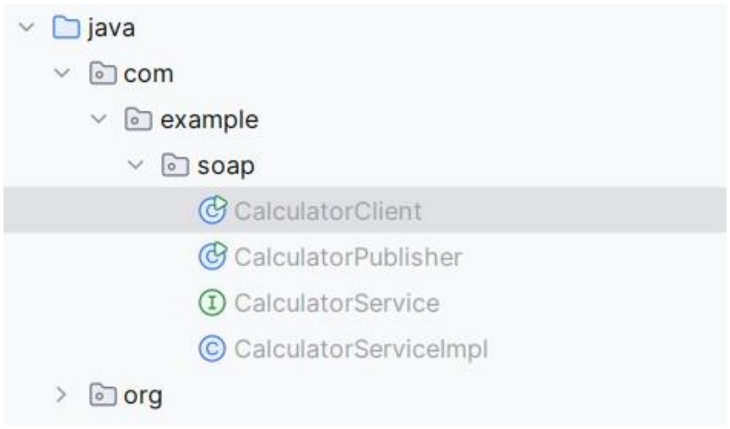
Practical No 2

Aim: Create a Simple SOAP service

Solution:

Step 1: Download JDK 8

Step 2: Create New Project in Intellij Idea & Select Java 8 & Maven Structure



CalculatorServiceImpl.java

```
package com.example.soap;

import javax.jws.WebService;

@WebService(endpointInterface =
"com.example.soap.CalculatorService") public class
CalculatorServiceImpl implements CalculatorService {

   @Override    public int
add(int num1, int num2) {
return num1 + num2;
   }
}
```

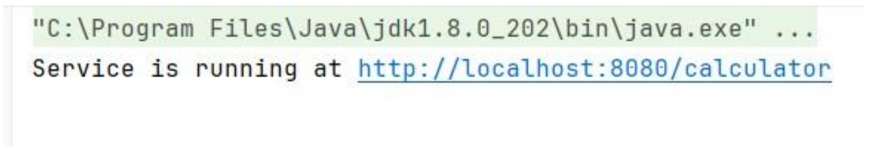
CalculatorService.java

```
package
com.example.soap;

import javax.jws.WebService;

@WebService
```

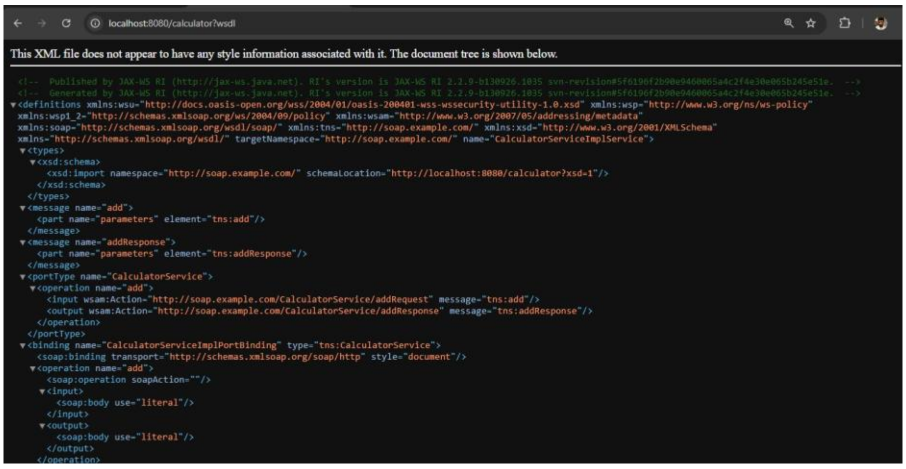
```
public interface
CalculatorService {    public
int add(int num1, int num2);
}
```

**CalculatorPublisher.ja
va**

```
package
com.example.soap;

import javax.xml.ws.Endpoint;

public class CalculatorPublisher {

  public static void main(String[] args) {
      // Publishing the service at a specific URL
      Endpoint.publish("http://localhost:8080/calculator", new
CalculatorServiceImpl());
      System.out.println("Service is running at http://localhost:8080/calculator");
  }
}
```



```
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Service is running at http://localhost:8080/calculator
```



localhost:8080/calculator?wsdl

This XML file does not appear to have any style information associated with it. The document tree is shown below.

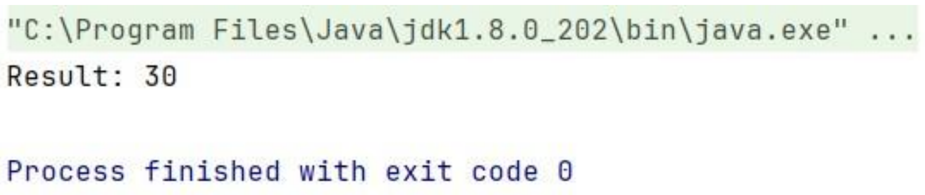
```
<!--  Published by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.9-b130926.1035 svn-revision#5f6196f2b90e9460065a4c2f4e30e065b245e51e.  -->
<!--  Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.9-b130926.1035 svn-revision#5f6196f2b90e9460065a4c2f4e30e065b245e51e.  -->
▼<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp="http://www.w3.org/ns/ws-policy"
 xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
 xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://soap.example.com/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://soap.example.com/" name="CalculatorServiceImplService">
  ▼<types>
    ▼<xsd:schema>
       <xsd:import namespace="http://soap.example.com/" schemaLocation="http://localhost:8080/calculator?xsd=1"/>
     </xsd:schema>
   </types>
  ▼<message name="add">
     <part name="parameters" element="tns:add"/>
   </message>
  ▼<message name="addResponse">
     <part name="parameters" element="tns:addResponse"/>
   </message>
  ▼<portType name="CalculatorService">
    ▼<operation name="add">
       <input wsam:Action="http://soap.example.com/CalculatorService/addRequest" message="tns:add"/>
       <output wsam:Action="http://soap.example.com/CalculatorService/addResponse" message="tns:addResponse"/>
     </operation>
   </portType>
  ▼<binding name="CalculatorServiceImplPortBinding" type="tns:CalculatorService">
     <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
    ▼<operation name="add">
       <soap:operation soapAction=""/>
      ▼<input>
         <soap:body use="literal"/>
       </input>
      ▼<output>
         <soap:body use="literal"/>
       </output>
     </operation>
```

CalculatorClient.java

```
package
com.example.soap;
```

```
import
javax.xml.namespace.QNa
me; import
javax.xml.ws.Service;
import java.net.URL;

public class CalculatorClient {

  public static void main(String[] args) throws Exception {
     // URL to the WSDL
     URL url = new URL("http://localhost:8080/calculator?wsdl");

     // Correct QName based on the WSDL (Check the actual service name)
     QName qname = new QName("http://soap.example.com/",
"CalculatorServiceImplService");

     // Creating service instance
     Service service = Service.create(url, qname);

     // Getting the port and invoking the method
     CalculatorService calculator =
service.getPort(CalculatorService.class);

     int result = calculator.add(10, 20);
     System.out.println("Result: " + result);
  }
}
```

Output:



```
"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" ...
Result: 30

Process finished with exit code 0
```

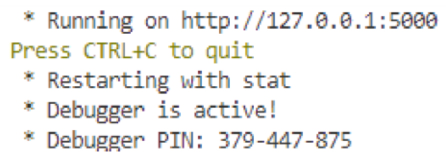
Practical No 3

Aim: Create a Simple REST
Service.

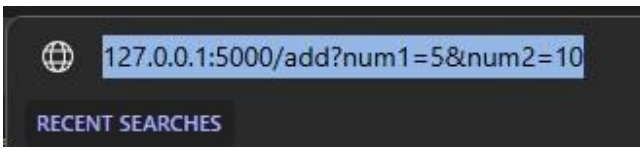
Solution:

add_numbers.py

```
from flask import Flask, request, jsonify
app = Flask(__name__)
# Route to add two numbers

@app.route('/add',
methods=['GET']) def
add_numbers():
    # Get numbers from query parameters
    num1 =
    float(request.args.get('num1')) num2
    = float(request.args.get('num2'))
    # Calculate the
    sum result = num1
    + num2
    # Return the result as JSON
    return jsonify({"result": result})
```



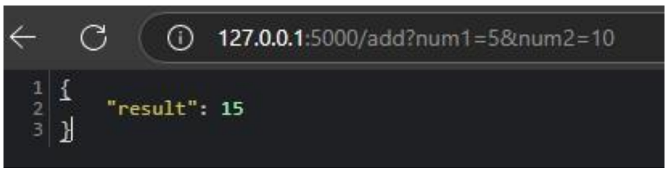
```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 379-447-875
```



127.0.0.1:5000/add?num1=5&num2=10

```
if __name__ == '__main__':
   app.run(debug=True)
```

Output:



127.0.0.1:5000/add?num1=5&num2=10

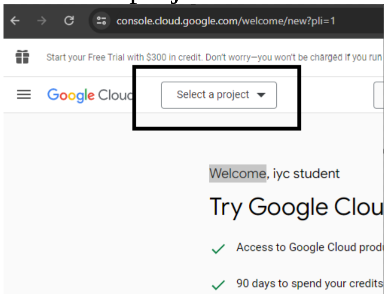
```
{
    "result": 15
}
```

Practical No 4

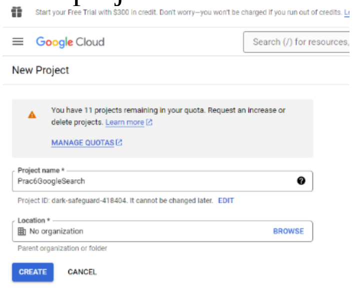
Aim:
Develop an application to consume Google's search / Google's Map RESTful Web service.

Solution:

Step 1: Search "console cloud google" on chrome and sign up
https://console.cloud.google.com
Click on project and create new project


console.cloud.google.com/welcome/new?pli=1
Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run
Google Cloud
Select a project
Welcome, iyc student
Try Google Clou
Access to Google Cloud prod
90 days to spend your credits

Give project name and then click create


Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits.

New Project

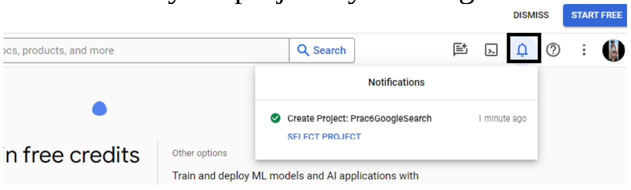
You have 11 projects remaining in your quota. Request an increase or delete projects. Learn more

MANAGE QUOTAS

Project name *
Prac6GoogleSearch
Project ID: dark-safeguard-418404. It cannot be changed later. EDIT

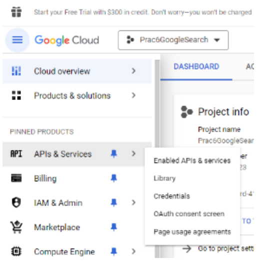
Location *
No organization
Parent organization or folder

You can see your project by clicking bell icon

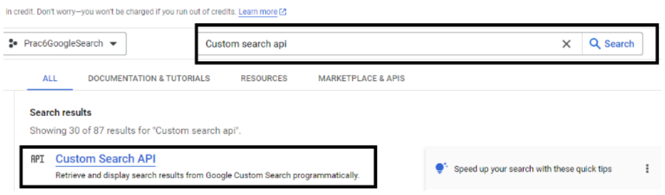

Notifications

Create Project: Prac6GoogleSearch 1 minute ago

Step 2: Click on bell icon and click "select project"
Once you select the project then click on top-left burger icon → Click "APIs and Services" → Enabled APIs and services



Search "Custom search api" on search bar and click on first link


in credit. Don't worry—you won't be charged if you run out of credits. Learn more

Prac6GoogleSearch | Custom search api | Search

ALL | DOCUMENTATION & TUTORIALS | RESOURCES | MARKETPLACE & APIS

Search results

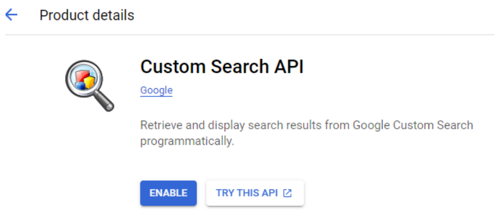
Showing 30 of 87 results for "Custom search api".

API Custom Search API

Retrieve and display search results from Google Custom Search programmatically.

Speed up your search with these quick tips

In google cloud project you always have to specify which functionality you want to enable.
Click on enable to enable custom search api
This enable the custom search api for this particular project


Product details

Custom Search API

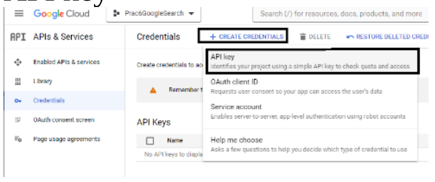
Google

Retrieve and display search results from Google Custom Search programmatically.

ENABLE | TRY THIS API

Step 3: Now you need to generate an api key to authenticate yourself from python.

In the Api & Services section click on credentials→ click create credentials→Click API key


Google Cloud | Prac6GoogleSearch | Search (/) for resources, docs, products, and more

APIs & Services | Credentials | + CREATE CREDENTIALS | DELETE | RESTORE DELETED CREDI

Enabled APIs & services
Library
Credentials
OAuth consent screen
Page usage agreements

Create credentials to ac

Remember t

API Keys

Name

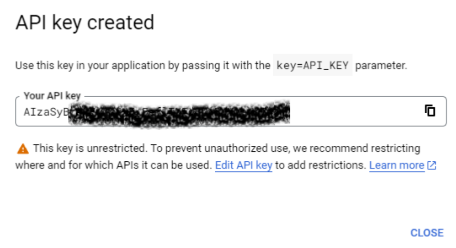
No API keys to displa

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use


API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

Your API key

AIzaSyB[redacted]

This key is unrestricted. To prevent unauthorized use, we recommend restricting where and for which APIs it can be used. Edit API key to add restrictions. Learn more

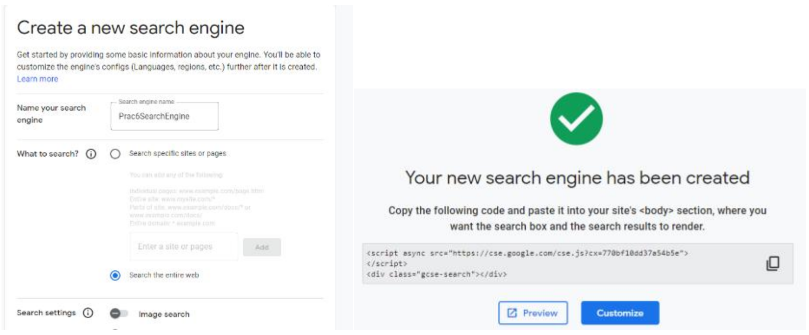
CLOSE

Create a file as API_KEY in your project and paste api_key value in it

Step 4: Create a search engine

Go to google search programmable search engine → Open first website and click get started

Give name to your search engine and select the "search the entire website " and click create


Create a new search engine

Get started by providing some basic information about your engine. You'll be able to customize the engine's configs (Languages, regions, etc.) further after it is created. Learn more

Name your search engine: Search engine name: Prac6SearchEngine

What to search? ( ) Search specific sites or pages (●) Search the entire web

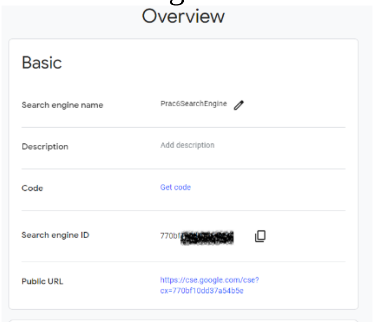
Search settings: Image search

Your new search engine has been created

Copy the following code and paste it into your site's <body> section, where you want the search box and the search results to render.

```
<script async src="https://cse.google.com/cse.js?cx=770bf10dd37a54b5e">
</script>
<div class="gcse-search"></div>
```

Step 5: Click “go back to all search engine” → click on your search engine → Copy the search engine ID


Overview

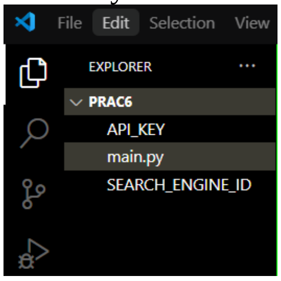
Basic

Search engine name	Prac6SearchEngine
Description	Add description
Code	Get code
Search engine ID	770bf[redacted]
Public URL	https://cse.google.com/cse?cx=770bf10dd37a54b5e

Create a new file on your project as SEARCH_ENGINE_ID and paste your search engine ID

Download requests using pip3 install requests in the command line .

Directory structure:



```
PRAC6
    API_KEY
    main.py
    SEARCH_ENGINE_ID
```

Step 6: Get text results using google search

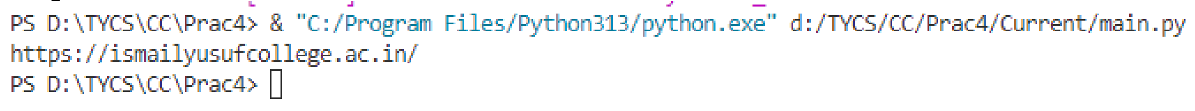
Create a file main.py and write a below code in it

Code:

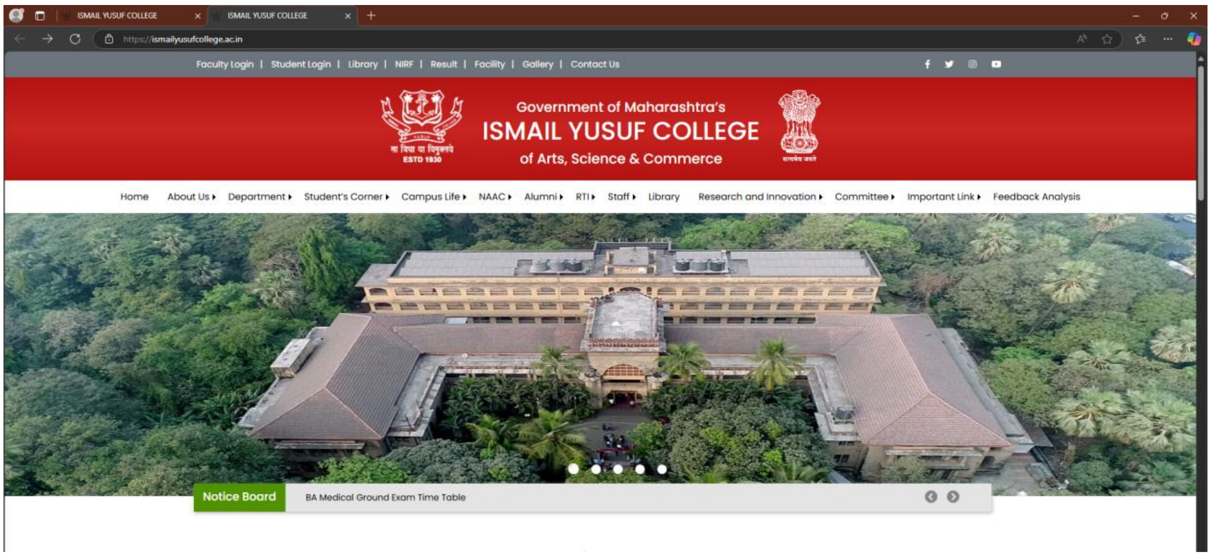
```
import requests

# Replace with your actual API key and Search Engine ID
API_KEY = 'Enter your Api Key'
SEARCH_ENGINE_ID = 'Enter your Engine ID'

search_query = 'ismail yusuf college'
```

```python
url = 'https://www.googleapis.com/customsearch/v1'

params = {
    'q': search_query,
    'key': API_KEY,
    'cx': SEARCH_ENGINE_ID
}

# Send request to the API
response = requests.get(url, params=params)

# Get the response data in JSON format
results = response.json()

# Check if there are results and print the first result's link
if 'items' in results:
    print(results['items'][0]['link'])
else:
    print("No results found.")
```

Output:



```
PS D:\TYCS\CC\Prac4> & "C:/Program Files/Python313/python.exe" d:/TYCS/CC/Prac4/Current/main.py
https://ismailyusufcollege.ac.in/
PS D:\TYCS\CC\Prac4> 
```



Practical No 5

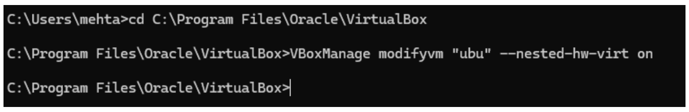
Aim:

Installation and Configuration of virtualization using KVM.

Solution:

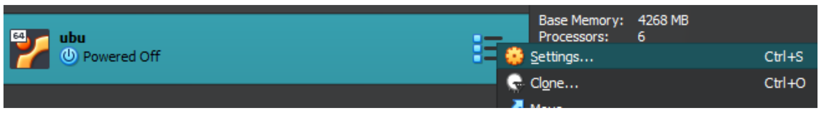
Step 1: Select Ubuntu platform Virtual Machine or Physical Ubuntu Machine recommended: go with Dual Boot

Step 2: In case of Virtual Machine use execute these command
"cd C:\Program Files\Oracle\VirtualBox"
"VBoxManage modifyvm "ubu" –nested—hw-virt on"



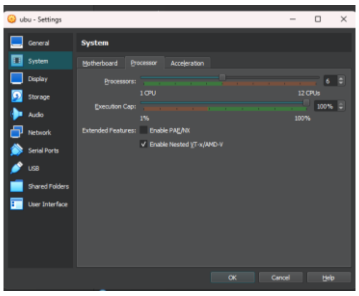
```
C:\Users\mehta>cd C:\Program Files\Oracle\VirtualBox

C:\Program Files\Oracle\VirtualBox>VBoxManage modifyvm "ubu" --nested-hw-virt on

C:\Program Files\Oracle\VirtualBox>
```

Step 3: open settings



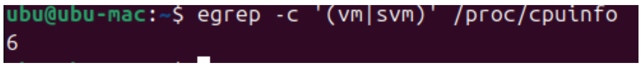
ubu
Powered Off
Base Memory: 4268 MB
Processors: 6

Step 4: Give minimum 6 Processors & Nested VTX Should ON through above command



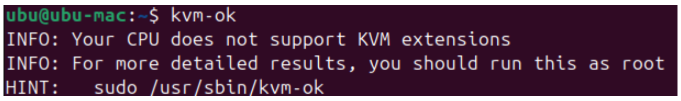
ubu - Settings
System
Processors: 6
1 CPU 12 CPUs
Execution Cap: 100%
1% 100%
Extended Features: Enable PAE/NX
✓ Enable Nested VT-x/AMD-V

Step 5: open terminal & run this command to check processors
$ egrep -c '(vm|svm)' /proc/cpuinfo



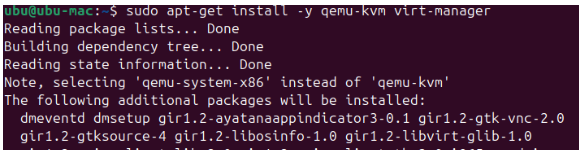
```
ubu@ubu-mac:~$ egrep -c '(vm|svm)' /proc/cpuinfo
6
```

Step 6: $ kvm-ok



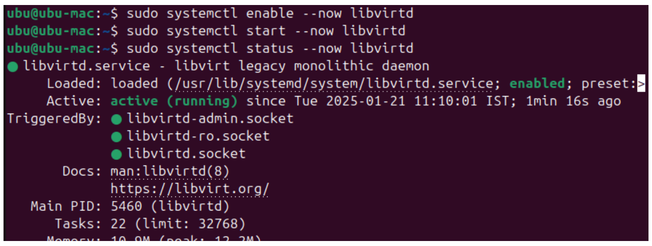
```
ubu@ubu-mac:~$ kvm-ok
INFO: Your CPU does not support KVM extensions
INFO: For more detailed results, you should run this as root
HINT:   sudo /usr/sbin/kvm-ok
```

Step 7: $ sudo apt-get install -y qemu-kvm virt-manager



```
ubu@ubu-mac:~$ sudo apt-get install -y qemu-kvm virt-manager
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'qemu-system-x86' instead of 'qemu-kvm'
The following additional packages will be installed:
  dmeventd dmsetup gir1.2-ayatanaappindicator3-0.1 gir1.2-gtk-vnc-2.0
  gir1.2-gtksource-4 gir1.2-libosinfo-1.0 gir1.2-libvirt-glib-1.0
```

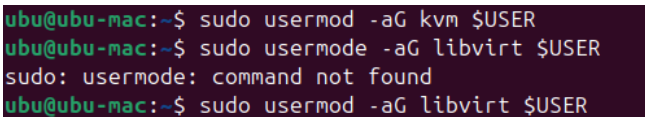
Step 8: $ sudo systemctl enable --now libvirtd
$ sudo systemctl start --now libvirtd
$ sudo systemctl status --now libvirtd



```
ubu@ubu-mac:~$ sudo systemctl enable --now libvirtd
ubu@ubu-mac:~$ sudo systemctl start --now libvirtd
ubu@ubu-mac:~$ sudo systemctl status --now libvirtd
● libvirtd.service - libvirt legacy monolithic daemon
     Loaded: loaded (/usr/lib/systemd/system/libvirtd.service; enabled; preset:>
     Active: active (running) since Tue 2025-01-21 11:10:01 IST; 1min 16s ago
TriggeredBy: ● libvirtd-admin.socket
             ● libvirtd-ro.socket
             ● libvirtd.socket
       Docs: man:libvirtd(8)
             https://libvirt.org/
   Main PID: 5460 (libvirtd)
      Tasks: 22 (limit: 32768)
     Memory: 10.9M (peak: 12.2M)
```

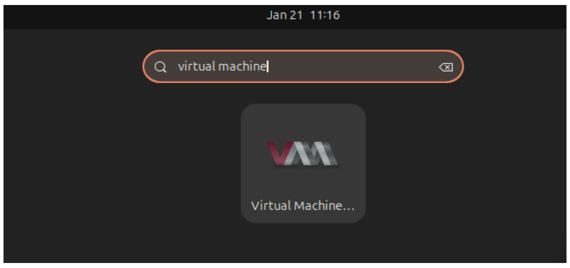
Step 9: $ sudo usermod -aG kvm $USER

$ sudo usermod -aG libvirt $USER

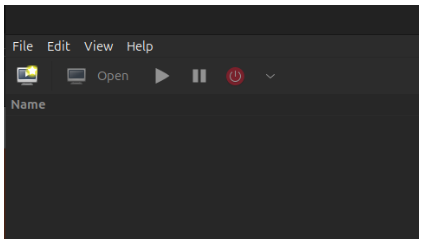


```
ubu@ubu-mac:~$ sudo usermod -aG kvm $USER
ubu@ubu-mac:~$ sudo usermode -aG libvirt $USER
sudo: usermode: command not found
ubu@ubu-mac:~$ sudo usermod -aG libvirt $USER
```

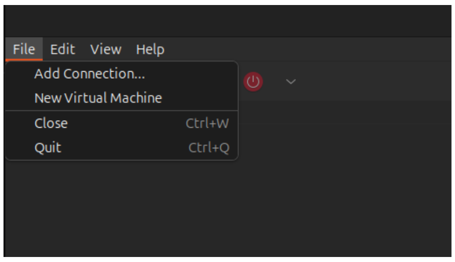
Step 10: Search Virtual Machine and open



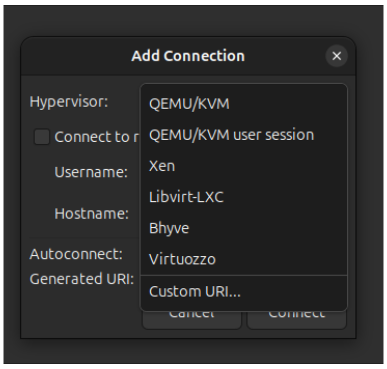
Step 11: Click on File



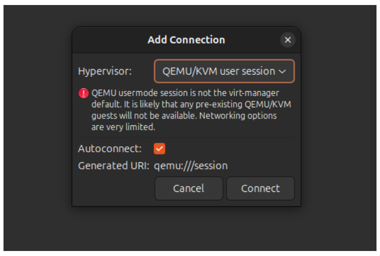
Step 12: Add Connection



Step 13: QEMU/KVM user session



Step 14: click on “Connect” and then yes

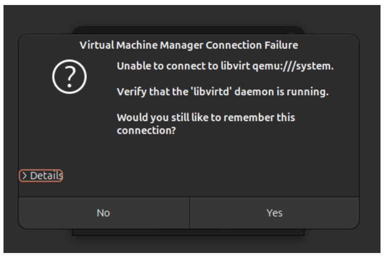

Add Connection

Hypervisor: QEMU/KVM user session

QEMU usermode session is not the virt-manager default. It is likely that any pre-existing QEMU/KVM guests will not be available. Networking options are very limited.

Autoconnect: ✓

Generated URI: qemu:///session

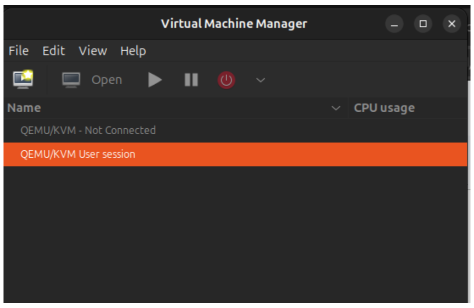

Virtual Machine Manager Connection Failure

Unable to connect to libvirt qemu:///system.

Verify that the 'libvirtd' daemon is running.

Would you still like to remember this connection?

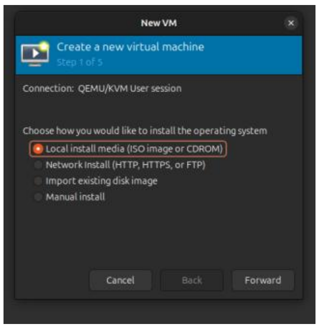
Step 15: now right click on “user session” and create new virtual machine



QEMU/KVM - Not Connected

QEMU/KVM User session

Step 16: Select 1st option and then forward

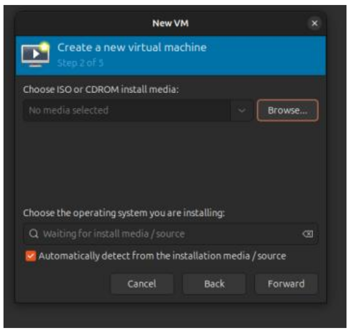

New VM

Create a new virtual machine
Step 1 of 5

Connection: QEMU/KVM User session

Choose how you would like to install the operating system
- ● Local install media (ISO image or CDROM)
- Network Install (HTTP, HTTPS, or FTP)
- Import existing disk image
- Manual install

Step 17: Download ubuntu ISO File 16 & Click on Browse


New VM

Create a new virtual machine
Step 2 of 5

Choose ISO or CDROM install media:

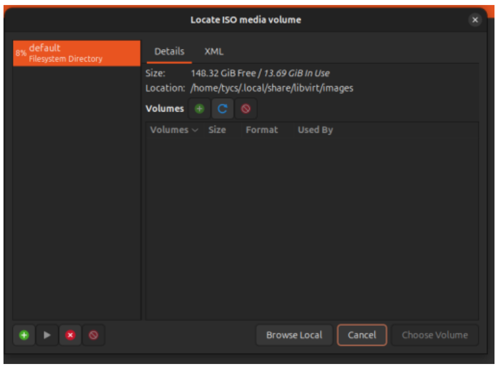
No media selected

Choose the operating system you are installing:

Waiting for install media / source

✓ Automatically detect from the installation media / source

Step 18: Browse local


Locate ISO media volume

8% default
Filesystem Directory

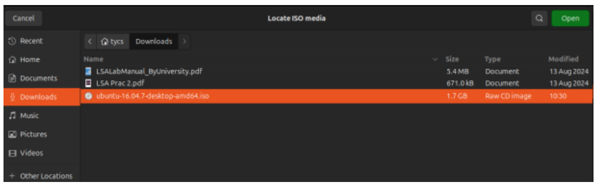
Size: 148.32 GiB Free / 13.69 GiB In Use

Location: /home/tycs/.local/share/libvirt/images

Volumes

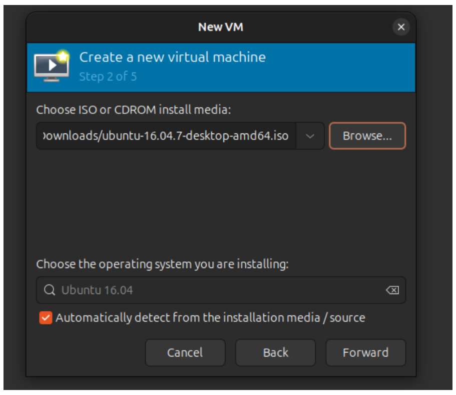
Volumes	Size	Format	Used By

Select ISO file and then forward



Locate ISO media

Name	Size	Type	Modified
LSALabManual_ByUniversity.pdf	5.4 MB	Document	13 Aug 2024
LSA Prac 2.pdf	671.0 kB	Document	13 Aug 2024
ubuntu-16.04.7-desktop-amd64.iso	1.7 GB	Raw CD image	10:30



New VM

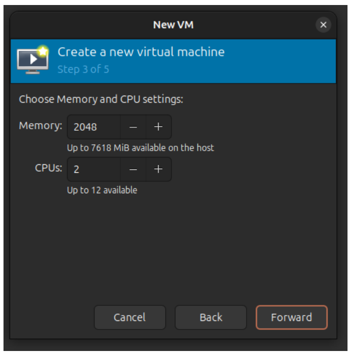
Create a new virtual machine
Step 2 of 5

Choose ISO or CDROM install media: ›ownloads/ubuntu-16.04.7-desktop-amd64.iso

Choose the operating system you are installing: Ubuntu 16.04

[✓] Automatically detect from the installation media / source

Step 18: Allocate Memory & CPUs and then forward



New VM

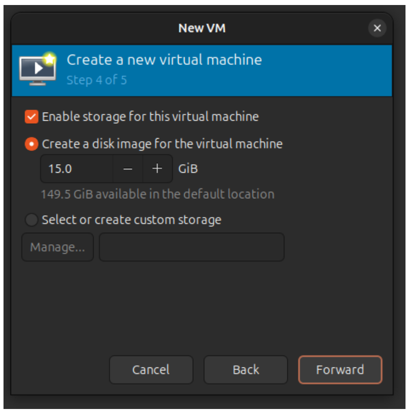
Create a new virtual machine
Step 3 of 5

Choose Memory and CPU settings:

Memory: 2048
Up to 7618 MiB available on the host

CPUs: 2
Up to 12 available

Step 19: Allocate Disk Space & then finish



New VM

Create a new virtual machine
Step 4 of 5

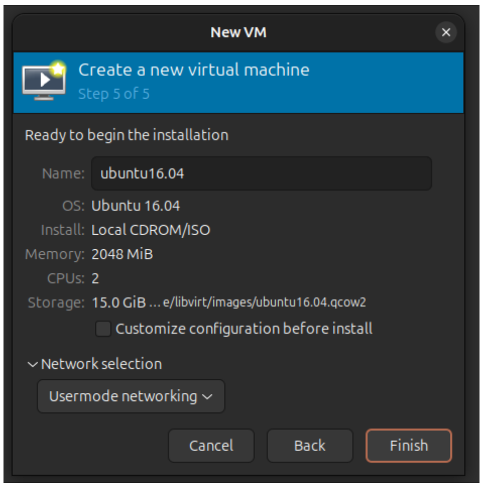
[✓] Enable storage for this virtual machine

(●) Create a disk image for the virtual machine

15.0 GiB

149.5 GiB available in the default location

( ) Select or create custom storage



New VM

Create a new virtual machine
Step 5 of 5

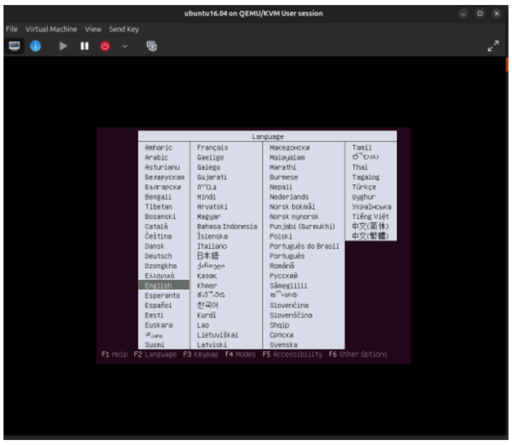
Ready to begin the installation

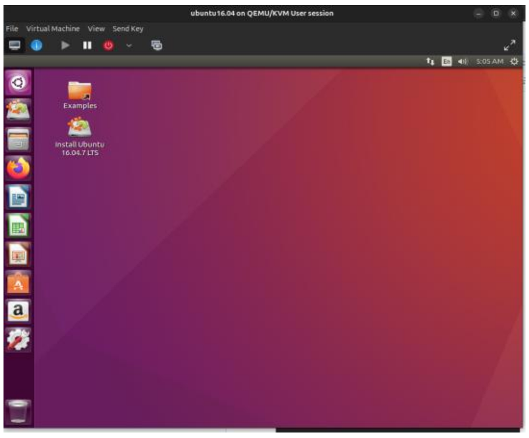
Name: ubuntu16.04
OS: Ubuntu 16.04
Install: Local CDROM/ISO
Memory: 2048 MiB
CPUs: 2
Storage: 15.0 GiB ...e/libvirt/images/ubuntu16.04.qcow2

[ ] Customize configuration before install

Network selection

Usermode networking





ubuntu16.04 on QEMU/KVM User session
File
Virtual Machine
View
Send Key
Examples
Install Ubuntu 16.04.7 LTS

Practical No 6

Aim: Develop application to download image/video from server or upload image/video to server using MTOM techniques.

Solution:

Step 1: Create app.py file & Run

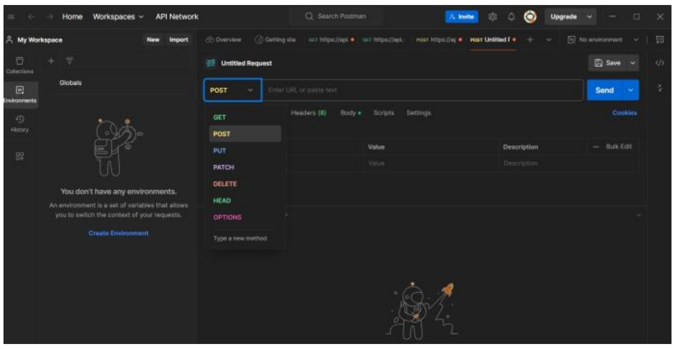
app.py

```
from flask import Flask, request,
send_from_directory import os
app = Flask(__name__)
# Namaste Bacho ! Uploads ke liye folder banane ka
UPLOAD_FOLDER = 'uploads'
os.makedirs(UPLOAD_FOLDER, exist_ok=True)
# 1. File upload endpoint
@app.route('/upload',
methods=['POST']) def
upload_file():
  file =
  request.files.get('file')
  if file:
    file_path = os.path.join(UPLOAD_FOLDER, file.filename)
    file.save(file_path)
    return {"message": f"File '{file.filename}' uploaded successfully!"}, 200
  return {"error": "No file uploaded"}, 400
# 2. File download endpoint
@app.route('/download/<filename>',
methods=['GET']) def
download_file(filename): try:
    return send_from_directory(UPLOAD_FOLDER, filename, as_attachment=True)
  except FileNotFoundError:
    return {"error": "File not
    found"}, 404
if __name__ ==
  '__main__':
  app.run(debug=True)
```

```
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 219-099-923
```

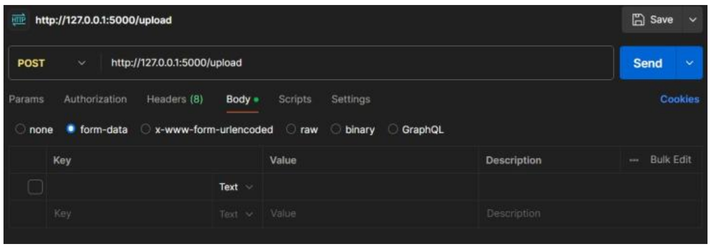
After running this code it will automictically create uploads folder

Step 2: Open postman and select method "POST"



POST

Step 3: Enter url http://127.0.0.1:5000/upload & select Body & then select form-data and replace text With File



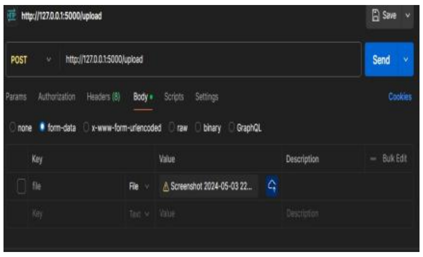
POST http://127.0.0.1:5000/upload

Body: form-data

Key		Value	Description
	Text		

Step 4: Click on Values and + New file from local machine

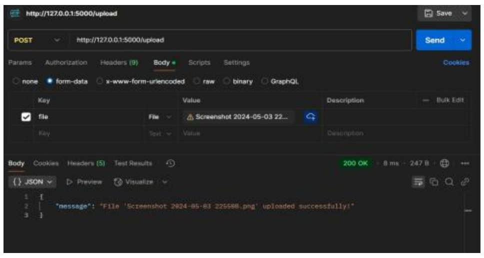
Step 5: After selecting file click on send button



POST http://127.0.0.1:5000/upload

Body: form-data

Key		Value	Description
file	File	Screenshot 2024-05-03 22...	



POST http://127.0.0.1:5000/upload

Body: form-data

Key		Value	Description
file	File	Screenshot 2024-05-03 22...	

200 OK · 8 ms · 247 B

```
{
    "message": "File 'Screenshot 2024-05-03 225508.png' uploaded successfully!"
}
```

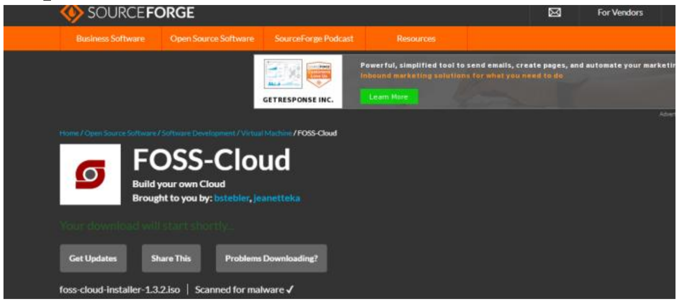
Practical No 7

Aim: Implement FOSS-Cloud Functionality VSI (Virtual Server Infrastructure) Infrastructure as a Service (IaaS), Storage

Solution:

Note: If your windows machine support then You can do this practical in windows otherwise install virtual box in Physical Dual Booted Ubuntu Machine and follow same steps except one step

Step 1: Download Foss Cloud ISO



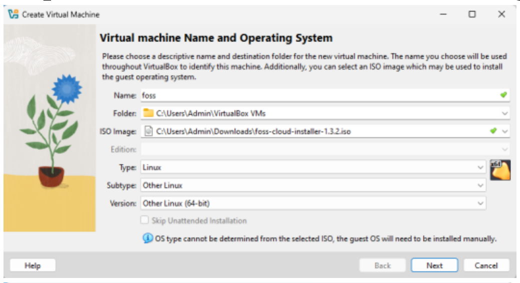
Home / Open Source Software / Software Development / Virtual Machine / FOSS-Cloud

FOSS-Cloud
Build your own Cloud
Brought to you by: bstebler, jeanetteka

Your download will start shortly...

foss-cloud-installer-1.3.2.iso | Scanned for malware ✓

Step 2: Create Virtual with the below configurations


Create Virtual Machine

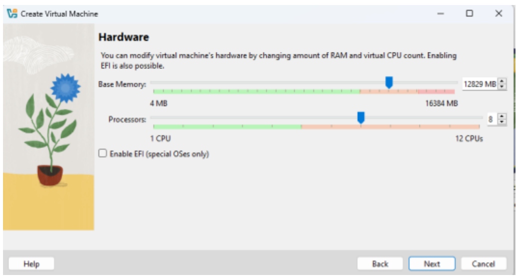
Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name:	foss
Folder:	C:\Users\Admin\VirtualBox VMs
ISO Image:	C:\Users\Admin\Downloads\foss-cloud-installer-1.3.2.iso
Edition:	
Type:	Linux
Subtype:	Other Linux
Version:	Other Linux (64-bit)

☐ Skip Unattended Installation

OS type cannot be determined from the selected ISO, the guest OS will need to be installed manually.


Create Virtual Machine

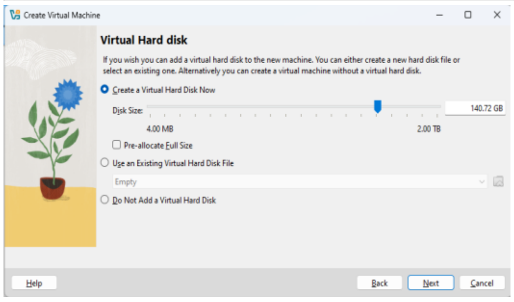
Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 12829 MB (4 MB – 16384 MB)

Processors: 8 (1 CPU – 12 CPUs)

☐ Enable EFI (special OSes only)


Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

[✓ Create a Virtual Hard Disk Now]

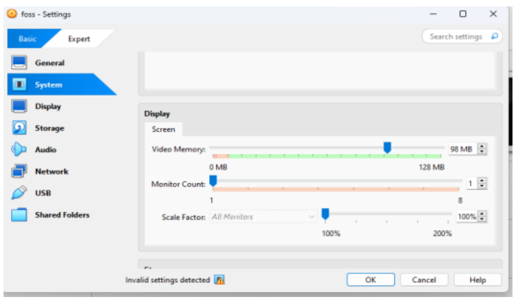
Disk Size: 140.72 GB (4.00 MB – 2.00 TB)

☐ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

Empty

○ Do Not Add a Virtual Hard Disk


foss - Settings

Display

Screen

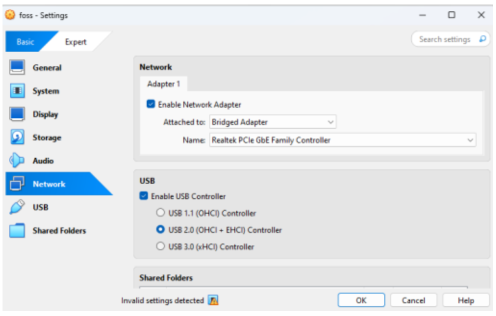
Video Memory: 98 MB (0 MB – 128 MB)

Monitor Count: 1 (1 – 8)

Scale Factor: All Monitors 100% (100% – 200%)

Invalid settings detected

Step 3: Select Bridged Adapter


foss - Settings

Network

Adapter 1

☑ Enable Network Adapter

Attached to: Bridged Adapter

Name: Realtek PCIe GbE Family Controller

USB

☑ Enable USB Controller

○ USB 1.1 (OHCI) Controller

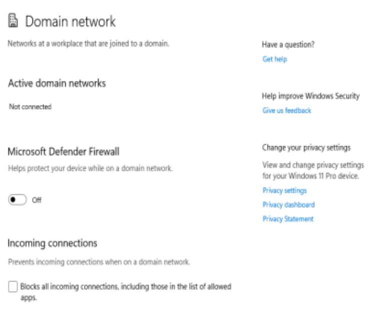
[✓ USB 2.0 (OHCI + EHCI) Controller]

○ USB 3.0 (xHCI) Controller

Shared Folders

Invalid settings detected

Step 4: Turn off firewall


Domain network

Networks at a workplace that are joined to a domain.

Active domain networks

Not connected

Microsoft Defender Firewall

Helps protect your device while on a domain network.

Off

Incoming connections

Prevents incoming connections when on a domain network.

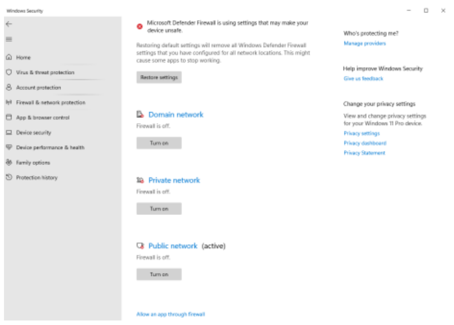
☐ Blocks all incoming connections, including those in the list of allowed apps.

Have a question?
Get help

Help improve Windows Security
Give us feedback

Change your privacy settings

View and change privacy settings for your Windows 11 Pro device.
Privacy settings
Privacy dashboard
Privacy Statement



Microsoft Defender Firewall is using settings that may make your device unsafe.

Restoring default settings will remove all Windows Defender Firewall settings that you have configured for all network locations. This might cause some apps to stop working.

Restore settings

Domain network
Firewall is off.
Turn on

Private network
Firewall is off.
Turn on

Public network (active)
Firewall is off.
Turn on

Allow an app through firewall

Who's protecting me?
Manage providers

Help improve Windows Security
Give us feedback

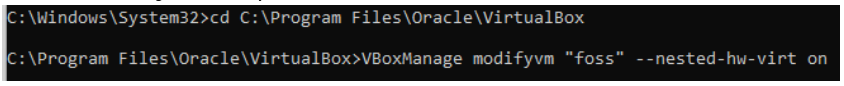
Change your privacy settings
View and change privacy settings for your Windows 11 Pro device.
Privacy settings
Privacy dashboard
Privacy Statement

For Linux: Execute “ufw disable” in root user mode

Step 5: Open Run cmd as administrative then execute the below commands

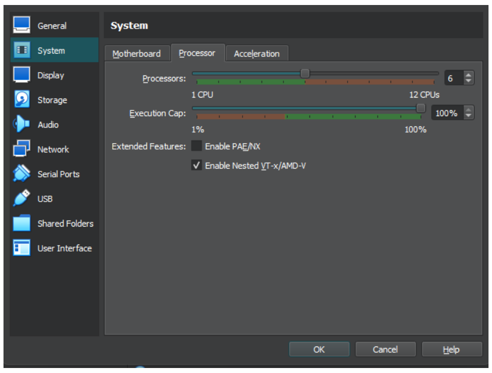
“cd C:\Program Files\Oracle\VirtualBox“

“VBoxManage modifyvm “foss” –nested—hw-virt on”



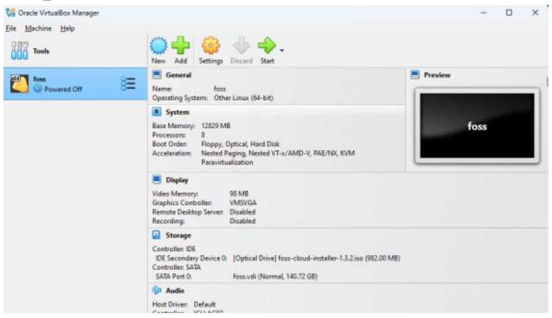
```
C:\Windows\System32>cd C:\Program Files\Oracle\VirtualBox

C:\Program Files\Oracle\VirtualBox>VBoxManage modifyvm "foss" --nested-hw-virt on
```

For Linux: You need to enable it manually by clicking on Enable Nested VT-x/AMD-V



Processors: 6
1 CPU 12 CPUs
Execution Cap: 100%
1% 100%
Extended Features: ☐ Enable PAE/NX
✓ Enable Nested VT-x/AMD-V

Step 6: Start the virtual Machine



General
Name: foss
Operating System: Other Linux (64-bit)

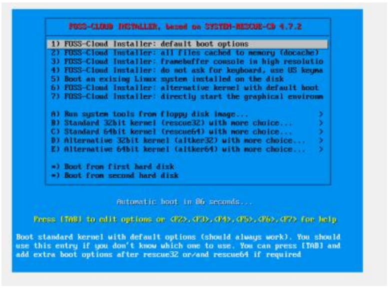
System
Base Memory: 12829 MB
Processors: 8
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, Nested VT-x/AMD-V, PAE/NX, KVM Paravirtualization

Display
Video Memory: 98 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage
Controller: IDE
IDE Secondary Device 0: [Optical Drive] foss-cloud-installer-1.3.2.iso (982.00 MB)
Controller: SATA
SATA Port 0: foss.vdi (Normal, 140.72 GB)

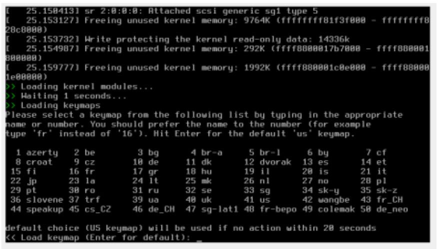
Audio
Host Driver: Default

Step 7: Select 1st option



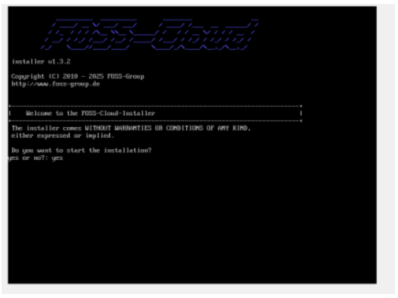
```
FOSS-CLOUD INSTALLER, based on SYSTEM-RESCUE-CD 4.7.2

1) FOSS-Cloud Installer: default boot options
2) FOSS-Cloud Installer: all files cached to memory (docache)
3) FOSS-Cloud Installer: framebuffer console in high resolutio
4) FOSS-Cloud Installer: do not ask for keyboard, use US keyma
5) Boot an exising Linux system installed on the disk
6) FOSS-Cloud Installer: alternative kernel with default boot
7) FOSS-Cloud Installer: directly start the graphical environm

A) Run system tools from floppy disk image...              >
B) Standard 32bit kernel (rescue32) with more choice...    >
C) Standard 64bit kernel (rescue64) with more choice...    >
D) Alternative 32bit kernel (altker32) with more choice... >
E) Alternative 64bit kernel (altker64) with more choice... >

*) Boot from first hard disk
*) Boot from second hard disk

          Automatic boot in 86 seconds...

  Press [TAB] to edit options or <F2>,<F3>,<F4>,<F5>,<F6>,<F7> for help

Boot standard kernel with default options (should always work). You should
use this entry if you don't know which one to use. You can press [TAB] and
add extra boot options after rescue32 or/and rescue64 if required
```

- Hit Enter



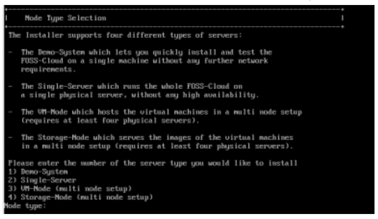
```
[   25.150413] sr 2:0:0:0: Attached scsi generic sg1 type 5
[   25.153127] Freeing unused kernel memory: 9764K (ffffffff81f3f000 - ffffffff8
28c8000)
[   25.153732] Write protecting the kernel read-only data: 14336k
[   25.154987] Freeing unused kernel memory: 292K (ffff8800017b7000 - ffff880001
800000)
[   25.159777] Freeing unused kernel memory: 1992K (ffff880001c0e000 - ffff88000
1e00000)
>> Loading kernel modules...
>> Waiting 1 seconds...
>> Loading keymaps
Please select a keymap from the following list by typing in the appropriate
name or number. You should prefer the name to the number (for example
type 'fr' instead of '16'). Hit Enter for the default 'us' keymap.

  1 azerty   2 be        3 bg       4 br-a     5 br-l     6 by       7 cf
  8 croat    9 cz       10 de      11 dk      12 dvorak  13 es      14 et
 15 fi      16 fr       17 gr      18 hu      19 il      20 is      21 it
 22 jp      23 la       24 lt      25 mk      26 nl      27 no      28 pl
 29 pt      30 ro       31 ru      32 se      33 sg      34 sk-y    35 sk-z
 36 slovene 37 trf      39 ua      40 uk      41 us      42 wangbe  43 fr_CH
 44 speakup 45 cs_CZ    46 de_CH   47 sg-lat1 48 fr-bepo 49 colemak 50 de_neo

default choice (US keymap) will be used if no action within 20 seconds
<< Load keymap (Enter for default): _
```

- Yes



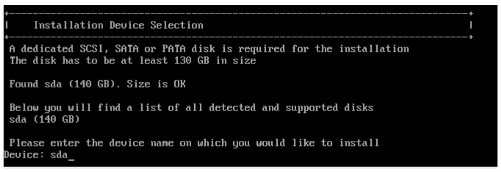
```
installer v1.3.2

Copyright (C) 2010 - 2025 FOSS-Group
http://www.foss-group.de


+------------------------------------------------------------------+
|    Welcome to the FOSS-Cloud-Installer                           |
+------------------------------------------------------------------+
 The installer comes WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
 either expressed or implied.

 Do you want to start the installation?
yes or no?: yes
```

- Select 1) Demo-System



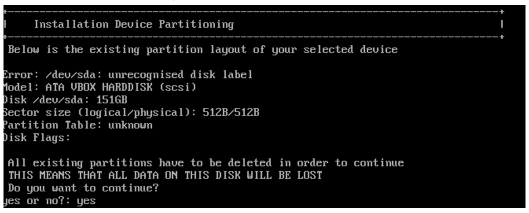
```
+------------------------------------------------------------------+
|    Node Type Selection                                           |
+------------------------------------------------------------------+
 The Installer supports four different types of servers:

 -  The Demo-System which lets you quickly install and test the
    FOSS-Cloud on a single machine without any further network
    requirements.

 -  The Single-Server which runs the whole FOSS-Cloud on
    a single physical server, without any high availability.

 -  The VM-Node which hosts the virtual machines in a multi node setup
    (requires at least four physical servers).

 -  The Storage-Node which serves the images of the virtual machines
    in a multi node setup (requires at least four physical servers).

 Please enter the number of the server type you would like to install
 1) Demo-System
 2) Single-Server
 3) VM-Node (multi node setup)
 4) Storage-Node (multi node setup)
Node type:
```

- Enter sda



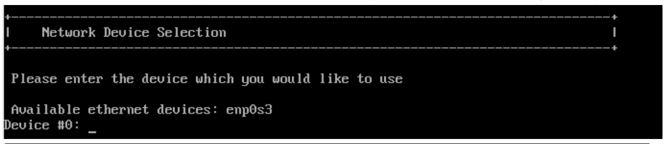
```
+------------------------------------------------------------------+
|    Installation Device Selection                                 |
+------------------------------------------------------------------+
 A dedicated SCSI, SATA or PATA disk is required for the installation
 The disk has to be at least 130 GB in size

 Found sda (140 GB). Size is OK

 Below you will find a list of all detected and supported disks
 sda (140 GB)

 Please enter the device name on which you would like to install
Device: sda_
```

- Enter yes

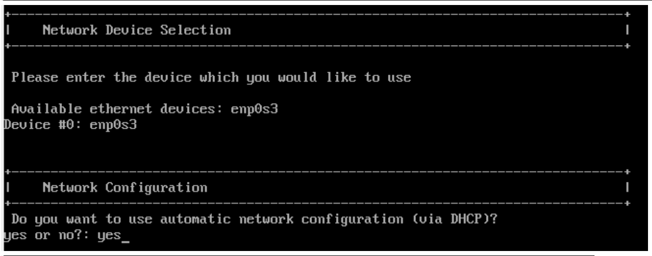


```
+------------------------------------------------------------------+
|    Installation Device Partitioning                              |
+------------------------------------------------------------------+
 Below is the existing partition layout of your selected device

Error: /dev/sda: unrecognised disk label
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sda: 151GB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:

 All existing partitions have to be deleted in order to continue
 THIS MEANS THAT ALL DATA ON THIS DISK WILL BE LOST
 Do you want to continue?
yes or no?: yes
```

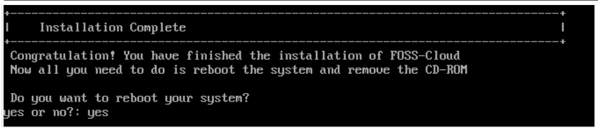
- Enter enp0s3 and then yes and then again yes for reboot



```
+------------------------------------------------------------------+
|    Network Device Selection                                      |
+------------------------------------------------------------------+

 Please enter the device which you would like to use

 Available ethernet devices: enp0s3
Device #0: _
```

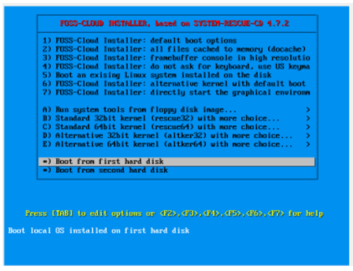


```
+------------------------------------------------------------------+
|    Network Device Selection                                      |
+------------------------------------------------------------------+

 Please enter the device which you would like to use

 Available ethernet devices: enp0s3
Device #0: enp0s3



+------------------------------------------------------------------+
|    Network Configuration                                         |
+------------------------------------------------------------------+
 Do you want to use automatic network configuration (via DHCP)?
yes or no?: yes_
```



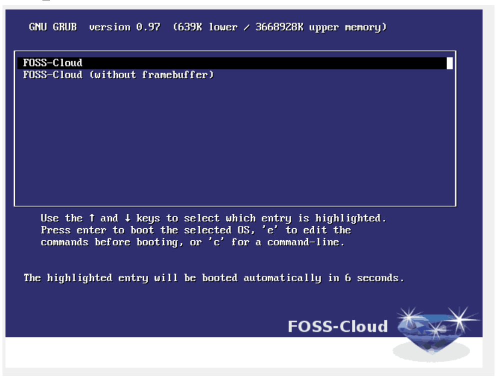
```
+------------------------------------------------------------------+
|    Installation Complete                                         |
+------------------------------------------------------------------+
 Congratulation! You have finished the installation of FOSS-Cloud
 Now all you need to do is reboot the system and remove the CD-ROM

 Do you want to reboot your system?
yes or no?: yes
```

Step 8: Boot from first Hard disk

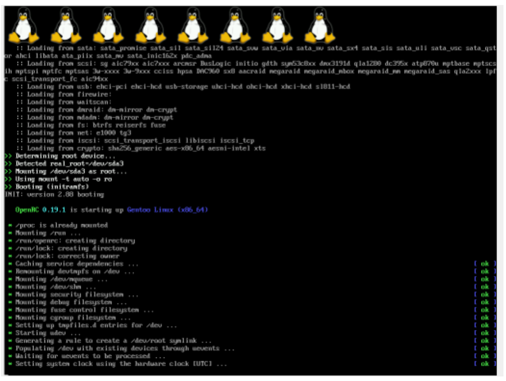


```
FOSS-CLOUD INSTALLER, based on SYSTEM-RESCUE-CD 4.7.2

1) FOSS-Cloud Installer: default boot options
2) FOSS-Cloud Installer: all files cached to memory (docache)
3) FOSS-Cloud Installer: framebuffer console in high resolutio
4) FOSS-Cloud Installer: do not ask for keyboard, use US keyma
5) Boot an exising Linux system installed on the disk
6) FOSS-Cloud Installer: alternative kernel with default boot
7) FOSS-Cloud Installer: directly start the graphical environm

A) Run system tools from floppy disk image...                >
B) Standard 32bit kernel (rescue32) with more choice...      >
C) Standard 64bit kernel (rescue64) with more choice...      >
D) Alternative 32bit kernel (altker32) with more choice...   >
E) Alternative 64bit kernel (altker64) with more choice...   >

*) Boot from first hard disk
*) Boot from second hard disk

   Press [TAB] to edit options or <F2>,<F3>,<F4>,<F5>,<F6>,<F7> for help

Boot local OS installed on first hard disk
```

Step 9: Select FOSS-Cloud

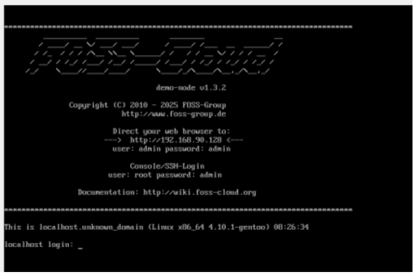


```
  GNU GRUB  version 0.97  (639K lower / 3668928K upper memory)

 FOSS-Cloud
 FOSS-Cloud (without framebuffer)

    Use the ↑ and ↓ keys to select which entry is highlighted.
    Press enter to boot the selected OS, 'e' to edit the
    commands before booting, or 'c' for a command-line.

 The highlighted entry will be booted automatically in 6 seconds.
```



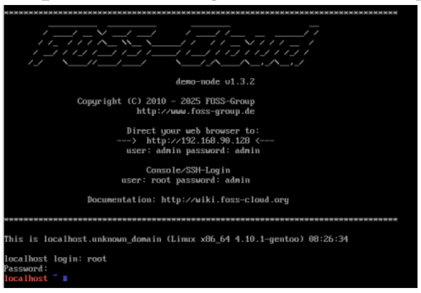
```
   :: Loading from sata: sata_promise sata_sil sata_sil24 sata_svw sata_via sata_nv sata_sx4 sata_sis sata_uli sata_vsc sata_qst
or ahci libata ata_piix sata_mv sata_inic162x pdc_adma
   :: Loading from scsi: sg aic79xx aic7xxx arcmsr BusLogic initio gdth sym53c8xx dmx3191d qla1280 dc395x atp870u mptbase mptscs
ih mptspi mptfc mptsas 3w-xxxx 3w-9xxx cciss hpsa DAC960 sx8 aacraid megaraid megaraid_mbox megaraid_mm megaraid_sas qla2xxx lpf
c scsi_transport_fc aic94xx
   :: Loading from usb: ehci-pci ehci-hcd usb-storage uhci-hcd ohci-hcd xhci-hcd sl811-hcd
   :: Loading from firewire:
   :: Loading from waitscan:
   :: Loading from dmraid: dm-mirror dm-crypt
   :: Loading from mdadm: dm-mirror dm-crypt
   :: Loading from fs: btrfs reiserfs fuse
   :: Loading from net: e1000 tg3
   :: Loading from iscsi: scsi_transport_iscsi libiscsi iscsi_tcp
   :: Loading from crypto: sha256_generic aes-x86_64 aesni-intel xts
>> Determining root device...
>> Detected real_root=/dev/sda3
>> Mounting /dev/sda3 as root...
>> Using mount -t auto -o ro
>> Booting (initramfs)
INIT: version 2.88 booting

   OpenRC 0.19.1 is starting up Gentoo Linux (x86_64)

 * /proc is already mounted
 * Mounting /run ...
 * /run/openrc: creating directory
 * /run/lock: creating directory
 * /run/lock: correcting owner
 * Caching service dependencies ...                                    [ ok ]
 * Remounting devtmpfs on /dev ...                                     [ ok ]
 * Mounting /dev/mqueue ...                                            [ ok ]
 * Mounting /dev/shm ...                                               [ ok ]
 * Mounting security filesystem ...                                    [ ok ]
 * Mounting debug filesystem ...                                       [ ok ]
 * Mounting fuse control filesystem ...                                [ ok ]
 * Mounting cgroup filesystem ...                                      [ ok ]
 * Setting up tmpfiles.d entries for /dev ...                          [ ok ]
 * Starting udev ...                                                   [ ok ]
 * Generating a rule to create a /dev/root symlink ...                 [ ok ]
 * Populating /dev with existing devices through uevents ...           [ ok ]
 * Waiting for uevents to be processed ...                             [ ok ]
 * Setting system clock using the hardware clock [UTC] ...             [ ok ]
```

After this IP Address should come



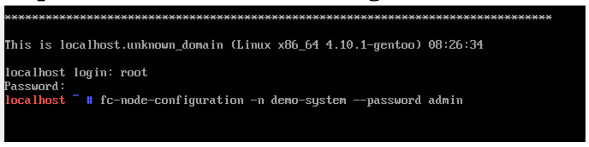
```
******************************************************************************

                          demo-node v1.3.2

               Copyright (C) 2010 - 2025 FOSS-Group
                          http://www.foss-group.de

                      Direct your web browser to:
                    ---> http://192.168.90.128 <---
                      user: admin password: admin

                          Console/SSH-Login
                      user: root password: admin

               Documentation: http://wiki.foss-cloud.org

******************************************************************************

This is localhost.unknown_domain (Linux x86_64 4.10.1-gentoo) 08:26:34

localhost login: _
```

Step 10: Now Login with root and password admin

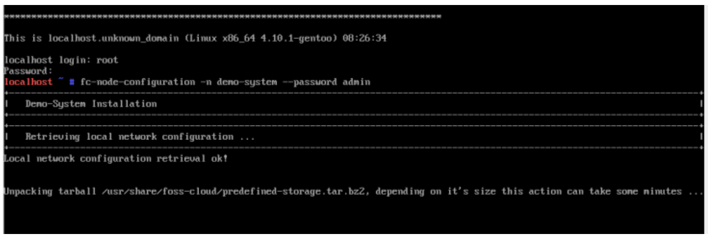


```
******************************************************************************

                          demo-node v1.3.2

               Copyright (C) 2010 - 2025 FOSS-Group
                          http://www.foss-group.de

                      Direct your web browser to:
                    ---> http://192.168.90.128 <---
                      user: admin password: admin

                          Console/SSH-Login
                      user: root password: admin

               Documentation: http://wiki.foss-cloud.org

******************************************************************************

This is localhost.unknown_domain (Linux x86_64 4.10.1-gentoo) 08:26:34

localhost login: root
Password:
localhost ~ #
```

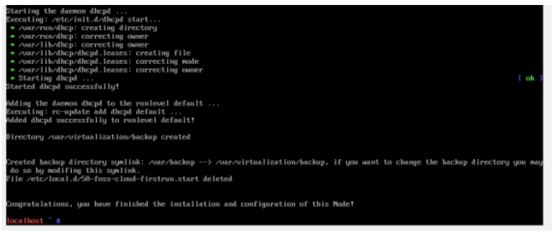
Step 11: Execute fc-node-configuration -n demo-system -password admin



```
******************************************************************************

This is localhost.unknown_domain (Linux x86_64 4.10.1-gentoo) 08:26:34

localhost login: root
Password:
localhost ~ # fc-node-configuration -n demo-system --password admin
```

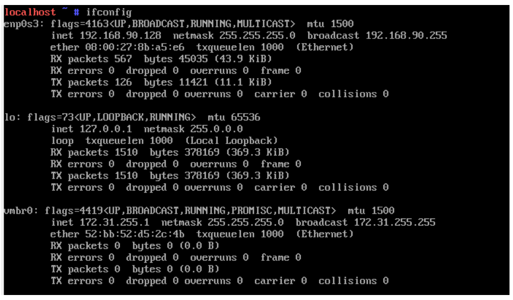


```
******************************************************************************

This is localhost.unknown_domain (Linux x86_64 4.10.1-gentoo) 08:26:34

localhost login: root
Password:
localhost ~ # fc-node-configuration -n demo-system --password admin
+------------------------------------------------------------------------------------------------+
|   Demo-System Installation                                                                     |
+------------------------------------------------------------------------------------------------+
+------------------------------------------------------------------------------------------------+
|   Retrieving local network configuration ...                                                   |
+------------------------------------------------------------------------------------------------+
Local network configuration retrieval ok!


Unpacking tarball /usr/share/foss-cloud/predefined-storage.tar.bz2, depending on it's size this action can take some minutes ...
```



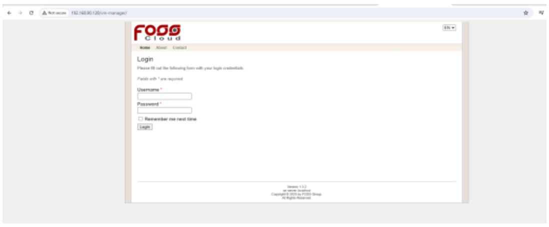
```
Starting the daemon dhcpd ...
Executing: /etc/init.d/dhcpd start...
 * /var/run/dhcp: creating directory
 * /var/run/dhcp: correcting owner
 * /var/lib/dhcp: correcting owner
 * /var/lib/dhcp/dhcpd.leases: creating file
 * /var/lib/dhcp/dhcpd.leases: correcting mode
 * /var/lib/dhcp/dhcpd.leases: correcting owner
 * Starting dhcpd ...                                                                          [ ok ]
Started dhcpd successfully!

Adding the daemon dhcpd to the runlevel default ...
Executing: rc-update add dhcpd default ...
Added dhcpd successfully to runlevel default!

Directory /var/virtualization/backup created


Created backup directory symlink: /var/backup --> /var/virtualization/backup. if you want to change the backup directory you may
 do so by modifing this symlink.
File /etc/local.d/50-foss-cloud-firstrun.start deleted


Congratulations, you have finished the installation and configuration of this Node!

localhost ~ #
```

- Ifconfig and then note this ip address



```
localhost ~ # ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.90.128  netmask 255.255.255.0  broadcast 192.168.90.255
        ether 08:00:27:8b:a5:e6  txqueuelen 1000  (Ethernet)
        RX packets 567  bytes 45035 (43.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 126  bytes 11421 (11.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1510  bytes 378169 (369.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1510  bytes 378169 (369.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vmbr0: flags=4419<UP,BROADCAST,RUNNING,PROMISC,MULTICAST>  mtu 1500
        inet 172.31.255.1  netmask 255.255.255.0  broadcast 172.31.255.255
        ether 52:bb:52:d5:2c:4b  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Step 12: Open Browser in your main machine and enter IP address of Foss Virtual Machine



Not secure 192.168.90.128/vm-manager/

Login

Please fill out the following form with your login credentials:

Fields with * are required.

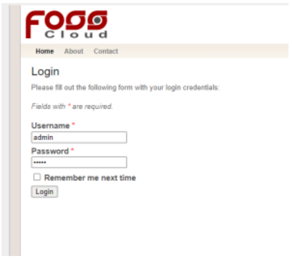
Username *

Password *

Remember me next time

Login

Step 13: Login with username "admin" & password "admin"



Login

Please fill out the following form with your login credentials:

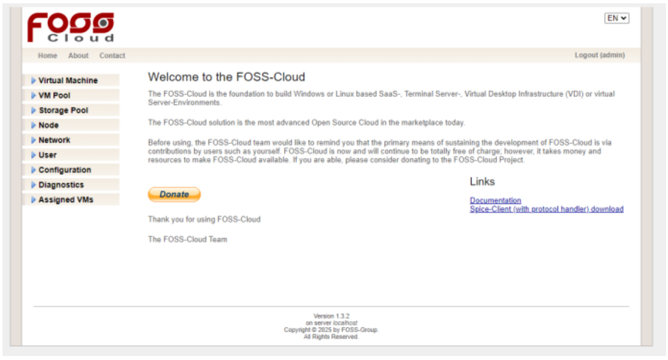
Fields with * are required.

Username *
admin

Password *
•••••

Remember me next time

Login



Welcome to the FOSS-Cloud

The FOSS-Cloud is the foundation to build Windows or Linux based SaaS-, Terminal Server-, Virtual Desktop Infrastructure (VDI) or virtual Server-Environments.

The FOSS-Cloud solution is the most advanced Open Source Cloud in the marketplace today.

Before using, the FOSS-Cloud team would like to remind you that the primary means of sustaining the development of FOSS-Cloud is via contributions by users such as yourself. FOSS-Cloud is now and will continue to be totally free of charge; however, it takes money and resources to make FOSS-Cloud available. If you are able, please consider donating to the FOSS-Cloud Project.

Links

Documentation
Spice-Client (with protocol handler) download

Thank you for using FOSS-Cloud

The FOSS-Cloud Team

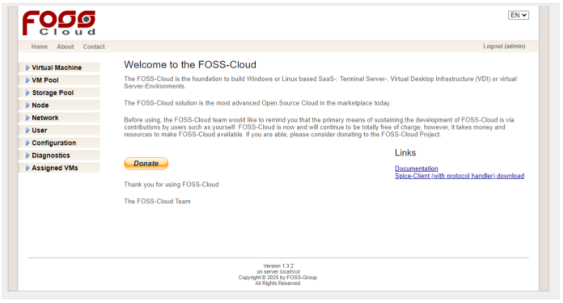
Version 1.3.2
on server localhost
Copyright © 2025 by FOSS-Group.
All Rights Reserved.

Practical No 8

Aim: Implement FOSS-Cloud Functionality - VSI Platform as a Service (PaaS)

Solution:

Step 1: Login Foss Cloud



EN

Home About Contact Logout (admin)

- Virtual Machine
- VM Pool
- Storage Pool
- Node
- Network
- User
- Configuration
- Diagnostics
- Assigned VMs

Welcome to the FOSS-Cloud

The FOSS-Cloud is the foundation to build Windows or Linux based SaaS-, Terminal Server-, Virtual Desktop Infrastructure (VDI) or virtual Server-Environments.

The FOSS-Cloud solution is the most advanced Open Source Cloud in the marketplace today.

Before using, the FOSS-Cloud team would like to remind you that the primary means of sustaining the development of FOSS-Cloud is via contributions by users such as yourself. FOSS-Cloud is now and will continue to be totally free of charge; however, it takes money and resources to make FOSS-Cloud available. If you are able, please consider donating to the FOSS-Cloud Project.

Donate

Thank you for using FOSS-Cloud

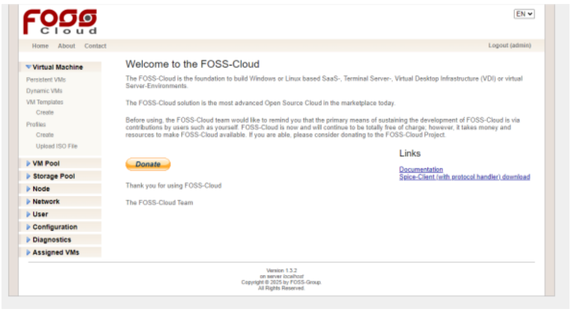
The FOSS-Cloud Team

Links

Documentation
Spice-Client (with protocol handler) download

Version 1.3.2
on server *localhost*
Copyright © 2025 by FOSS-Group.
All Rights Reserved.

Step 2: Expand Virtual Machine Tab



EN

Home About Contact Logout (admin)

- Virtual Machine
  - Persistent VMs
  - Dynamic VMs
  - VM Templates
    - Create
  - Profiles
    - Create
    - Upload ISO File
- VM Pool
- Storage Pool
- Node
- Network
- User
- Configuration
- Diagnostics
- Assigned VMs

Welcome to the FOSS-Cloud

The FOSS-Cloud is the foundation to build Windows or Linux based SaaS-, Terminal Server-, Virtual Desktop Infrastructure (VDI) or virtual Server-Environments.

The FOSS-Cloud solution is the most advanced Open Source Cloud in the marketplace today.

Before using, the FOSS-Cloud team would like to remind you that the primary means of sustaining the development of FOSS-Cloud is via contributions by users such as yourself. FOSS-Cloud is now and will continue to be totally free of charge; however, it takes money and resources to make FOSS-Cloud available. If you are able, please consider donating to the FOSS-Cloud Project.

Donate

Thank you for using FOSS-Cloud

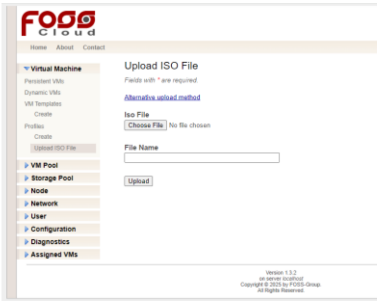
The FOSS-Cloud Team

Links

Documentation
Spice-Client (with protocol handler) download

Version 1.3.2
on server *localhost*
Copyright © 2025 by FOSS-Group.
All Rights Reserved.

Step 3:Upload ISO File of Ubuntu 16



Home About Contact

Upload ISO File

*Fields with * are required.*

Alternative upload method

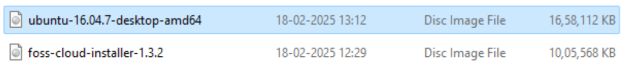
Iso File
Choose File No file chosen

File Name

Upload

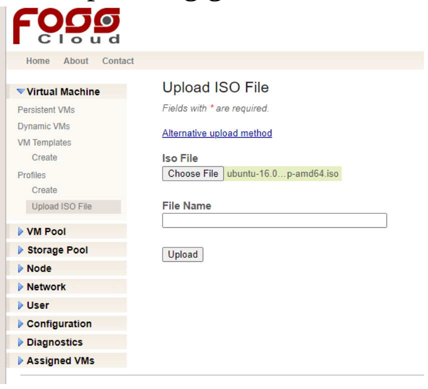
Version 1.3.2
on server *localhost*
Copyright © 2025 by FOSS-Group.
All Rights Reserved.

- Select & upload



ubuntu-16.04.7-desktop-amd64	18-02-2025 13:12	Disc Image File	16,58,112 KB
foss-cloud-installer-1.3.2	18-02-2025 12:29	Disc Image File	10,05,568 KB

After uploading give File name



Home About Contact

- Virtual Machine
  - Persistent VMs
  - Dynamic VMs
  - VM Templates
    - Create
  - Profiles
    - Create
    - Upload ISO File
- VM Pool
- Storage Pool
- Node
- Network
- User
- Configuration
- Diagnostics
- Assigned VMs

Upload ISO File

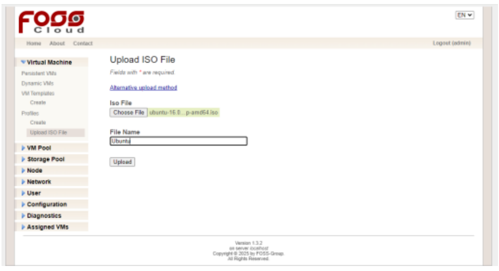
*Fields with * are required.*

Alternative upload method

Iso File
Choose File ubuntu-16.0...p-amd64.iso

File Name

Upload


Upload ISO File

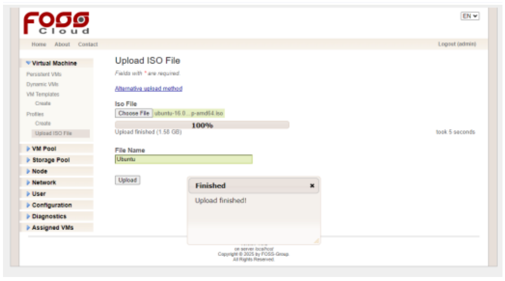
Fields with * are required.

Alternative upload method

Iso File: Choose File ubuntu-16.0...p-amd64.iso

File Name: Ubuntu

Version 1.3.2
on server localhost
Copyright © 2025 by FOSS-Group.
All Rights Reserved.


Upload ISO File

Fields with * are required.

Alternative upload method

Iso File: Choose File ubuntu-16.0...p-amd64.iso

100%

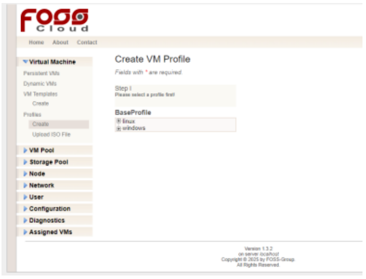
Upload finished (1.58 GB) took 5 seconds

File Name: Ubuntu

Finished

Upload finished!

Step 4: Create VM Profile

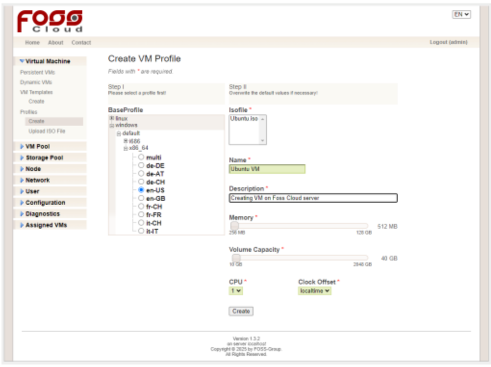

Create VM Profile

Fields with * are required.

Step I
Please select a profile first!

BaseProfile
linux
windows

- Now select BaseProfile [Your OS Windows or Linux] → x86_64 →en-US then give Name ,Description, memory, Vol capacity, CPU & localtime then create


Create VM Profile

Fields with * are required.

Step I
Please select a profile first!

Step II
Overwrite the default values if necessary!

BaseProfile: linux / windows / default / i686 / x86_64: multi, de-DE, de-AT, de-CH, (●) en-US, en-GB, fr-CH, fr-FR, it-CH, it-IT

Isofile *: Ubuntu iso

Name *: Ubuntu VM

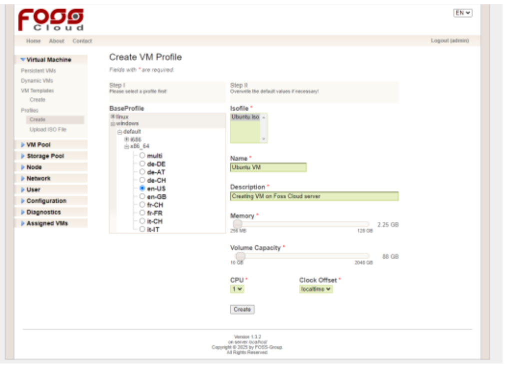
Description *: Creating VM on Foss Cloud server

Memory *: 256 MB – 128 GB, 512 MB

Volume Capacity *: 10 GB – 2048 GB, 40 GB

CPU *: 1

Clock Offset *: localtime


Create VM Profile

BaseProfile: windows / default / x86_64: (●) en-US

Isofile *: Ubuntu iso

Name *: Ubuntu VM

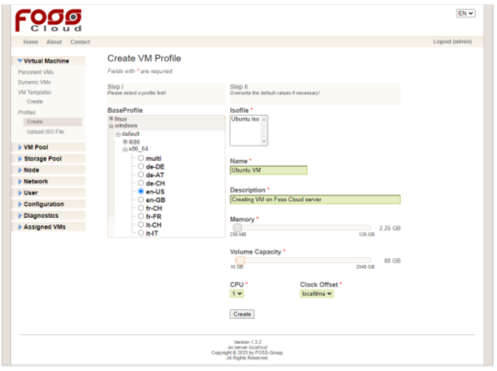
Description *: Creating VM on Foss Cloud server

Memory *: 256 MB – 128 GB, 2.25 GB

Volume Capacity *: 10 GB – 2048 GB, 88 GB

CPU *: 1

Clock Offset *: localtime


Create VM Profile

BaseProfile: windows / default / x86_64: (●) en-US

Isofile *: Ubuntu iso

Name *: Ubuntu VM

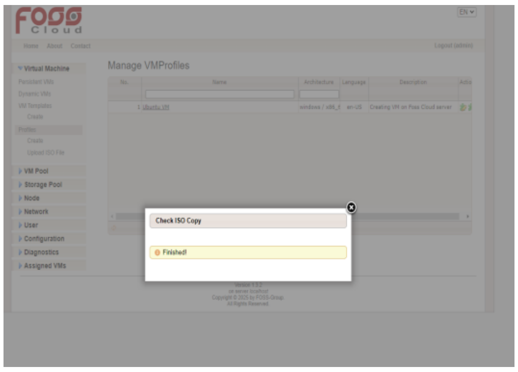
Description *: Creating VM on Foss Cloud server

Memory *: 256 MB – 128 GB, 2.25 GB

Volume Capacity *: 10 GB – 2048 GB, 88 GB

CPU *: 1

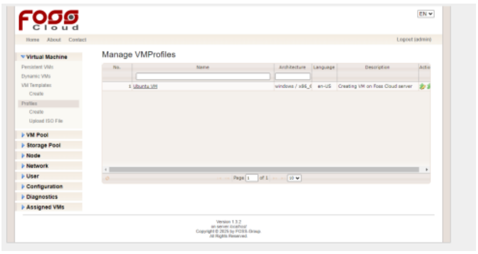
Clock Offset *: localtime


Manage VMProfiles

No.	Name	Architecture	Language	Description
1	Ubuntu VM	windows / x86_64	en-US	Creating VM on Foss Cloud server

Check ISO Copy

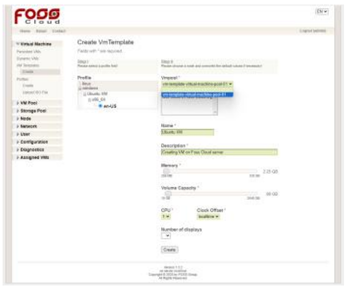
Finished!


Manage VMProfiles

No.	Name	Architecture	Language	Description
1	Ubuntu VM	windows / x86_64	en-US	Creating VM on Foss Cloud server

Page 1 of 1

Step 5: Now Navigate to VM Template then click on create after that select Profile [os Windows or Linux] → [VM Name] → x86_64→en-US and then select Vmpool & number of display 1 and create


Create VmTemplate

Profile: windows › Ubuntu VM › x86_64 › en-US

Vmpool *: vm-template-virtual-machine-pool-01

Name *: Ubuntu VM

Description *: Creating VM on Foss Cloud server

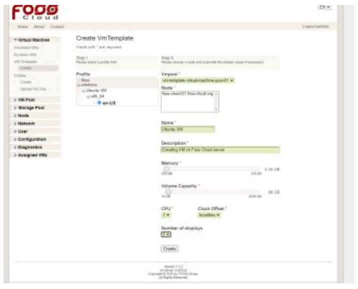
Memory *: 2.25 GB

CPU *: 1

Clock Offset *: localtime

Number of displays

Create


Create VmTemplate

Profile: windows › Ubuntu VM › x86_64 › en-US

Vmpool *: vm-template-virtual-machine-pool-01

Name *: Ubuntu VM

Description *: Creating VM on Foss Cloud server

Memory *: 2.25 GB

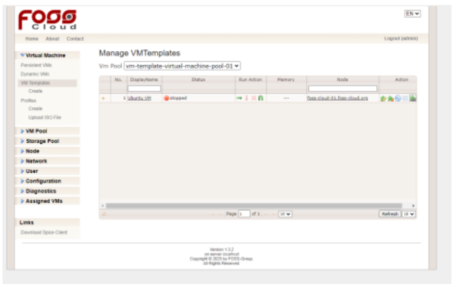
CPU *: 1

Clock Offset *: localtime

Number of displays: 1

Create

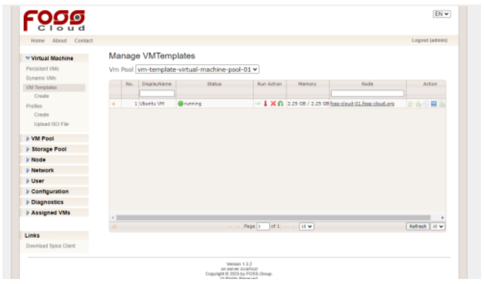
Step 6: Click on Green Arrow → to start VM


Manage VMTemplates

Vm Pool: vm-template-virtual-machine-pool-01

No.	DisplayName	Status	Run Action	Memory	Node	Action
1	Ubuntu VM	stopped		---	foss-cloud-01.foss-cloud.org	

Version 1.3.2
on server localhost
Copyright © 2025 by FOSS-Group.
All Rights Reserved.


Manage VMTemplates

Vm Pool: vm-template-virtual-machine-pool-01

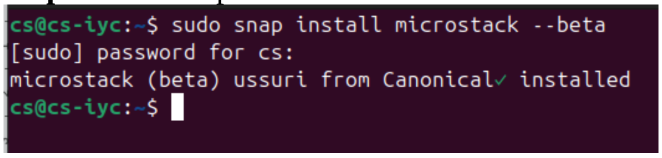
No.	DisplayName	Status	Run Action	Memory	Node	Action
1	Ubuntu VM	running		2.25 GB / 2.25 GB	foss-cloud-01.foss-cloud.org	

Version 1.3.2
on server localhost
Copyright © 2025 by FOSS-Group.

Practical No 10

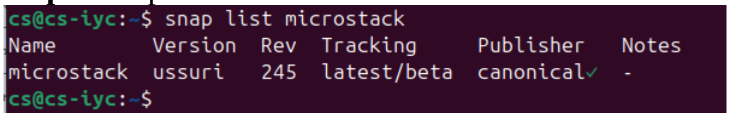
Aim:
Implementation of Openstack with user and private network creation.

Solution:
Step 1: sudo snap install microstack --beta



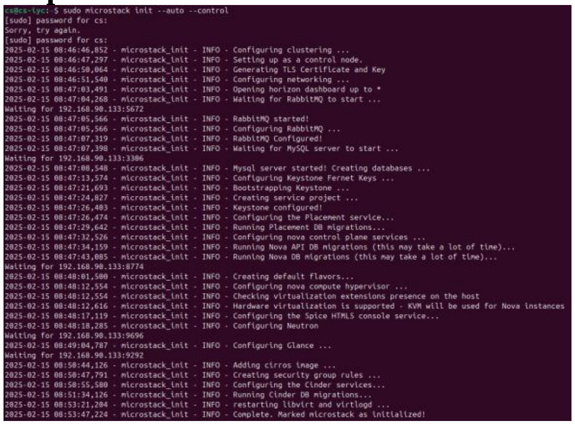
```
cs@cs-iyc:~$ sudo snap install microstack --beta
[sudo] password for cs:
microstack (beta) ussuri from Canonical✓ installed
cs@cs-iyc:~$
```

Step 2: snap list microstack



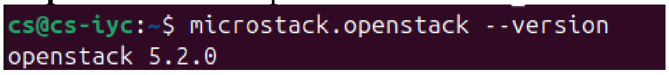
```
cs@cs-iyc:~$ snap list microstack
Name        Version  Rev  Tracking     Publisher   Notes
microstack  ussuri   245  latest/beta  canonical✓  -
cs@cs-iyc:~$
```

Step 3: sudo microstack init --auto --control



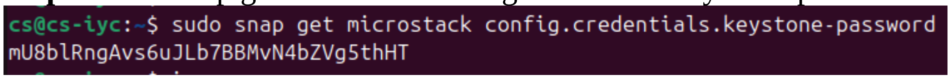
```
cs@cs-iyc:~$ sudo microstack init --auto --control
[sudo] password for cs:
Sorry, try again.
[sudo] password for cs:
2025-02-15 08:46:46,852 - microstack_init - INFO - Configuring clustering ...
2025-02-15 08:46:47,297 - microstack_init - INFO - Setting up as a control node.
2025-02-15 08:46:50,064 - microstack_init - INFO - Generating TLS Certificate and Key
2025-02-15 08:46:51,540 - microstack_init - INFO - Configuring networking ...
2025-02-15 08:47:03,491 - microstack_init - INFO - Opening horizon dashboard up to *
2025-02-15 08:47:04,268 - microstack_init - INFO - Waiting for RabbitMQ to start ...
Waiting for 192.168.90.133:5672
2025-02-15 08:47:05,566 - microstack_init - INFO - RabbitMQ started!
2025-02-15 08:47:05,566 - microstack_init - INFO - Configuring RabbitMQ ...
2025-02-15 08:47:07,319 - microstack_init - INFO - RabbitMQ Configured!
2025-02-15 08:47:07,398 - microstack_init - INFO - Waiting for MySQL server to start ...
Waiting for 192.168.90.133:3306
2025-02-15 08:47:08,548 - microstack_init - INFO - Mysql server started! Creating databases ...
2025-02-15 08:47:13,574 - microstack_init - INFO - Configuring Keystone Fernet Keys ...
2025-02-15 08:47:21,693 - microstack_init - INFO - Bootstrapping Keystone ...
2025-02-15 08:47:24,827 - microstack_init - INFO - Creating service project ...
2025-02-15 08:47:26,403 - microstack_init - INFO - Keystone configured!
2025-02-15 08:47:26,474 - microstack_init - INFO - Configuring the Placement service...
2025-02-15 08:47:29,642 - microstack_init - INFO - Running Placement DB migrations...
2025-02-15 08:47:32,526 - microstack_init - INFO - Configuring nova control plane services ...
2025-02-15 08:47:34,159 - microstack_init - INFO - Running Nova API DB migrations (this may take a lot of time)...
2025-02-15 08:47:43,085 - microstack_init - INFO - Running Nova DB migrations (this may take a lot of time)...
Waiting for 192.168.90.133:8774
2025-02-15 08:48:01,500 - microstack_init - INFO - Creating default flavors...
2025-02-15 08:48:12,554 - microstack_init - INFO - Configuring nova compute hypervisor ...
2025-02-15 08:48:12,554 - microstack_init - INFO - Checking virtualization extensions presence on the host
2025-02-15 08:48:12,616 - microstack_init - INFO - Hardware virtualization is supported - KVM will be used for Nova instances
2025-02-15 08:48:17,119 - microstack_init - INFO - Configuring the Spice HTML5 console service...
2025-02-15 08:48:18,285 - microstack_init - INFO - Configuring Neutron
Waiting for 192.168.90.133:9696
2025-02-15 08:49:04,787 - microstack_init - INFO - Configuring Glance ...
Waiting for 192.168.90.133:9292
2025-02-15 08:50:44,126 - microstack_init - INFO - Adding cirros image ...
2025-02-15 08:50:47,791 - microstack_init - INFO - Creating security group rules ...
2025-02-15 08:50:55,580 - microstack_init - INFO - Configuring the Cinder services...
2025-02-15 08:51:34,126 - microstack_init - INFO - Running Cinder DB migrations...
2025-02-15 08:53:21,204 - microstack_init - INFO - restarting libvirt and virtlogd ...
2025-02-15 08:53:47,224 - microstack_init - INFO - Complete. Marked microstack as initialized!
```

Step 4: microstack.openstack --version



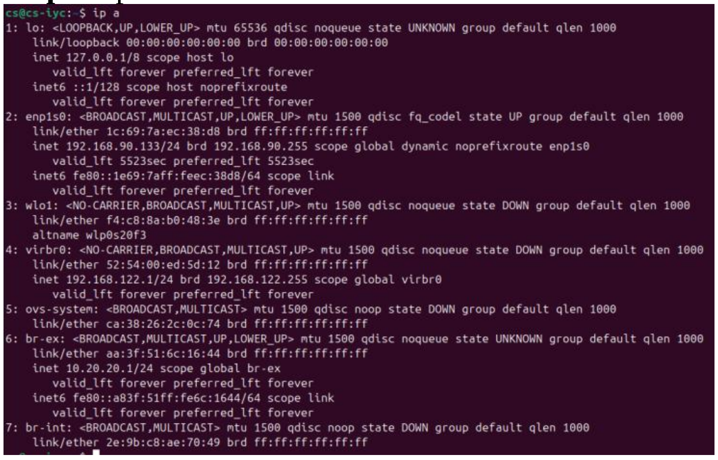
```
cs@cs-iyc:~$ microstack.openstack --version
openstack 5.2.0
```

Step 5: sudo snap get microstack config.credentials.keystone-password



```
cs@cs-iyc:~$ sudo snap get microstack config.credentials.keystone-password
mU8blRngAvs6uJLb7BBMvN4bZVg5thHT
```

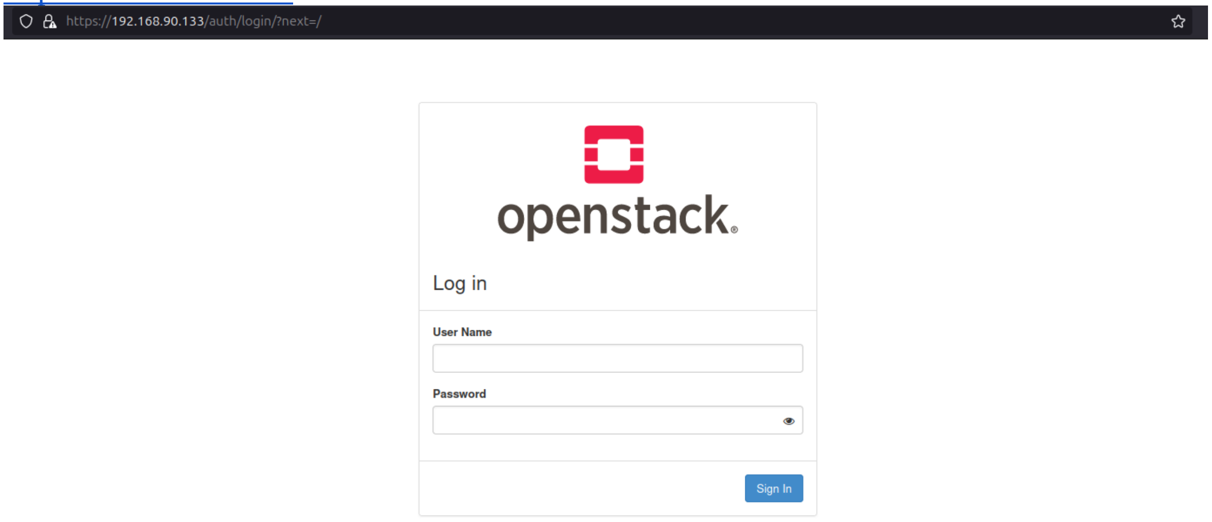
Step 6: ip a



```
cs@cs-iyc:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 1c:69:7a:ec:38:d8 brd ff:ff:ff:ff:ff:ff
    inet 192.168.90.133/24 brd 192.168.90.255 scope global dynamic noprefixroute enp1s0
       valid_lft 5523sec preferred_lft 5523sec
    inet6 fe80::1e69:7aff:feec:38d8/64 scope link
       valid_lft forever preferred_lft forever
3: wlo1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether f4:c8:8a:b0:48:3e brd ff:ff:ff:ff:ff:ff
    altname wlp0s20f3
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:ed:5d:12 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
       valid_lft forever preferred_lft forever
5: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether ca:38:26:2c:0c:74 brd ff:ff:ff:ff:ff:ff
6: br-ex: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
    link/ether aa:3f:51:6c:16:44 brd ff:ff:ff:ff:ff:ff
    inet 10.20.20.1/24 scope global br-ex
       valid_lft forever preferred_lft forever
    inet6 fe80::a83f:51ff:fe6c:1644/64 scope link
       valid_lft forever preferred_lft forever
7: br-int: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 2e:9b:c8:ae:70:49 brd ff:ff:ff:ff:ff:ff
```

Step 7: now open Browser and enter https://yourip

https://192.168.90.133


https://192.168.90.133/auth/login/?next=/

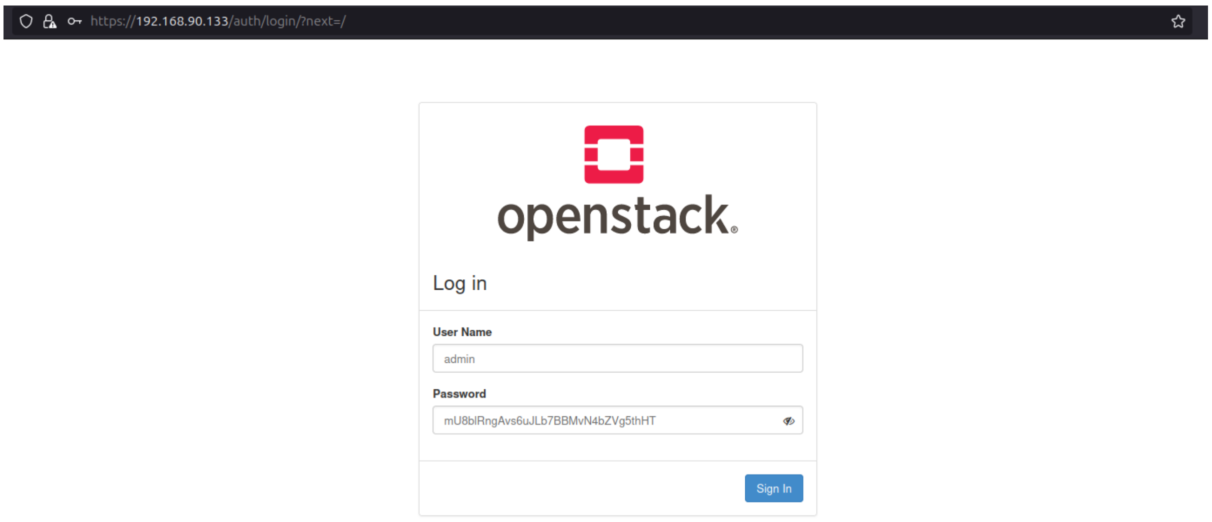
openstack.

Log in

User Name

Password

Step 8: Login user name: admin & password from them terminal which got through previous command


https://192.168.90.133/auth/login/?next=/

openstack.

Log in

User Name

admin

Password

mU8blRngAvs6uJLb7BBMvN4bZVg5thHT