

THIAGARAJAR COLLEGE OF ENGINEERING

(Govt. Aided & Autonomous Institution, Affiliated to Anna University, Chennai)



DEPARTMENT OF INFORMATION TECHNOLOGY

18IT670-CLOUD COMPUTING LAB

[RECORD]

THIAGARAJAR COLLEGE OF ENGINEERING

(Govt. Aided & Autonomous Institution, Affiliated to Anna University, Chennai)



B.TECH INFORMATION TECHNOLOGY

18IT670 – CLOUD COMPUTING LAB

[RECORD]

REG. NO : 18IT116 ROLL NO : 61854

NAME : YAJITH VISHWA S

BRANCH : INFORMATION TECHNOLOGY

THIAGARAJAR COLLEGE OF ENGINEERING

(Govt. Aided & Autonomous Institution, Affiliated to Anna University, Chennai)



This is certified to be the Bonafide Record of 18IT670-CLOUD COMPUTING Lab Practical work done by Mr. / Ms. YAJITH VISHWA S for the Sixth Semester - B.Tech Information Technology Degree Course, Department of Information Technology, Thiagarajar college of Engineering, Madurai - 625015, during the academic year 2021-2022.

Staff-In Charge

[S.Thiruchadai Pandeeswari]

Register No: 18IT116

Roll No: 61854

Submitted for the Continuous Assessment Test - Practical Examination held on _____

EXTERNAL EXAMINER

INTERNAL EXAMINER

TABLE OF CONTENTS

S NO	TITLE OF EXPERIMENT	PAGE NO
1	Study on Virtualization Platforms	5
2	Implementation of SOAP based and RESTFUL web services	9
3	Implementation of RMI in VMs	20
4	Implementation of PaaS using IBM Cloud Foundry	27
5	Implementation of DBaaS using MongoDB atlas	34
6	Implementation of IDaaS using Auth0	57
7	Implementation of Storage as a Service using DropBox	63
8	Study on Live migration using Hyper V and Xen	66
9	Study on AR-VR as a service from Cloud	110
10	Simulation of cloud scenario using cloudsim	130
11	Study on Private Cloud Implementation using OpenStack	135
12	Implementation of MapReduce Program on Hadoop	143
13	Study on LibVirt	160

Ex.No : 1	Study on Virtualization Platforms
27.02.2021	

Aim

To do a study on cloud services provided by major service providers.

Description:

Write about different cloud services – SaaS, PaaS, IaaS, DaaS, IDaaS, Analytics as a service, Storage as a service, Monitoring as a service, Communication as a service.

Present your study on the service products of major cloud service providers like AWS, Microsoft, HP, Alibaba cloud, Salesforce, IBM etc., you may present details like name of the service offering, parameters of the service like number of connections/computations allowed, Pricing etc.

SaaS:

Software as a service (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.

PaaS:

The cloud has dramatically changed how business applications are built and run. Delivering a new application is now as fast as opening your Internet browser.

Platform as a service (PaaS) is a proven model for running applications without the hassle of maintaining on-premises hardware and software infrastructure at your company.

IaaS:

Infrastructure as a service (IaaS) is an instant computing infrastructure, provisioned and managed over the internet. It is one of the four types of cloud services, along with software as a service (SaaS), platform as a service (PaaS) and serverless.

DaaS:

Desktop as a Service (DaaS) is a cloud computing offering where a service provider delivers virtual desktops to end users over the Internet, licensed with a per-user subscription.

The provider takes care of backend management for small businesses that find creating their own virtual desktop infrastructure to be too expensive or resource-consuming. This management typically includes maintenance, back-up, updates, and data storage.

Analytics as a Service:

Analytics-as-a-Service (AaaS) provides subscription-based data analytics software and procedures through the cloud. AaaS typically offers a fully customizable BI solution with end-to-end capabilities, organizing, analyzing, and presenting data in a way that lets even non-IT professionals gain insight and take action.

Storage as a Service:

Storage as a service (STaaS) is a managed service in which the provider supplies the customer with access to a data storage platform. The service can be delivered on premises from infrastructure that is dedicated to a single customer, or it can be delivered from the public cloud as a shared service that's purchased by subscription and is billed according to one or more usage metrics.

Monitoring as a Service:

Cloud Monitoring as a Service is referred to a type of on demand IT service that provides cloud monitoring and management tools for monitoring cloud based platforms, websites, servers, IT Infrastructure etc.

Communication as a Service:

Communication as a Service (CaaS) is a trending service provided to the customers by CaaS vendors that is implemented over cloud computing technology.

Services	AWS	Google	Microsoft	IBM	HP	Salesforce
SaaS	AWS Elastic Beanstalk	Google Workspace	Playfab	Mendix	Helion Developer	Salesforce Platform
PaaS	AWS infrastructure	App Engine, Cloud Storage, Virtual Private Cloud, Persistent Disk	Azure Resource Manager templates	IBM Infrastructure Suite	HPE greenlake for Virtual Machines	
IaaS	Amazon Web Services	Compute Engine	Microsoft Windows Virtual Desktop	IBM Db2	HP Device as a Service	Force.com DaaS
DaaS	AWS Data Exchange	Virtual desktops	Azure Active Directory	IBM Security Verify		Salesforce Identity

IDaaS	AWS Identity & Access Management	Cloud Identity	Azure Communication Services	IBM PushaS Notifications	HP UCa	
Communication as a service	Amazon Simple Notification Service(SNS)	Google Hangouts	Microsoft Azure Portal	IBM Z Monitoring Suite	Salesforce Service Analytics	
Monitoring as a service	Amazon cloudwatch	Google cloud operation suite	Azure Databricks	IBM Analytics Engine	Tableau	
Analytics as a service	Amazon Athena	Dataflow ,BigQuery	Playfab	Mendix	Helion Developer	Salesforce Platform

Result

Thus the study on cloud service providers is done for various cloud service providers.

Ex.No : 2
27.02.2021

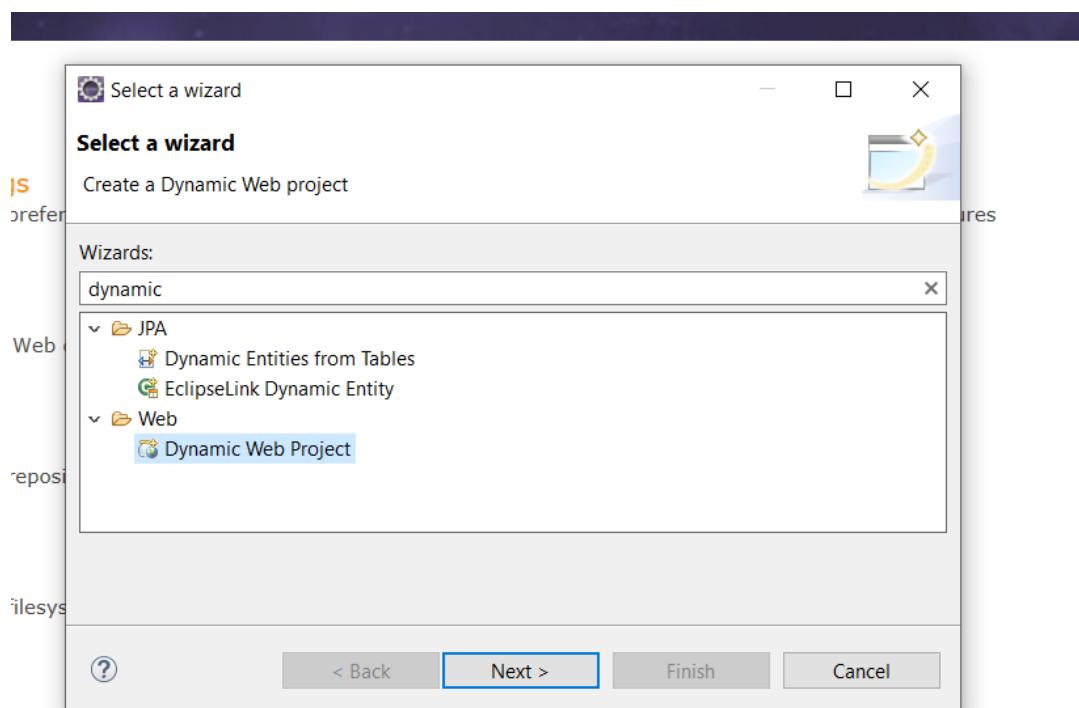
Implementation of SOAP based and RESTFUL web services

Aim

To implement SOAP based webservice in Java.

Procedure

1. Create a new dynamic web project and select Tomcat v9 as the runtime.



and install your Marketplace favorites

2. Create an Interface and a Java class

Interface

```
package com.yajith.soap;
```

```
public interface Student {  
    public String getName();  
    public void setName(String name);  
}
```

Implementation

```
package com.yajith.soap;
```

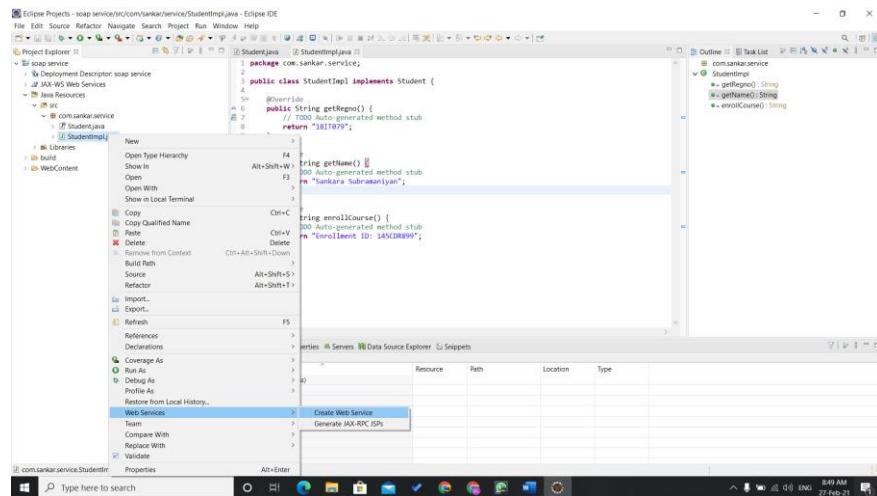
```
public class StudentImpl implements Student {  
    private String name=null;  
  
    @Override  
    public String getName() {  
        // TODO Auto-generated method stub  
        return this.name;  
    }  
  
    @Override  
    public void setName(String name) {  
        // TODO Auto-generated method stub  
        this.name = name;  
    }  
}
```

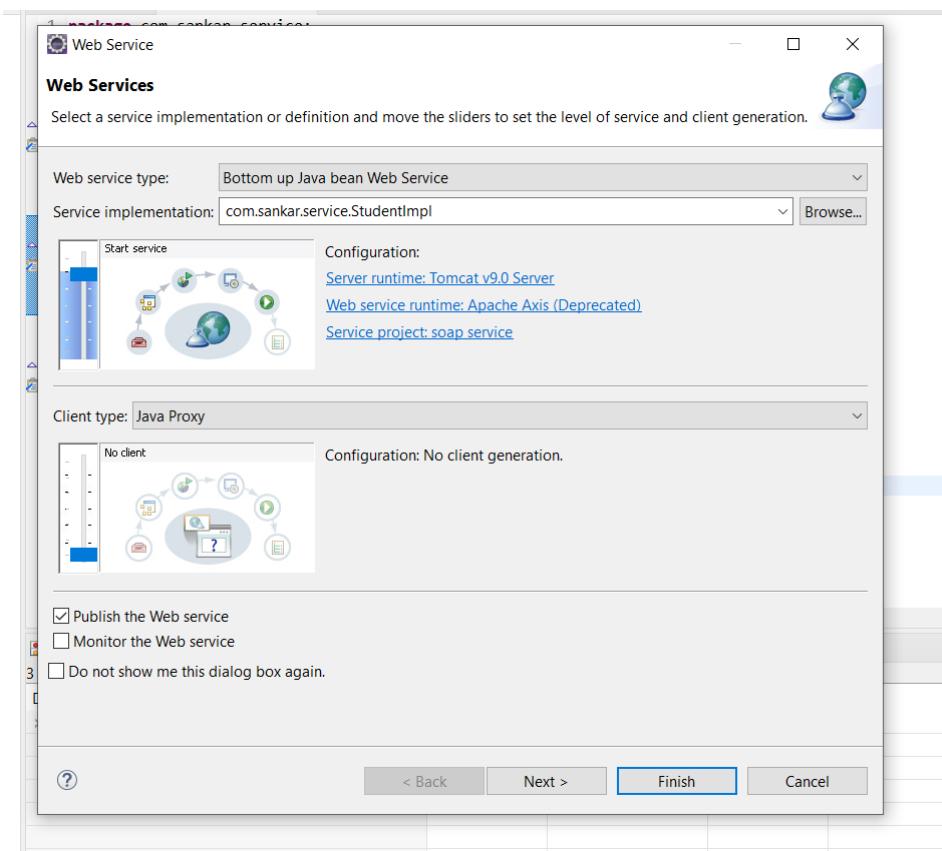
}

- > Deployment Descriptor: soap service
- > JAX-WS Web Services
- ✓ Java Resources
 - ✓ src
 - ✓ com.sankar.service
 - > Student.java
 - > StudentImpl.java
 - ✓ Libraries
- > build
- > WebContent

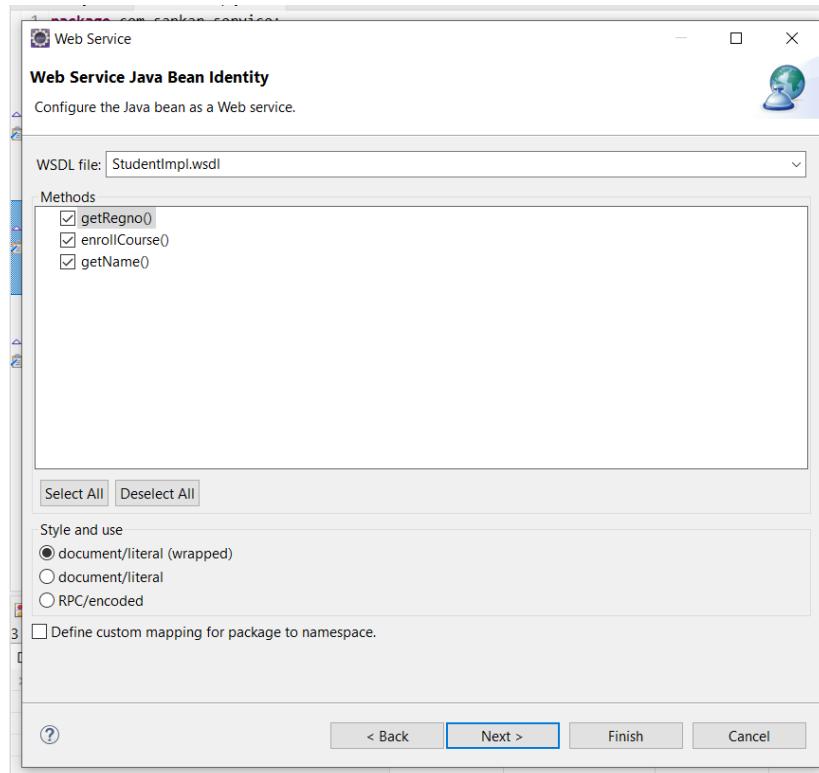
```
2
3 public class
4
5     @Override
6     public
7         // re1
8     }
9
10    @Override
11    public
12        // re1
13    }
14
15    }
16
```

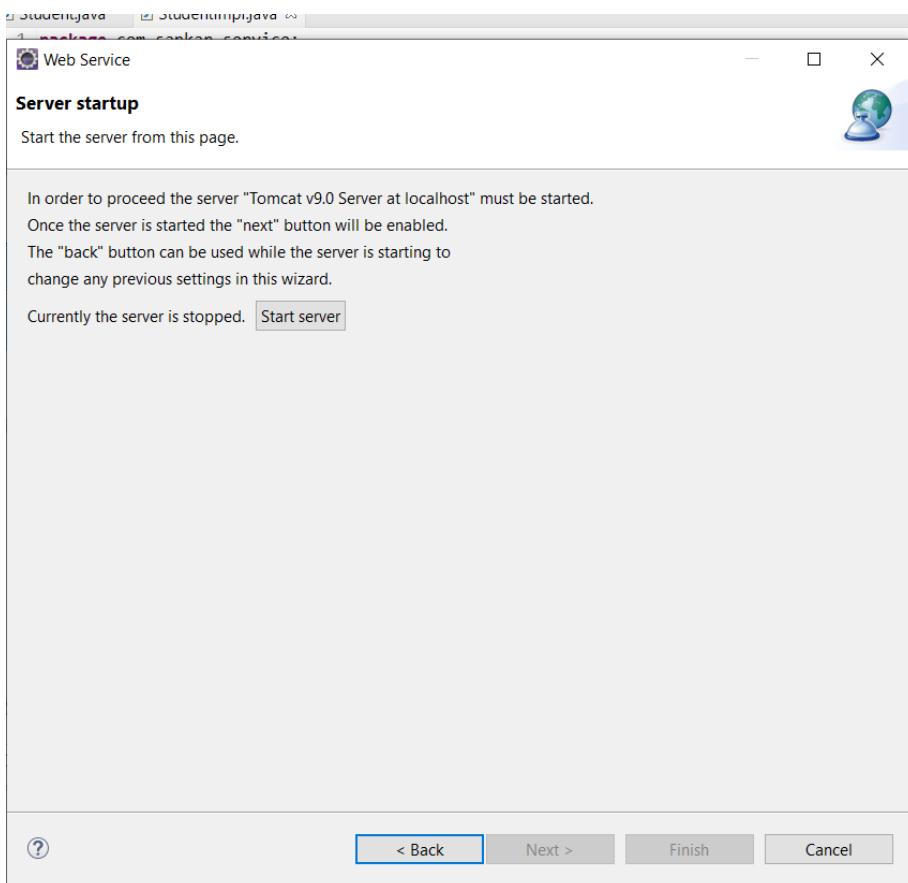
3. Create webservice



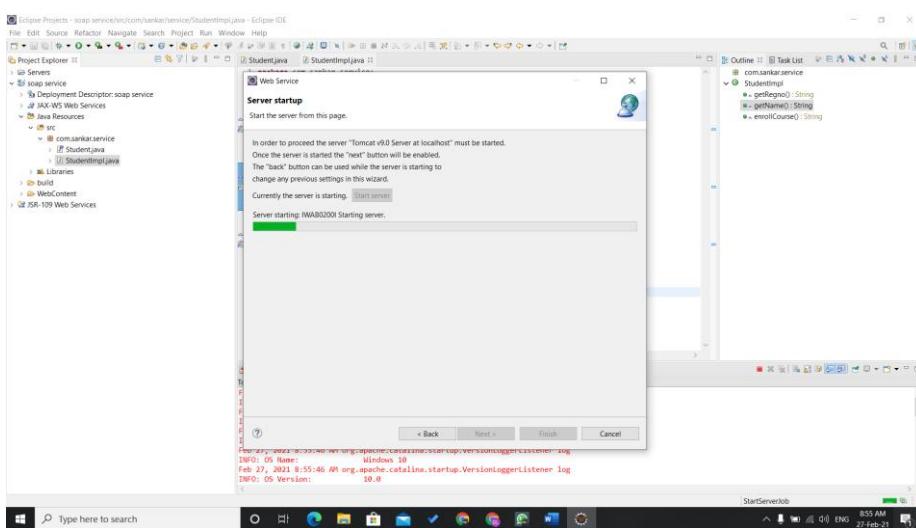


4. Select all the methods





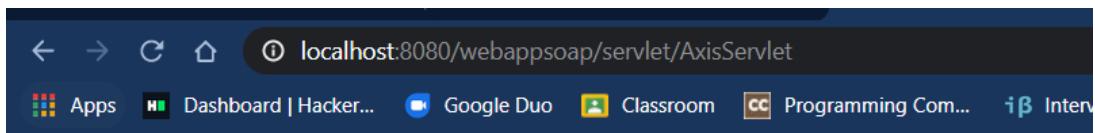
5. Start the server and view localhost at port 80.



```

Student.java StudentImpl.java web.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/
3   <display-name>soap service</display-name>
4   <welcome-file-list>
5     <welcome-file>index.html</welcome-file>
6     <welcome-file>index.htm</welcome-file>
7     <welcome-file>index.jsp</welcome-file>
8     <welcome-file>default.html</welcome-file>
9     <welcome-file>default.htm</welcome-file>
10    <welcome-file>default.jsp</welcome-file>
11  </welcome-file-list>
12  <servlet>
13    <display-name>Apache Axis Servlet</display-name>
14    <servlet-name>AxisServlet</servlet-name>
15    <servlet-class>org.apache.axis.transport.http.AxisServlet</servlet-class>
16  </servlet>
17  <servlet-mapping>
18    <servlet-name>AxisServlet</servlet-name>
19    <url-pattern>/servlet/AxisServlet</url-pattern>
20  </servlet-mapping>
21  <servlet-mapping>
22    <servlet-name>AxisServlet</servlet-name>
23    <url-pattern>*.jws</url-pattern>
24  </servlet-mapping>
25  <servlet-mapping>
26    <servlet-name>AxisServlet</servlet-name>
27    <url-pattern>/services/*</url-pattern>
28  </servlet-mapping>

```



And now... Some Services

- StudentImpl ([wsdl](#))
 - getName
 - setName
- AdminService ([wsdl](#))
 - AdminService
- Version ([wsdl](#))
 - getVersion

localhost:8080/webappsoap/services/StudentImpl?wsdl

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

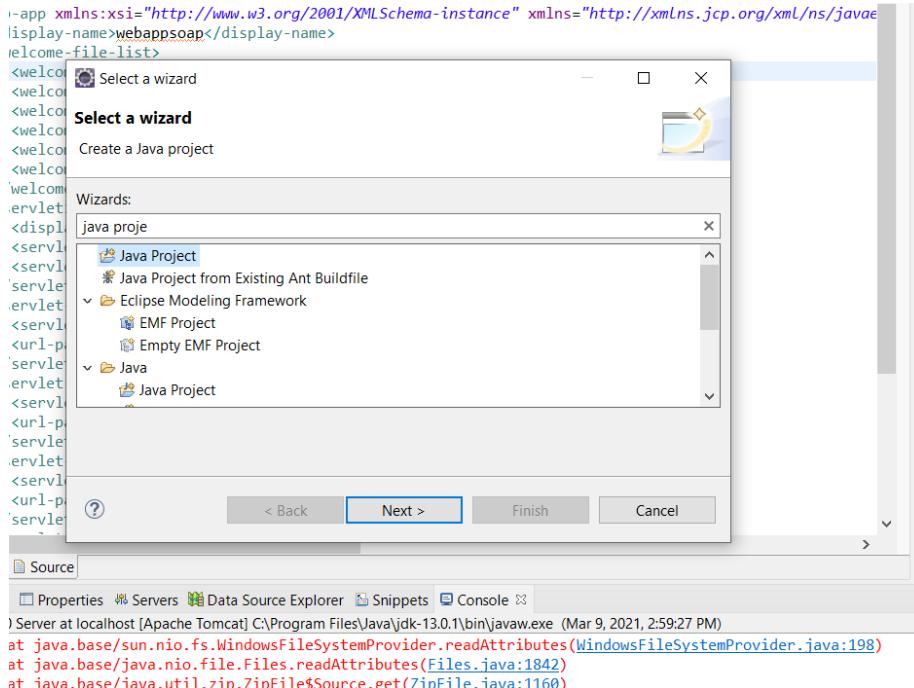
```

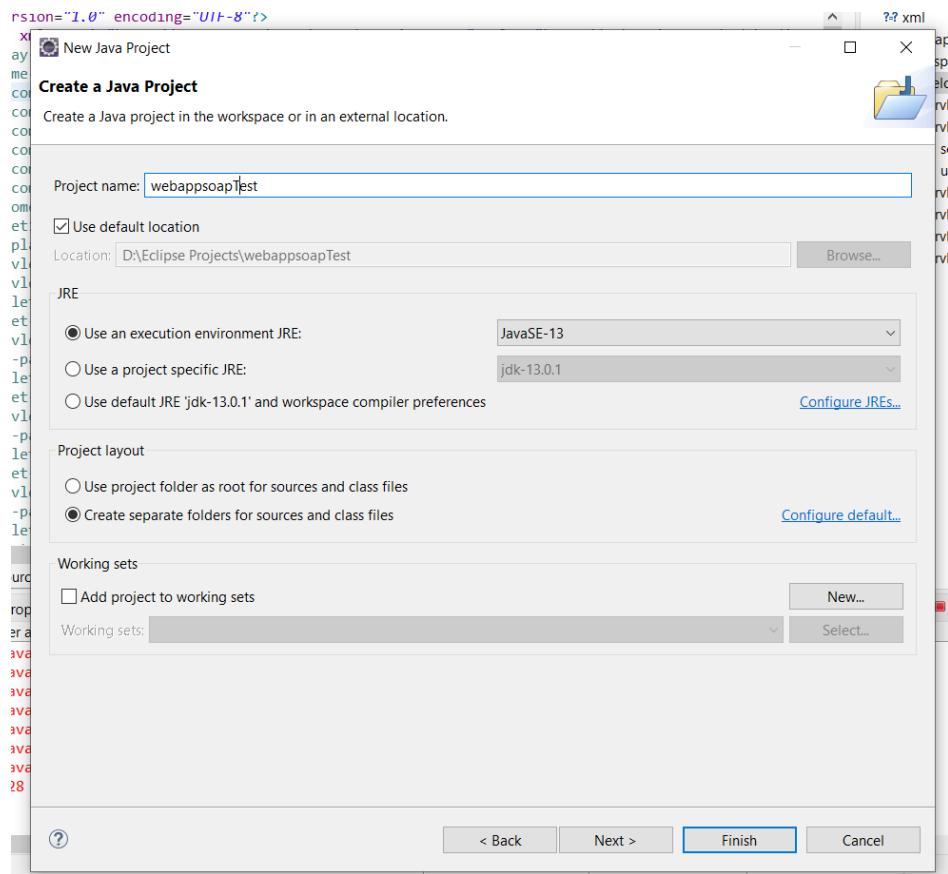
<?wsdl:definitions xmlns:apachesoap="http://xml.apache.org/xml-soap" xmlns:impl="http://soap.sankar.com" xmlns:is="http://www.w3.org/2001/XMLSchema-instance" targetNamespace="http://soap.sankar.com" wsdl:version="1.1" wsdl:encodingStyle="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="http://soap.sankar.com" xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/">
    <!-- WSDL created by Apache Axis version: 1.4
        Built on Apr 22, 2006 (06:55:48 PDT) -->
    <?wsdl:types>
        <?schema xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" targetNamespace="http://soap.sankar.com">
            <element name="getName">
                <complexType>
                    </complexType>
            </element>
            <element name="getNameResponse">
                <complexType>
                    <sequence>
                        <element name="getNameReturn" type="xsd:string"/>
                    </sequence>
                </complexType>
            </element>
            <element name="setName">
                <complexType>
                    <sequence>
                        <element name="name" type="xsd:string"/>
                    </sequence>
                </complexType>
            </element>
            <element name="setNameResponse">
                <complexType>
                    <sequence>
                    </sequence>
                </complexType>
            </element>
        </schema>
    </?wsdl:types>
    <?wsdl:message name="setNameResponse">
        <wsdl:part element="impl:setNameResponse" name="parameters"> </wsdl:part>
    </?wsdl:message>
    <?wsdl:message name="getNameResponse">
        <wsdl:part element="impl:getNameResponse" name="parameters"> </wsdl:part>
    </?wsdl:message>
    <?wsdl:message name="setNameRequest">
        <wsdl:part element="impl:setName" name="parameters"> </wsdl:part>
    </?wsdl:message>

```

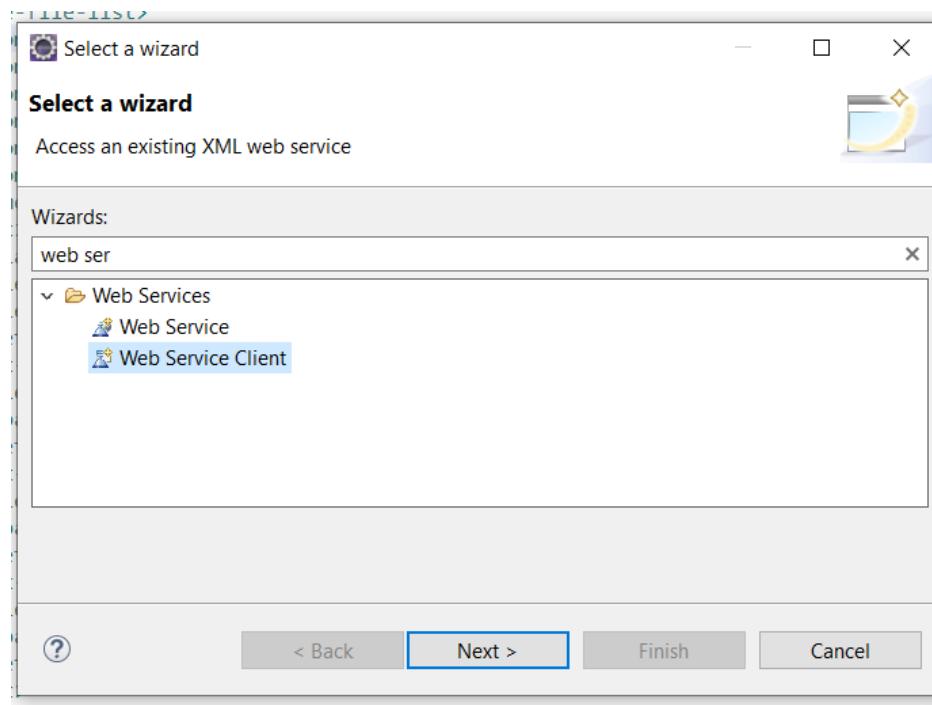
Client

6. Create a java project.

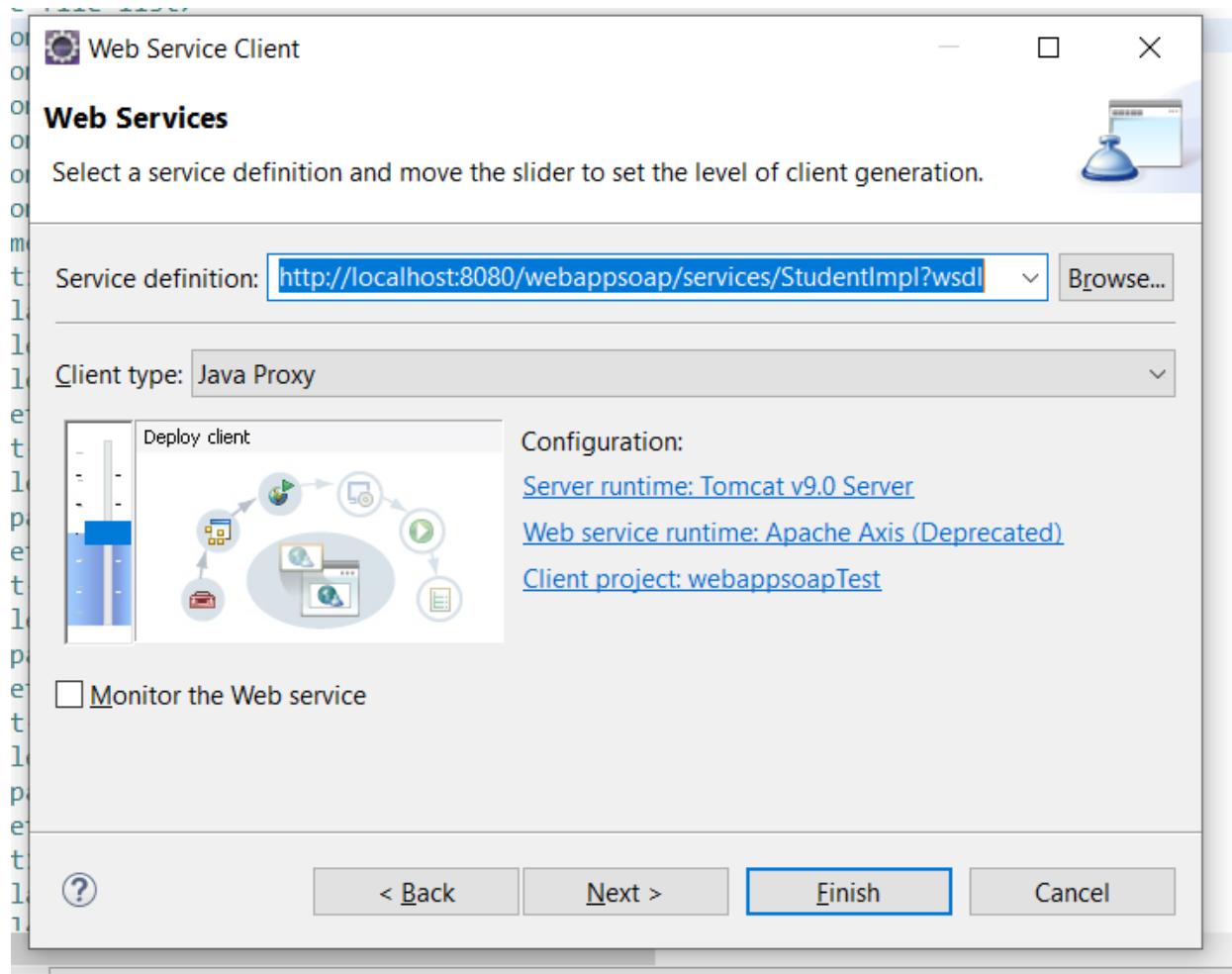




7. Right click src and create new Web Service Client



8. Paste the wsdl url here



9. Paste the url in command prompt with the wsimport -keep command.

```
wsimport -keep http://xml.apache.org/xml-soap/impl?wsdl
//schemas.xmlsoap.org/wsdl/
Microsoft Windows [Version 10.0.19042.746]
(c) 2020 Microsoft Corporation. All rights reserved.
C:\Users\SANKARASUBRAMANIYAN>cd Desktop
C:\Users\SANKARASUBRAMANIYAN\Desktop>mkdir service
C:\Users\SANKARASUBRAMANIYAN\Desktop>cd service
C:\Users\SANKARASUBRAMANIYAN\Desktop\service>wsimport -keep http://localhost:8080/webappsoap/services/StudentImpl?wsdl
```

Name	Date modified	Type	Size
CropClass.class	20-02-2021 20:51	CLASS File	2 KB
CropClassService.class	20-02-2021 20:51	CLASS File	3 KB
GetAge.class	20-02-2021 20:51	CLASS File	1 KB
GetAgeResponse.class	20-02-2021 20:51	CLASS File	1 KB
GetMoisture.class	20-02-2021 20:51	CLASS File	1 KB
GetMoistureResponse.class	20-02-2021 20:51	CLASS File	1 KB
GetName.class	20-02-2021 20:51	CLASS File	1 KB
GetNameResponse.class	20-02-2021 20:51	CLASS File	1 KB
ObjectFactory.class	20-02-2021 20:51	CLASS File	2 KB
package-info.class	20-02-2021 20:51	CLASS File	1 KB
SetAge.class	20-02-2021 20:51	CLASS File	1 KB
SetAgeResponse.class	20-02-2021 20:51	CLASS File	1 KB
SetMoisture.class	20-02-2021 20:51	CLASS File	1 KB
SetMoistureResponse.class	20-02-2021 20:51	CLASS File	1 KB
SetName.class	20-02-2021 20:51	CLASS File	1 KB
SetNameResponse.class	20-02-2021 20:51	CLASS File	1 KB
CropClass.java	20-02-2021 20:51	IntelliJ IDEA Com...	4 KB
CropClassService.java	20-02-2021 20:51	IntelliJ IDEA Com...	3 KB
GetAge.java	20-02-2021 20:51	IntelliJ IDEA Com...	1 KB
GetAgeResponse.java	20-02-2021 20:51	IntelliJ IDEA Com...	2 KB
GetMoisture.java	20-02-2021 20:51	IntelliJ IDEA Com...	1 KB
GetMoistureResponse.java	20-02-2021 20:51	IntelliJ IDEA Com...	2 KB

Client Program

```
package com.yajith.test;

import com.yajith.soap.StudentImpl;

import com.yajith.soap.StudentImplService;

public class TestService {

    public static void main(String[] args){

        StudentImplService wsdl = new StudentImplService();

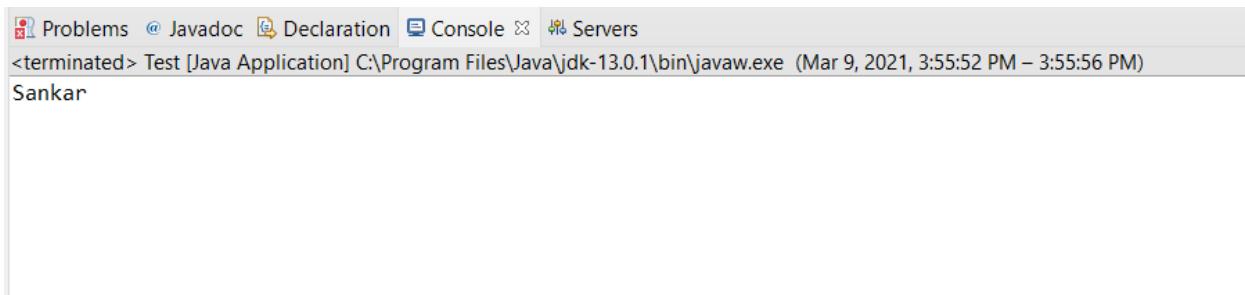
        StudentImpl service = wsdl.getStudentImpl();

        service.setName("Yajith");

        System.out.println(service.getName());

    }

}
```



The screenshot shows the Eclipse IDE interface with the Java Console view open. The title bar includes tabs for Problems, Javadoc, Declaration, Console, and Servers. The Console tab is selected. The main area displays the output of a terminated Java application named 'Test'. The output text is: '<terminated> Test [Java Application] C:\Program Files\Java\jdk-13.0.1\bin\javaw.exe (Mar 9, 2021, 3:55:52 PM – 3:55:56 PM)' followed by the name 'Sankar'.

Result

Thus, the soap webservice is created and verified.

Ex.No : 3	Implementation of RMI in VMs
6/02/21	

Aim:

To run RMI in two different virtual machines.

Code:

RemoteInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface RemoteInterface extends Remote {
    void square(int n) throws RemoteException;
}
```

RemoteImp.java

```
import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;
public class RemoteImp extends UnicastRemoteObject implements
RemoteInterface {
    RemoteImp() throws RemoteException {
```

```
super();  
}  
  
@Override  
  
public void square(int n) throws RemoteException {  
    System.out.println("Square of number: " + n*n);  
}  
}
```

RemoteClient.java

```
import java.rmi.Naming;  
  
import java.rmi.registry.LocateRegistry;  
  
import java.rmi.registry.Registry;  
  
  
public class RemoteClient {  
  
    private RemoteClient() {}  
  
    public static void main(String[] args) {  
        try {  
            RemoteInterface  
            stub=(RemoteInterface)Naming.lookup("rmi:192.168.43.220:5000");  
            stub.square(4);  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
        } catch (Exception e) {  
  
            System.err.println("Client exception: " + e.toString());  
  
            e.printStackTrace();  
  
        }  
  
    }  
  
}
```

RemoteServer.java

```
// package cloudlab;  
  
  
  
import java.rmi.registry.Registry;  
  
import java.rmi.Naming;  
  
import java.rmi.registry.LocateRegistry;  
  
import java.rmi.server.UnicastRemoteObject;;  
  
  
  
  
public class RemoteServer {  
  
    public RemoteServer() {}  
  
    public static void main(String args[]) {  
  
        try {  
  
            RemoteInterface obj = new RemoteImp();  
  
            Naming.rebind("rmi: 192.168.43.220:5000", obj);  
  
        } catch (Exception e) {  
  
            System.err.println("Server exception: " + e.toString());  
  
            e.printStackTrace();  
  
        }  
  
    }  
  
}
```

```

        System.err.println("RemoteServer ready");

    } catch (Exception e) {

        System.err.println("RemoteServer exception: " + e.toString());

        e.printStackTrace();

    }

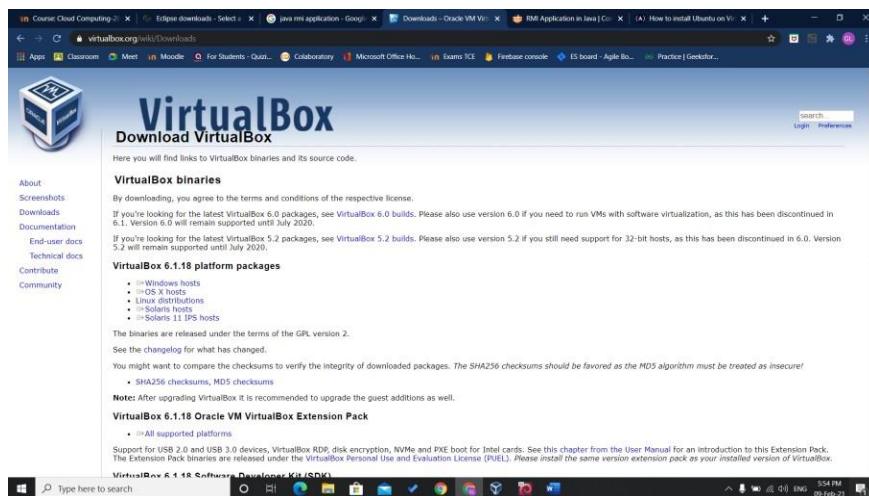
}

}

```

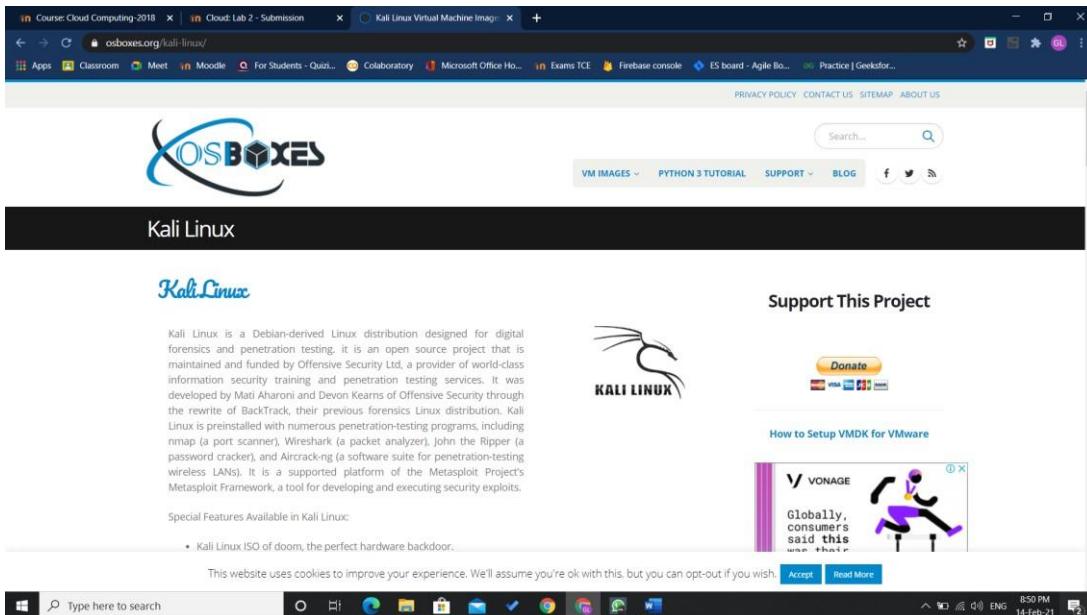
Procedure:

1. Install Virtual Box.



2. Download linux os from the website.

<https://www.osboxes.org/kali-linux/>



3. Load the VDI image.
4. Install OpenJDK in kali.

Sudo apt default-jre

```
root@kali:~# java -version
openjdk version "11.0.3" 2019-04-16
OpenJDK Runtime Environment (build 11.0.3+1-Debian-1)
OpenJDK 64-Bit Server VM (build 11.0.3+1-Debian-1, mixed mode, sharing)
root@kali:~# javac -version
javac 11.0.3
root@kali:~#
```

5. Copy the java and Class files to the vm.

6. Run the following commands

- a. rmic AdderRemote
- b. rmiregistry port
- c. rmi java server

7. Run the client program.

8. Server displays the square of the number once client connected and calls the math function

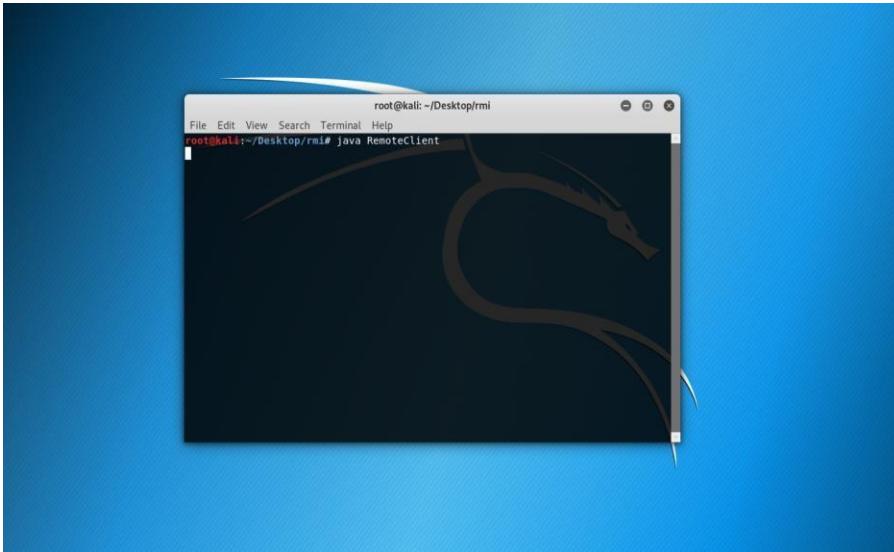
```
root@kali:~/Desktop/rmi
File Edit View Search Terminal Help
root@kali:~/Desktop/rmi# java RemoteServer
RemoteServer ready
Square of number: 16
root@kali:~/Desktop/rmi# java RemoteClient
root@kali:~/Desktop/rmi# java RemoteClient
root@kali:~/Desktop/rmi# 

File Edit View Search Terminal Help
root@kali:~/Desktop/rmi# javac *.java
root@kali:~/Desktop/rmi# rmic RemoteInterface
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
error: RemoteInterface is an interface; stubs are needed only for remote object
classes.
1 error
root@kali:~/Desktop/rmi# start rmiregistry
bash: start: command not found
root@kali:~/Desktop/rmi# rmiregistry 5000
```

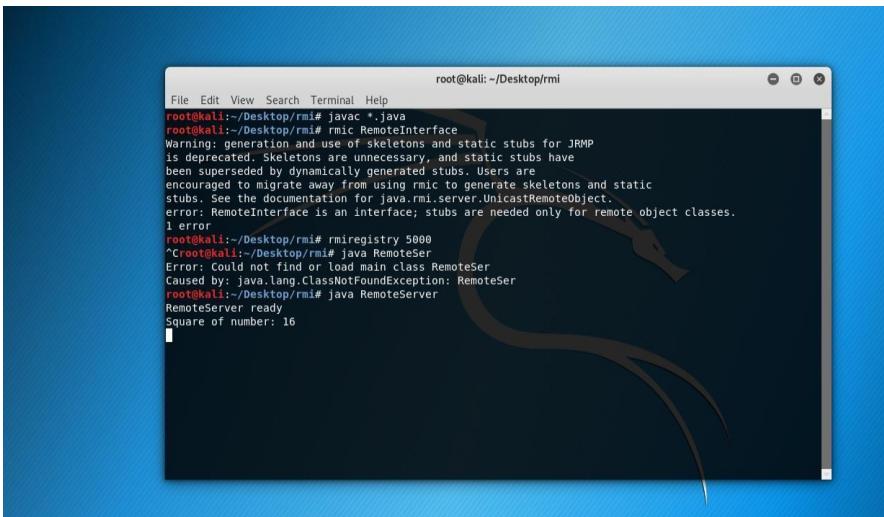
RMI is executed in the same machine.

Launch another VM and run the client program to see the result on server

VM 1 Client



VM 2 Server



Result:

The RMI program is executed and the output is obtained.

Ex.No : 4

12.04.2021

Implementation of PaaS using IBM Cloud Foundry

Aim

To create and deploy application in IBM Cloud Foundry.

Procedure

1. Login to IBM account

The screenshot shows the IBM Cloud dashboard. On the left, there's a sidebar with various icons for services like Meet, Classroom, Moodle, and Watson. The main area has a "Dashboard" header with a dropdown. Below it is a "Quick start" section with six cards:

- Build**: Explore IBM Cloud with this selection of easy starter tutorials and services. (3 min)
- Create a custom dashboard**: Create a shareable dashboard that you can customize with widgets, scope, and your own layout. (20 min)
- Create a Kubernetes cluster**: Automate deployments and manage your containerized apps in a native Kubernetes experience. (5 min)
- Create and deploy an application**: Browse our starter kits, and then select one to jump start the process to create and deploy your app. (15 min)
- Build a web app with Watson Speech to Text**: Deploy a conversational interface compatible with any application, device, or channel. (7 min)

On the right, there's a user profile for "Sankara Subramaniyan" with options to upgrade account, log in to CLI and API, guided tours, privacy, feedback, and log out. Below the profile is a "For you" section with links to host website assets in Cloud Object Storage and accelerate delivery of static files. At the bottom, there's a search bar and a taskbar with various icons.

2. Navigate to cloud foundry and create a service.

Select a pricing plan
Displayed prices do not include tax. Monthly prices shown are for country or location: [United States](#)

Plan	Features	Pricing
Lite	Lite apps are free You get up to 256 MB of memory while you work on your apps.	Free
	Lite apps sleep after 10 days of development inactivity.	
Standard	256 MB+	\$0.07 USD/GB-Hour

Configure your resource
Select a runtime

- [**java** Liberty for Java™](#)
- [**.js** SDK for Node.js™](#)
- [**.net** ASP.NET Core](#)
- [**.go** Go Community](#)
- PHP**
- [Python](#)
- [Ruby](#)
- [.rb Ruby Community](#)
- [.swift Runtime for Swift](#)

Type here to search

Cloud Foundry App
Location: London
Plan: Lite
Runtime: PHP
Domain: eu-gb.cf.appdomain.cloud
Org: sankara@student.tce.edu
Space: dev

Create Add to estimate View terms FEEDBACK

Select a runtime

- [**java** Liberty for Java™](#)
- [**.js** SDK for Node.js™](#)
- [**.net** ASP.NET Core](#)
- [**.go** Go Community](#)
- .php PHP Community**
- [.py Python Community](#)
- [.rb Ruby Community](#)
- [.swift Runtime for Swift](#)

App name
sankara's webapp

Host name
sankara-webapp

Domain
eu-gb.cf.appdomain.cloud

Choose an organization
sankara@student.tce.edu

Choose a space
dev

Tags [\(1\)](#)
Examples: env:dev, version-1

Type here to search

Cloud Foundry App
Location: London
Plan: Lite
Runtime: PHP
App name: sankara's webapp
Host name: sankara-webapp
Domain: eu-gb.cf.appdomain.cloud
Org: sankara@student.tce.edu
Space: dev

Create Add to estimate View terms FEEDBACK

3. Our service home page is displayed

WhatsApp | IBM Cloud Foundry | Cloud Foundry | IBM | Application Details - IBM Cloud | https://yajithvishwa.eu-gb.cfapps.io/

IBM Cloud | Search resources and offerings... | Catalog | Docs | Support | Manage | Yajith Vishwa S...

Resource list / yajithvishwa | Running | Visit App URL | Add tags | Details | Actions...

Getting started

Overview

Runtime

Connections

Logs

API Management

Autoscaling

Instances

Health

100%

1/1 instance(s) are running

Instances

1

MB memory per instance

0 — 64

Runtime

PHP

256 Total MB allocation

0 MB still available

Used Free

Runtime cost

Current and estimated cost excludes connected services.

\$ 0.00

Estimated total for billing period

May 1, 2021 - May 31, 2021

Connections (0)

No services are connected to this app

Windows Start | Search for anything | Taskbar icons | System tray: ENG 10-05-2021

4. Enable continuous delivery

WhatsApp | IBM Cloud Foundry | Cloud Foundry | IBM | Application Details - IBM Cloud | https://yajithvishwa.eu-gb.cfapps.io/

IBM Cloud | Search resources and offerings... | Catalog | Docs | Support | Manage | Yajith Vishwa S...

Toolchains / yajithvishwa | Details | Actions...

Overview

Connections

Manage

Continuous Delivery service required:

Cloud Foundry org-based toolchains are deprecated. To ensure uninterrupted use, add a Continuous Delivery Service to a resource group and create this toolchain again within that resource group. Alternatively, you can contact IBM Support to request the creation of a Continuous Delivery service instance in this toolchain's org.

Think

Issues yajithvishwa

Code

Git yajithvishwa

Deliver

Delivery Pipeline yajithvishwa

Eclipse Orion Web IDE

ASK A QUESTION

Windows Start | Search for anything | Taskbar icons | System tray: ENG 10-05-2021

5. Create a repository for our service

The screenshot shows the IBM Cloud interface for managing Git repositories. A new repository is being created with the following details:

- Server:** London (<https://eu-gb.git.cloud.ibm.com>)
- Repository type:** Clone
- Source repository URL:** https://cloud.ibm.com/conapi/template/phpHelloWorld/download/starter_code?manifest=applications%3A%
- Repository Owner:** sankara
- Repository Name:** sankarwebapp
- Checkboxes (selected):**
 - Make this repository private
 - Enable Issues
 - Track deployment of code changes

At the bottom, there are "Cancel" and "Create" buttons.

6. Create API Key

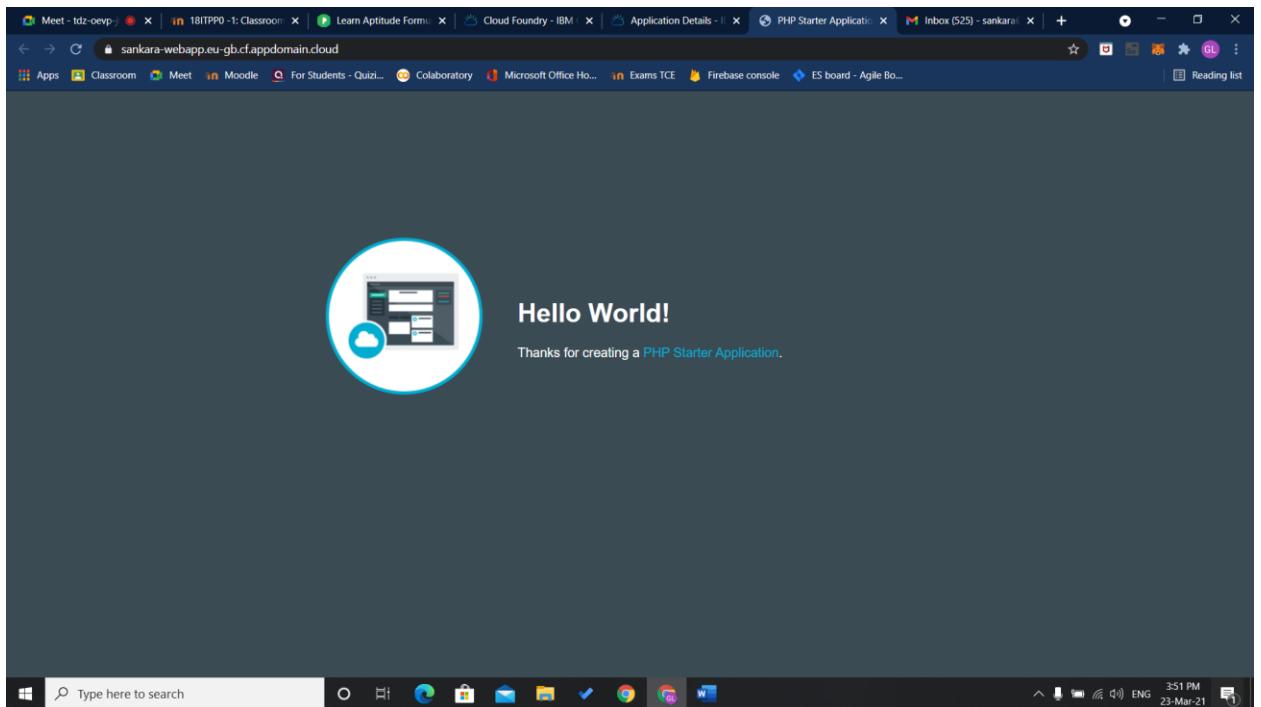
The screenshot shows the IBM Cloud interface for creating a Delivery Pipeline. The pipeline is being configured with the following details:

- Select Region:** London
- Select a resource group:** Default
- Description:** Pipeline for sankara's webapp

The "Delivery Pipeline" tab is selected. The pipeline status is shown as "Required".

At the bottom, there are "Cancel" and "Create" buttons.

7. Visit the sample page <https://yajithvishwa.eu-gb.cf.appdomain.cloud/>



8. Commit with the file that we created.

9. Build and Deploy the stages.

A screenshot of the IBM Cloud Delivery Pipeline interface. The pipeline is named 'sankara CD | Delivery Pipeline'. It consists of two stages: 'Build Stage' and 'Deploy Stage'.

- Build Stage:** Status: STAGE PASSED. Last commit by Sankara Subra... 3m ago. Update index.php v2. Jobs: Build (Passed now). Last execution result: Build 3.
- Deploy Stage:** Status: STAGE PASSED. Stage: Build Stage / Job: B... Build 3. Jobs: Deploy (Passed now). Last execution result: sankara's webapp View console. Build 3.

The interface includes a search bar, navigation tabs (Catalog, Docs, Support, Manage), and a sidebar with an 'ASK A QUESTION' button. The bottom of the screen shows a Windows taskbar.



User Profile Card



Yajith Vishwa S

3rd Year IT Dept
Thiagarajar College of Engineering

Contact

A user profile card for Yajith Vishwa S. It features a photo of a young boy standing outdoors. Below the photo, the name "Yajith Vishwa S" is displayed in bold. Underneath the name, it says "3rd Year IT Dept" and "Thiagarajar College of Engineering". At the bottom of the card is a black button labeled "Contact".

When we provide correct credentials a toast message is displayed indicating the successful login.

Result

Thus, a cloud service is created with PHP Runtime and deployed on cloud foundry.

Ex.No : 5

06.04.2021

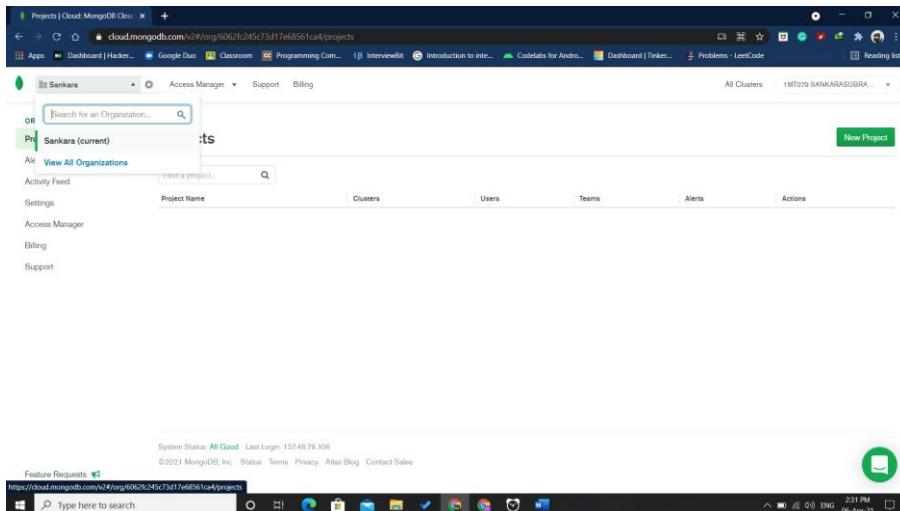
Implementation of DBaaS using MongoDB atlas

Aim

To create a cluster in mongodb atlas and access database via node.js.

Procedure

1. Create an account on <https://cloud.mongodb.com/> and login.
2. Create an organization.



3. Create a new project.

Create Project | Cloud: MongoDB

SANKARA > PROJECTS

ORGANIZATION

Projects

Alerts

Activity Feed

Settings

Access Manager

Billing

Support

Name Your Project

Add Members

Sankara's Cloud 1

Cancel Next

System Status: All Good Last Login: 1574679.106

©2021 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Feature Requests

Type here to search

2:32 PM ENG 06-Apr-21

SANKARA > PROJECTS

Create a Project

✓ Name Your Project Add Members

← Go Back Create Project

Add Members and Set Permissions

Invite new or existing users via email address...

sankara@student.tce.edu (you)

Project Owner

Cancel ← Go Back Create Project

4. Create a new cluster(Shared M0) after the project is created.

The screenshot shows the MongoDB Atlas Clusters page. On the left, there's a sidebar with sections for DATA STORAGE (Clusters, Triggers, Data Lake) and SECURITY (Database Access, Network Access, Advanced). The main area is titled 'Clusters' and has a sub-section 'SANKARA > SANKARA'S CLOUD 1'. It features a large green 'Create a cluster' button with a plus sign icon. Below it, a sub-section says 'Choose your cloud provider, region, and specs.' and includes a 'Build a Cluster' button. A note at the bottom says 'Once your cluster is up and running, live migrate an existing MongoDB database into Atlas with our Live Migration Service.' At the bottom of the page, there's a 'Feature Requests' section and a 'System Status: All Good' indicator.

5. Select the required fields and create cluster.

The screenshot shows the 'Create a Shared Cluster' page. At the top, it says 'Welcome to MongoDB Atlas! We've recommended some of our most popular options, but feel free to customize your cluster to your needs. For more information, check our documentation.' Below this is a 'Cloud Provider & Region' section. It shows 'AWS, N. Virginia (us-east-1)' selected. There are three tabs: AWS (selected), Google Cloud, and Azure. Under the AWS tab, it lists regions: ASIA (Mumbai (ap-south-1), Singapore (ap-southeast-1)), NORTH AMERICA (N. Virginia (us-east-1)★, Oregon (us-west-2)), EUROPE (Frankfurt (eu-central-1)★, Ireland (eu-west-1)★), and AUSTRALIA (Sydney (ap-southeast-2)). A 'FREE' badge is at the bottom left, and a note says 'Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.' A 'Create Cluster' button is at the bottom right. The Windows taskbar at the bottom shows various open applications like Google Chrome, Microsoft Word, and File Explorer.

Clusters

[Create a New Cluster](#)

Find a cluster... 

SANDBOX

Cluster0 Version 4.4

[CONNECT](#) [METRICS](#) [COLLECTIONS](#) [...](#)

CLUSTER TIER
M0 Sandbox (General)

REGION
AWS / N. Virginia (us-east-1)

TYPE
Replica Set - 3 nodes

LINKED REALM APP
None Linked

Your cluster is being created
New clusters take between 1-3 minutes to provision.



Clusters

[Create a New Cluster](#)

Find a cluster... 

SANDBOX

Cluster0 Version 4.4.4

[CONNECT](#) [METRICS](#) [COLLECTIONS](#) [...](#)

CLUSTER TIER
M0 Sandbox (General)

REGION
AWS / N. Virginia (us-east-1)

TYPE
Replica Set - 3 nodes

LINKED REALM APP
None Linked

This is a Shared Tier Cluster
If you need a database that's better for high-performance production applications, upgrade to a dedicated cluster.

[Upgrade](#)

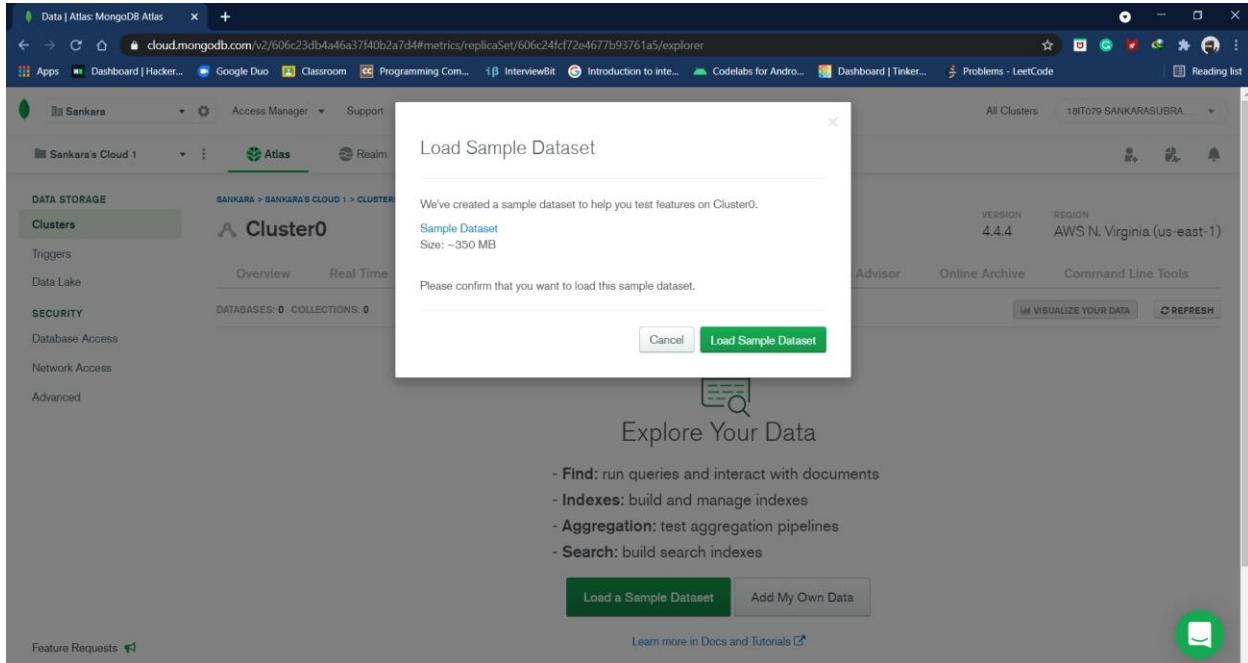
Operations R: 0 W: 0	100.0/s
Last 6 Hours	

Logical Size 0.0 B	512.0 ME max
Last 6 Hours	

Connections 0	500 max
Last 6 Hours	



6. Load the sample dataset into the collection



The screenshot shows the MongoDB Atlas interface with the "sample_airbnb.listingsAndReviews" collection selected. The top navigation bar includes "Find", "Indexes", "Schema Anti-Patterns", "Aggregation", and "Search Indexes". Below the navigation is a search bar with the filter "FILTER {\"filter\": \"example\"}" and buttons for "Find" and "Reset".

The main area displays the "QUERY RESULTS 1-20 OF MANY" for the document with _id: "10000546". The document fields include:

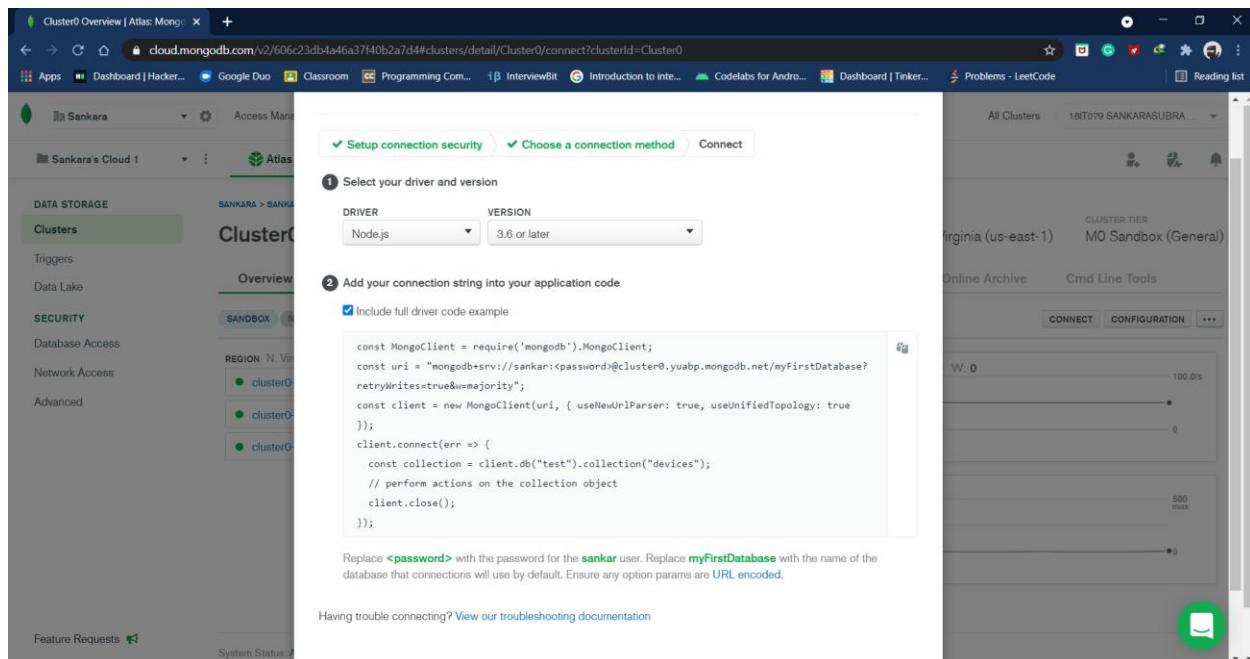
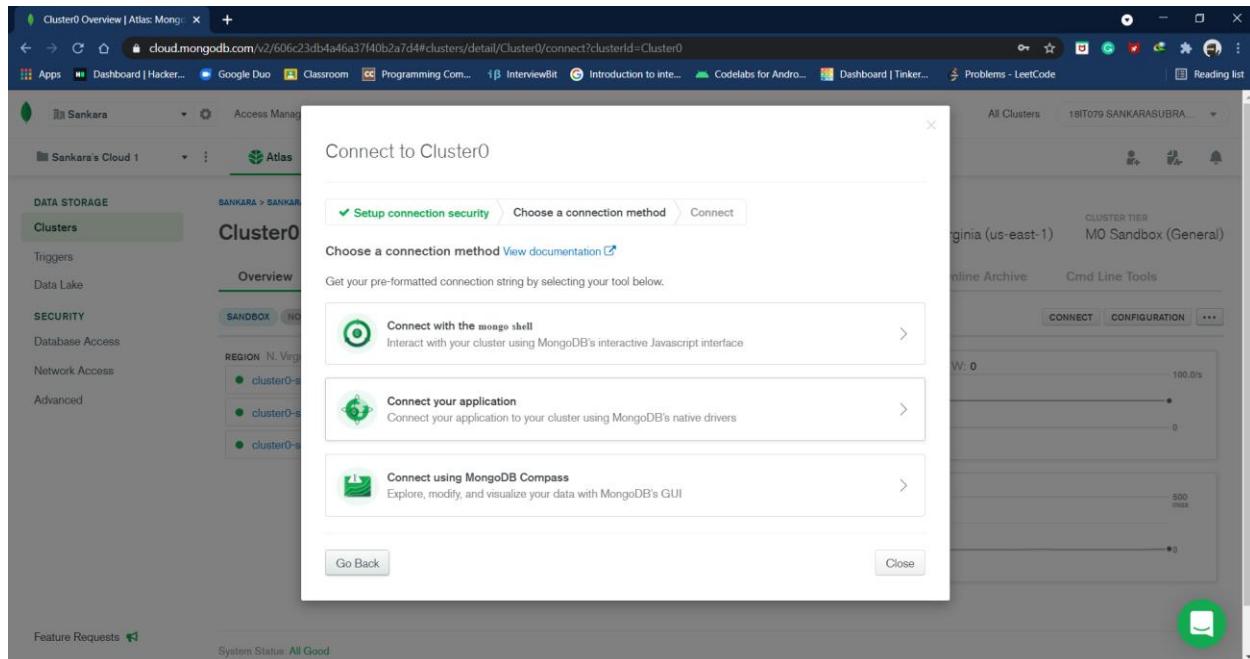
```
_id: "10000546"
listing_url: "https://www.airbnb.com/rooms/10000546"
name: "Ribeira - The Old Duplex"
summary: "Fantastic duplex apartment with three bedrooms, located in the histori..."
space: "Privileged views of the Douro River and Ribeira square, our apartment ..."
description: "Fantastic duplex apartment with three bedrooms, located in the histori..."
neighborhood_overview: "In the neighborhood of the river, you can find several restaurants as ..."
notes: "Lose yourself in the narrow streets and staircases zone, have lunch in..."
transit: "Transport: • Metro station and S. Bento railway Sml; • Bus stop a 50 ..."
access: "We are always available to help guests. The house is fully available t..."
interaction: "Cot - 10 € / night Dog - € 7,5 / night"
house_rules: "make the house your home..."
property_type: "House"
room_type: "Entire home/apt"
bed_type: "Real Bed"
minimum_nights: "2"
maximum_nights: "30"
cancellation_policy: "moderate"
last_scraped: "2019-02-16T05:00:00.000+00:00"
calendar_last_scraped: "2019-02-16T05:00:00.000+00:00"
first_review: "2010-01-01T05:00:00.000+00:00"
last_review: "2019-01-20T05:00:00.000+00:00"
accommodates: 8
```

7. Add IP address and Create database user.

The screenshot shows the MongoDB Atlas Cluster Overview page for 'Cluster0'. A modal window titled 'Create a Database User' is open. It contains two steps: 1. Add a connection IP address (with a note that an IP address has been added to the IP Access List) and 2. Create a Database User. Under 'Create a Database User', it says 'This first user will have **atlasAdmin** permissions for this project.' It includes fields for 'Username' (sankar) and 'Password' (a masked password). A 'Create Database User' button is at the bottom right. Below the modal, there's a 'Choose a connection method' link. The background shows the cluster configuration with 'CLUSTER TIER: virginia (us-east-1) MO Sandbox (General)' and various resource settings like W:0, R:0, and 500 max.

The screenshot shows the same MongoDB Atlas Cluster Overview page for 'Cluster0'. The modal window now displays the message 'You're ready to connect. Choose how you want to connect in the next step.' It lists the completed steps: 1. Add a connection IP address (IP added to IP Access List) and 2. Create a Database User (MongoDB user added with atlasAdmin permissions). It also includes a note: 'You'll need your MongoDB user's credentials in the next step.' Below the modal, there's a 'Choose a connection method' link. The background remains the same with cluster configuration details.

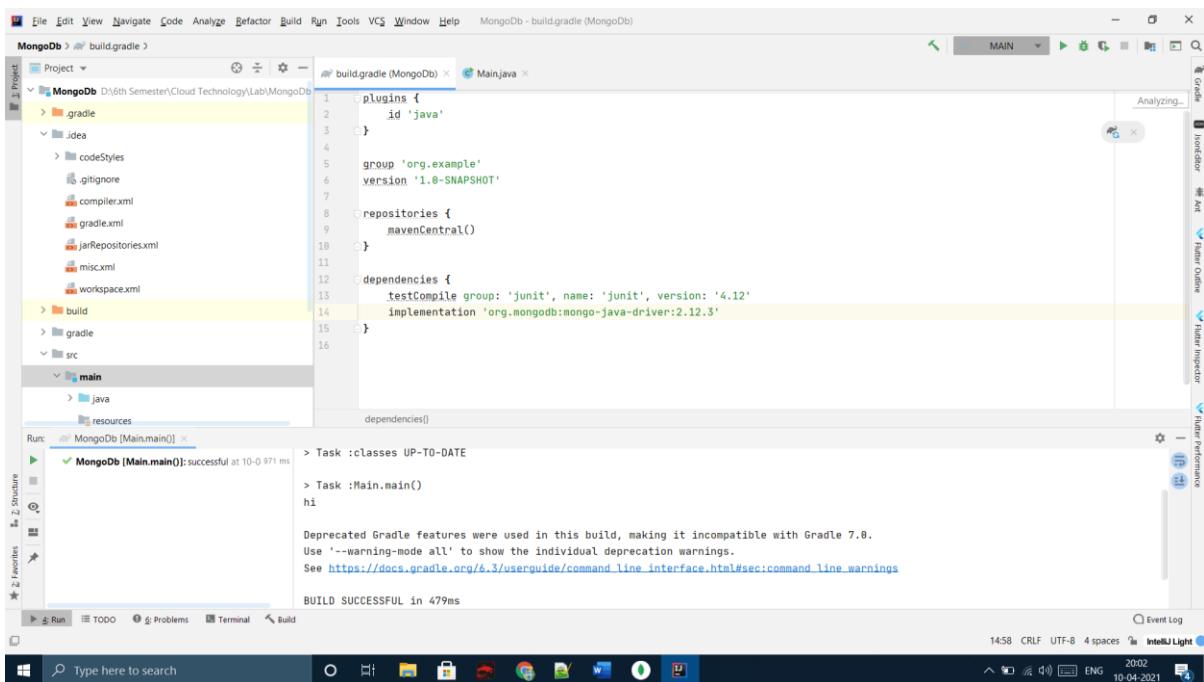
8. Connect your application to the cluster.



9. Perform some operation from node js and run to see the result.

```
Windows PowerShell
PS D:\College Works\2020 VI Sem\Cloud\Lab Exp> node connect.js
Client connecting...
PS D:\College Works\2020 VI Sem\Cloud\Lab Exp> node connect.js
Client connecting...
Databases:
-sample_airbnb
-sample_analytics
-sample_geospatial
-sample_mflix
-sample_restaurants
-sample_supplies
-sample_training
-sample_weatherdata
-admin
-local
PS D:\College Works\2020 VI Sem\Cloud\Lab Exp>
```

10. Import the necessary jar files for mongodb java driver.



Code for CRUD Operations(Select All):

```
import com.mongodb.*;
import org.json.JSONObject;

import java.util.Iterator;

public class Main {
    public static void main(String[] arg)
    {
        try {
            MongoClientURI uri = new MongoClientURI(
                "mongodb+srv://YYYY:XXXX@crop.ry3da.mongodb.net/myFirstDatabase?
retryWrites=true&w=majority");

            MongoClient mongoClient = new MongoClient(uri);
            DB db=mongoClient.getDB("Scriptons");
            DBCollection crop=db.getCollection("Water");
            DBCursor cursor=crop.find();
            JSONObject jsonObject=new JSONObject(cursor.next().toString());
            JSONObject
            water=jsonObject.getJSONObject("yajith").getJSONObject("water_level");
            Iterator<String> iterator=water.keys();
            while (iterator.hasNext())
            {
                String keys=iterator.next();
                System.out.println(keys+" "+water.getString(keys));
            }
        }
    }
}
```

```
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Code for CRUD Operations(Insert):

```
import com.mongodb.*;
```

```
import com.mongodb.client.MongoCollection;
```

```
import org.bson.Document;
```

```
import org.bson.types.ObjectId;
```

```
import org.json.JSONObject;
```

```
import java.util.ArrayList;
```

```
import java.util.HashMap;
```

```
import java.util.Iterator;

import java.util.List;
public class Main {

    public static void main(String[] arg)

    {

        try {

            MongoClientURI uri = new MongoClientURI(
                "mongodb+srv://yajith:vishwa@crop.ry3da.mongodb.net/myFirstDatabase?retryWrites=true&w=majority");

            MongoClient mongoClient = new MongoClient(uri);

            DB db=mongoClient.getDB("Scriptons");

            DBCollection crop=db.getCollection("Water");

```

```
HashMap<String,String> map=new HashMap<>();  
  
map.put("field1","20");  
  
map.put("field2","100");  
  
map.put("field3","10");  
  
map.put("field4","50");  
  
DBObject dbObject=new BasicDBObject("yajith",map);  
  
crop.insert(dbObject);  
  
}  
catch (Exception e) {  
    e.printStackTrace();  
}  
}
```

```
}
```

```
}
```

Code for CRUD Operations(Delete):

```
import com.mongodb.*;
```

```
import com.mongodb.client.MongoCollection;
```

```
import org.bson.Document;
```

```
import org.bson.types.ObjectId;
```

```
import org.json.JSONObject;
```

```
import java.util.ArrayList;
```

```
import java.util.HashMap;
```

```
import java.util.Iterator;
```

```
import java.util.List;
```

```
public class Main {
```

```
    public static void main(String[] args)
```

```
{
```

```
    try {
```

```
        MongoClientURI uri = new MongoClientURI(
```

```
"mongodb+srv://yajith:vishwa@crop.ry3da.mongodb.net/myFirstDatabase?retryWrites=true&w=majority");
```

```
MongoClient mongoClient = new MongoClient(uri);
```

```
DB db=mongoClient.getDB("Scriptons");
```

```
DBCollection crop=db.getCollection("Water");
```

```
HashMap<String,String> map=new HashMap<>();
```

```
map.put("field1","20");
```

```
map.put("field2","100");
```

```
map.put("field3","10");
```

```
map.put("field4","50");
```

```
DBObject dbObject=new BasicDBObject("yajith",map);
```

```
        crop.remove(dbObject);

    } catch (Exception e) {

        e.printStackTrace();

    }

}

}
```

Code for CRUD Operations(Create):

```
import com.mongodb.*;

import com.mongodb.client.MongoCollection;

import org.bson.Document;

import org.bson.types.ObjectId;
```

```
import org.json.JSONObject;  
  
import java.util.ArrayList;  
  
import java.util.HashMap;  
  
import java.util.Iterator;  
  
import java.util.List;  
  
  
  
  
  
public class Main {  
  
    public static void main(String[] arg)  
  
    {  
  
        try {
```

```
MongoClientURI uri = new MongoClientURI(  
  
"mongodb+srv://yajith:vishwa@crop.ry3da.mongodb.net/myFirstDatabase?retryWrites=true&w=majority");  
  
MongoClient mongoClient = new MongoClient(uri);  
  
DB db=mongoClient.getDB("Scriptons");  
  
DBCollection crop=db.getCollection("Water");  
  
HashMap<String,String> map=new HashMap<>();  
  
map.put("field1","20");  
  
map.put("field2","100");  
  
map.put("field3","10");  
  
map.put("field4","50");
```

```
DBObject dbObject=new BasicDBObject("yajith",map);

db.createCollection("Dummy",dbObject);

} catch (Exception e) {

    e.printStackTrace();

}

}

}
```

Screenshots:

Select All

The screenshot shows the IntelliJ IDEA interface with a Java project named "MongoDb". The code editor displays a Main.java file containing code to interact with a MongoDB database. The build log shows a successful build and execution of the Main.main() task, displaying a list of documents from a "Crop" collection.

```
try {
    MongoClientURI uri = new MongoClientURI(
        "mongodb+srv://vishwa@crop.ry3da.mongodb.net/myFirstDatabase?retryWrites=true&w=majority");
    MongoClient mongoClient = new MongoClient(uri);
    DB db = mongoClient.getDB("Scriptons");
    DBCollection cropdb = db.getCollection("Crop");
   DBObject dbObject = new BasicDBObject("Crop", "arid");
    DCursor cursor = cropdb.find();
    System.out.println(cursor.toArray());
} catch (Exception e) {
    e.printStackTrace();
}
```

Run: MongoDB [Main.main()]

Apr 10, 2021 8:38:33 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:4, serverValue:25806}] to crop-shard-00-01.ry3da.mongodb.net:27017
[{"_id": {"\$oid": "6071b4f2b7bac63468a843c4"}, "arid": ["wheat", "cotton", "corn (maize)", "millets", "pulses", "barley"], "Black_Soil": ["cotton", "rice", "sugar cane", "wheat", "jowar", "linseed", "sunflower", "cereal crops", "citrus fruits", "tomatoes", "tobacco", "groundnut"], "laterite": ["cotton", "rice", "wheat", "pulses", "tea", "coffee", "rubber", "coconut", "cashews"], "red": ["rice", "wheat", "sugarcane", "corn (maize)", "groundnut", "ragi(finger millet)", "potato", "oilseeds", "pulses", "millets", "mango", "orange", "citrus"]}]

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.3/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 6s
2 actionable tasks: 2 executed
20:38:33: Task execution finished 'Main.main()'.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "MongoDb" located at "D:\6th Semester\Cloud Technology\Lab\MongoDb".
- Code Editor:** The file "Main.java" contains Java code for interacting with a MongoDB database. It uses the MongoClient and DBCollection APIs to find documents in the "Water" collection of the "Scriptons" database.
- Terminal:** The terminal output shows the build process and the execution of the main method. It indicates a successful connection to the database and the execution of the query.
- System Tray:** The taskbar at the bottom shows various icons for system status and connectivity.

```

try {
    MongoClientURI uri = new MongoClientURI(
        "mongodb+srv://yajith:vishwa@crop.ry3da.mongodb.net/myFirstDatabase?retryWrites=true&w=majority");
    MongoClient mongoClient = new MongoClient(uri);
    DB db=mongoClient.getDB("Scriptons");
    DBCollection cropdb=db.getCollection("Water");
    DBCursor cursor=cropdb.find();
    JSONObject jsonObject=new JSONObject(cursor.next().toString());
    JSONObject water=jsonObject.getJSONObject("yajith").getJSONObject("water_level");
    Iterator<String> iterator=water.keys();
    while (iterator.hasNext())
    {
        String keys=iterator.next();
        System.out.println(keys+" "+water.getString(keys));
    }
}

```

Terminal Output:

```

Apr 11, 2021 9:06:42 AM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:4, serverValue:6747}] to crop-shard-00-02.ry3da.mongodb.net:27017
field1: 88
field3: 58
field2: 108
field4: 98

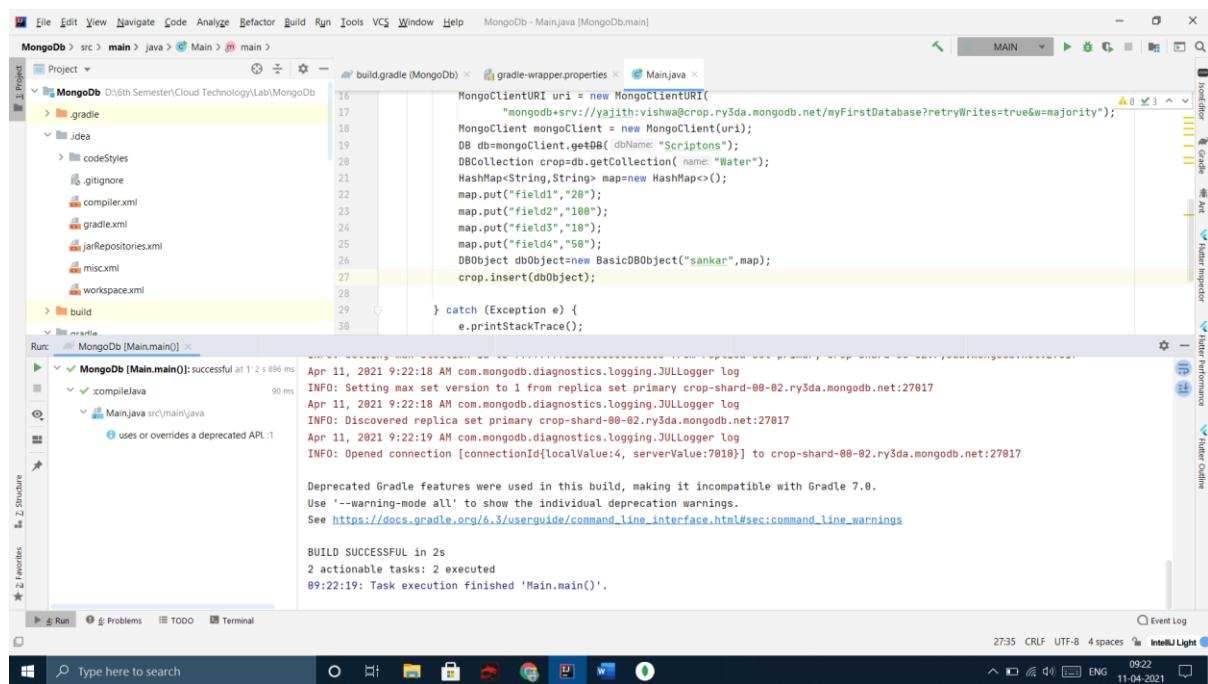
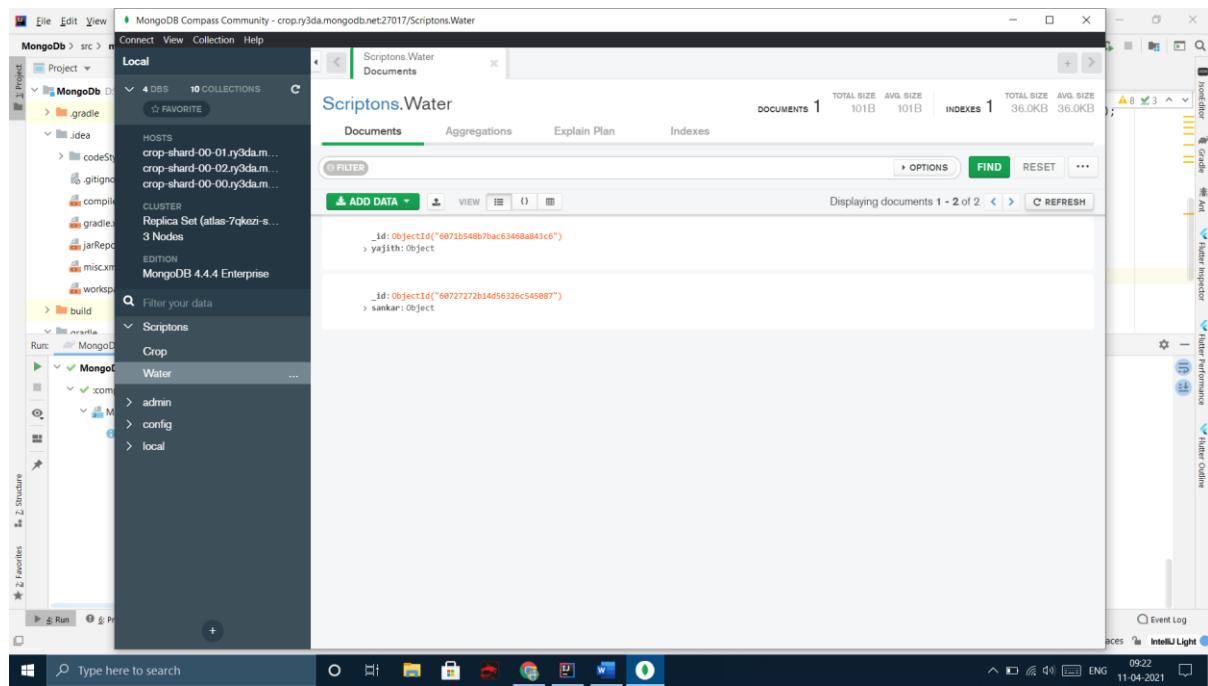
Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.3/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 2s
2 actionable tasks: 2 executed
09:06:43: Task execution finished 'Main.main()'.

```

System Tray:

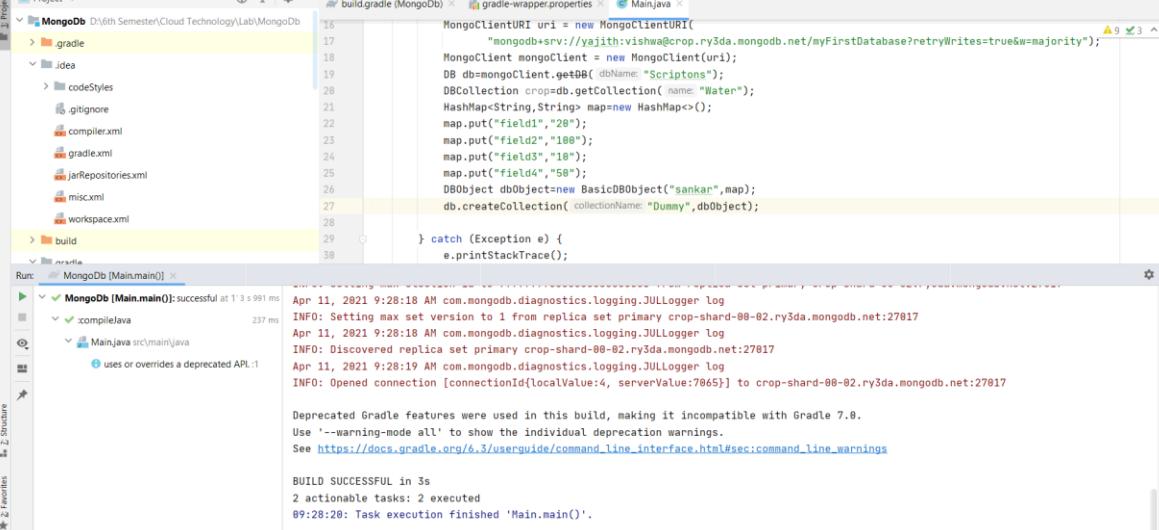
Insert Command



Create

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "MongoDb" and contains modules "gradle" and "Main".
- Build Process:** The "build.gradle" file is open, showing code to connect to a MongoDB database and insert a document. It includes a try-catch block for handling exceptions.
- Run/Debug:** A run configuration for "MongoDb [Main.main()]" is selected, showing successful execution times for compilation and execution.
- Output:** The terminal output shows the Gradle build process, including connection logs to the MongoDB database and a warning about deprecated Gradle features.
- Bottom Bar:** Shows tabs for Run, Problems, TODO, Terminal, and Event Log.



MongoClientURI uri = new MongoClientURI(
 "mongodb+srv://"+username+"@"+host+"/"+database+"?retryWrites=true&w=majority");
MongoClient mongoClient = new MongoClient(uri);
DB db=mongoClient.getDB(dbName: "Scriptors");
DBCollection crop=db.getCollection(name: "Water");
HashMap<String, String> map=new HashMap<>();
map.put("field1", "20");
map.put("field2", "100");
map.put("field3", "10");
map.put("field4", "50");
DBObject dboObject=new BasicDBObject("sankar", map);
db.createCollection(collectionName: "Dummy", dboObject);
}
} catch (Exception e) {
 e.printStackTrace();
}

MongoDb [Main.main()]

MongoDb [Main.main()]: successful at 1'3 s 991 ms
compileJava 237 ms

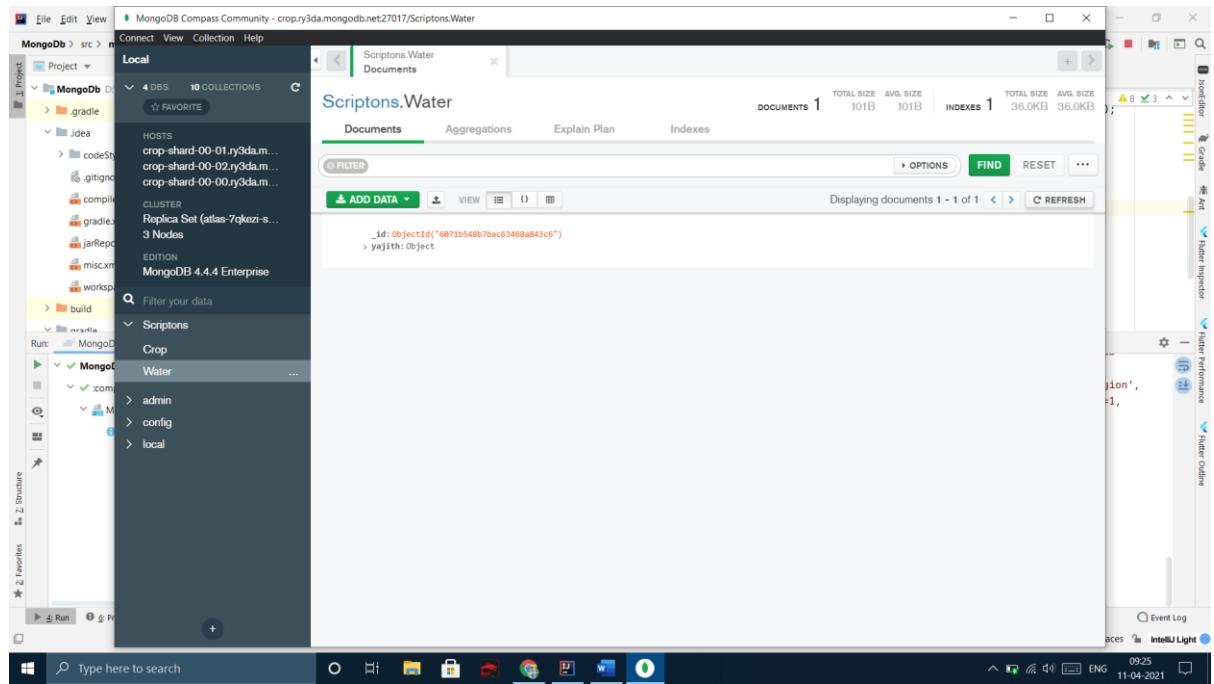
Main.java src/main/java
uses or overrides a deprecated API: 1

INFO: Setting mainVersion to 1 from replica set primary crop-shard-00-02.ry3da.mongodb.net:27017
INFO: Discovered replica set primary crop-shard-00-02.ry3da.mongodb.net:27017
INFO: Opened connection [connectionId{localValue:4, serverValue:7065}] to crop-shard-00-02.ry3da.mongodb.net:27017

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.3/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 3s
2 actionable tasks: 2 executed
09:28:20: Task execution finished 'Main.main()'.

Delete



Result

Thus, application is connected to mongoDb atlas and CRUD operations is performed and the result is obtained.

Ex.No : 6

27.04.2021

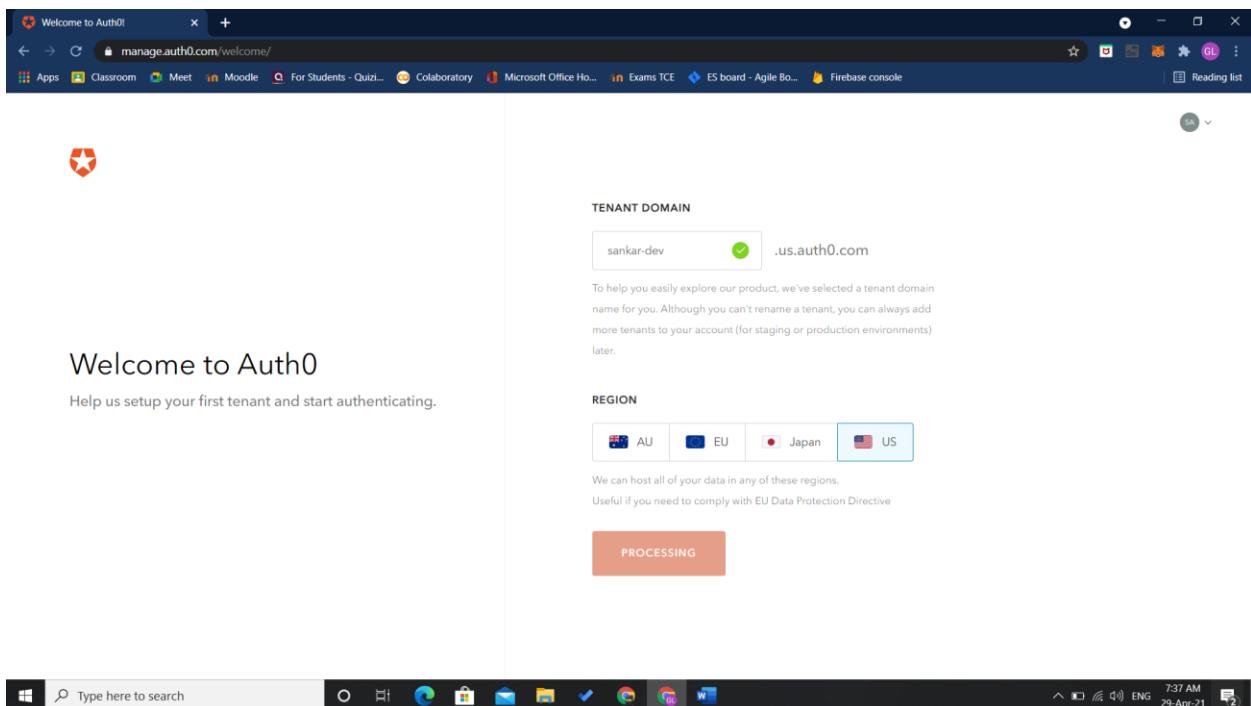
Implementation of IDaaS using Auth0

Aim

To create an identity as a service using auth0.

Procedure

1. Create auth0 account and get a domain.



Download the sample application and install the dependencies with composer.

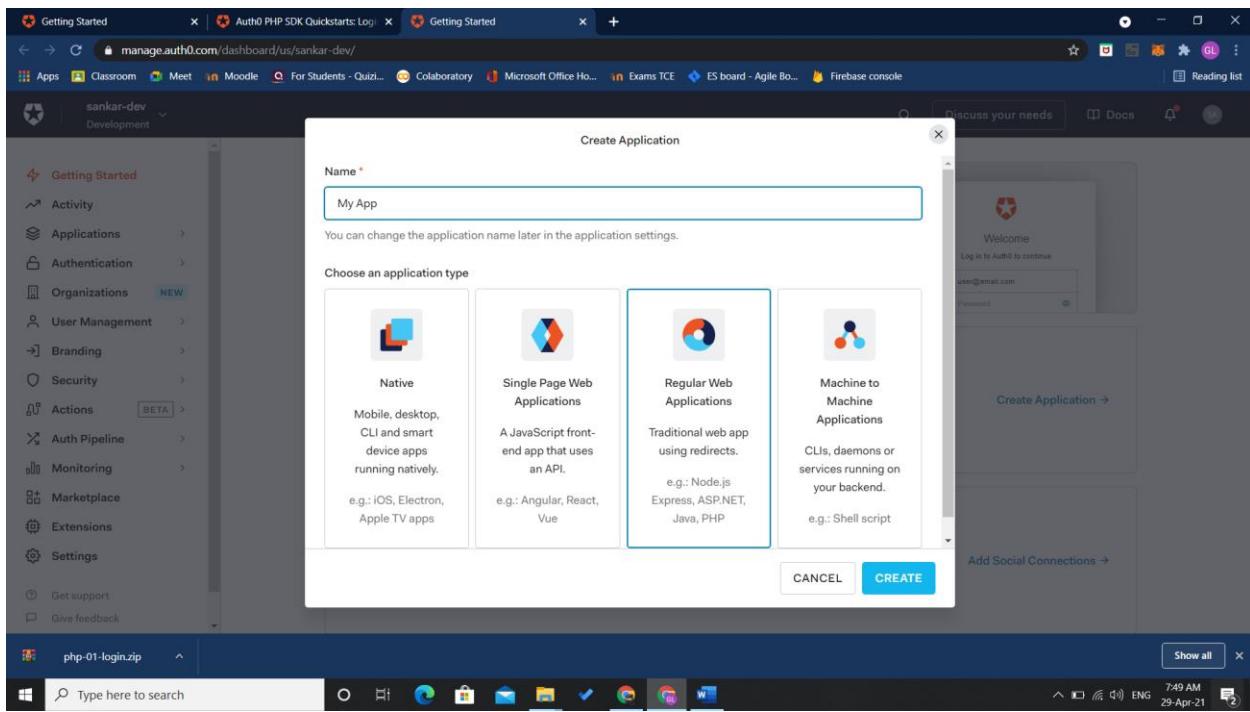
This tutorial demonstrates how to add authentication to your application.

I want to integrate with my app

- Configure auth
- Configure API for your auth
- Integrate authentication
- Integrate authorization

[VIEW ON GITHUB](#) [DOWNLOAD](#)

Create the application in application dashboard. Choose PHP application.

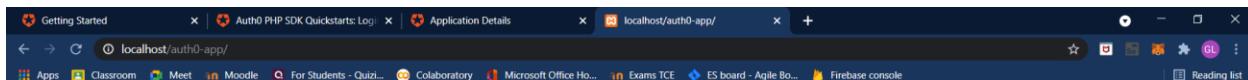


The screenshot shows the Auth0 Application Details page for an application named 'My App'. The left sidebar lists various settings sections: Getting Started, Activity, Applications (selected), APIs, SSO Integrations, Authentication (selected), Organizations (NEW), User Management, Branding, Security, Actions (BETA), Auth Pipeline, Monitoring, Marketplace, Extensions, and Settings. The main content area displays the 'Basic Information' section for 'My App'. It includes fields for Name (set to 'My App'), Domain (set to 'sankar-dev.us.auth0.com'), and Client ID (set to '4W9WdzH7AFdb3DgYa5uAh3FGKZhsqQgL'). A message at the top right encourages purchasing the Free Auth0 plan. The status bar at the bottom shows the date as 29-Apr-21 and the time as 7:51 AM.

Configure the allowed callback URLs.

The screenshot shows the 'Application URIs' section of the Auth0 Application Details page. It includes fields for 'Application Login URI' (set to 'https://myapp.org/login') and 'Allowed Callback URLs' (set to 'http://localhost/auth0-app/'). Below these fields, explanatory text discusses the purpose of these URLs and how they relate to user authentication. The left sidebar is identical to the previous screenshot, showing the 'Authentication' section is selected. The status bar at the bottom shows the date as 01-May-21 and the time as 9:14 AM.

Launch the Php app and use the authentication.



Auth0 Example

Zero friction identity infrastructure, built for developers

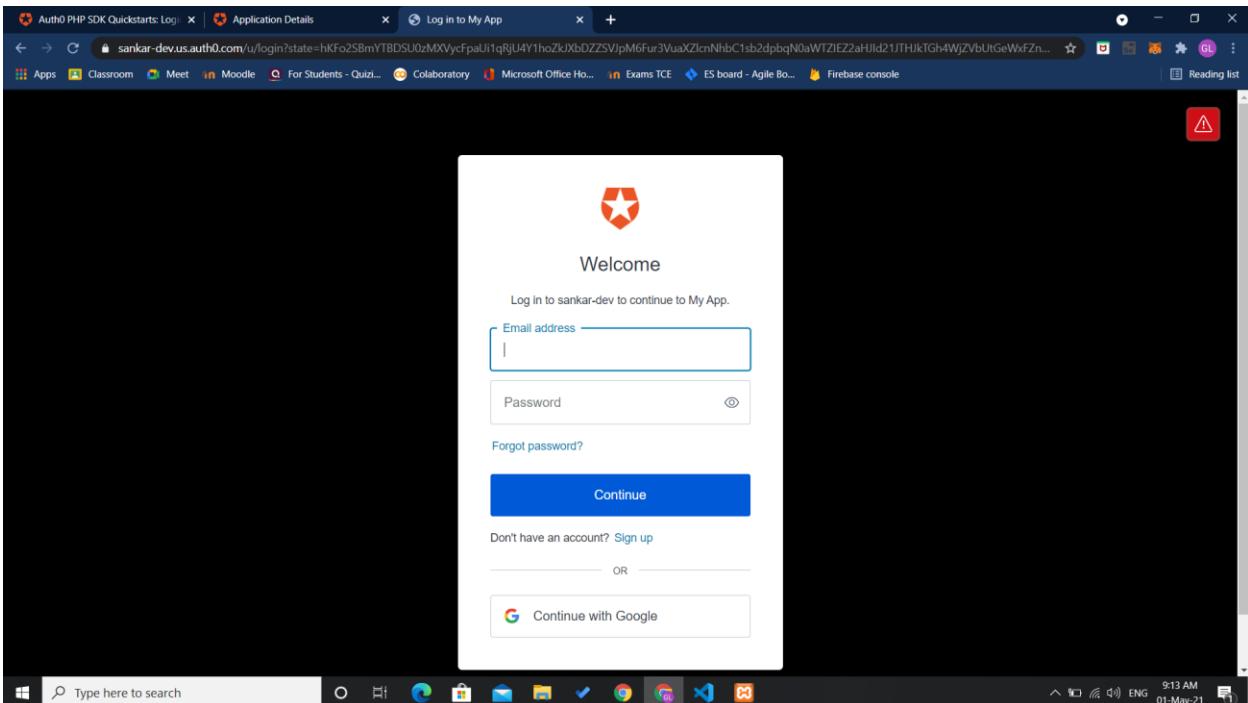
SIGN IN



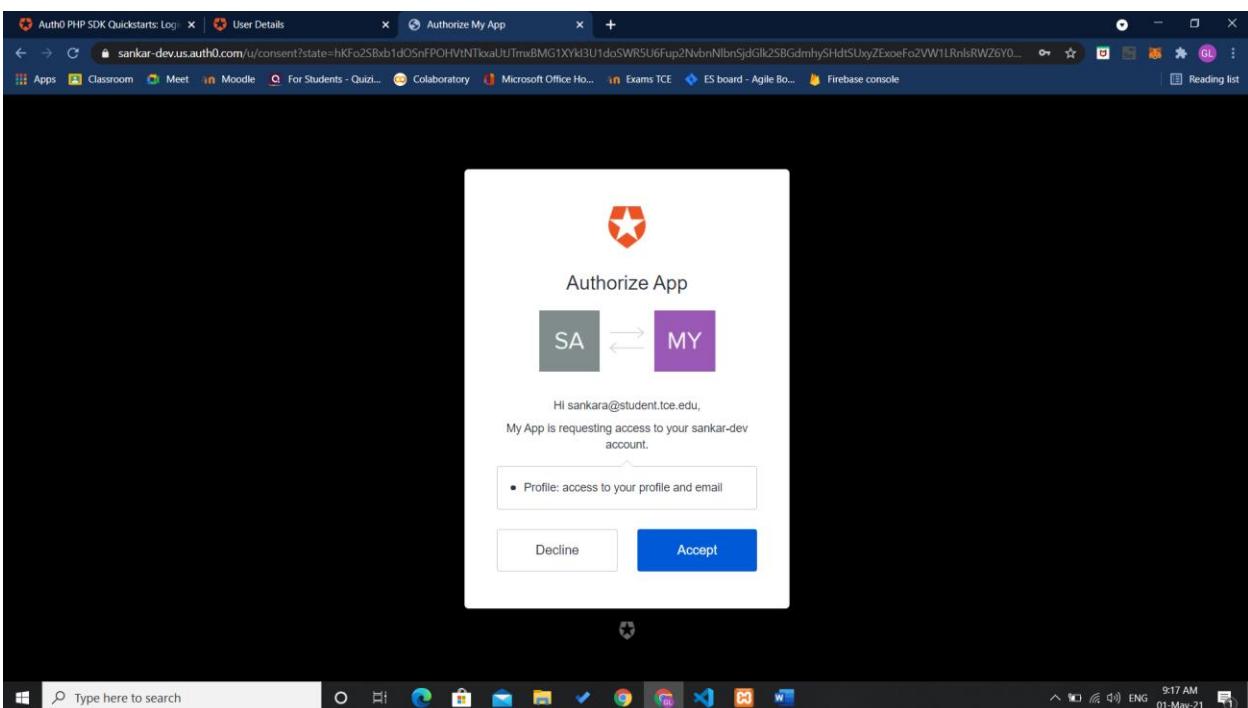
Create User in Auth0.

A screenshot of the Auth0 Management Dashboard. The left sidebar shows navigation options like 'Getting Started', 'Activity', 'Applications', 'Authentication', 'Organizations', 'User Management' (which is currently selected), 'Branding', 'Security', 'Actions (BETA)', 'Auth Pipeline', 'Monitoring', 'Marketplace', 'Extensions', and 'Settings'. The main content area has a message about a trial period and a 'Create user' dialog box. The dialog box contains fields for 'Email *' (sankara@student.tce.edu), 'Password *' (redacted), 'Repeat Password *' (redacted), and 'Connection *' (Username-Password-Authentication). There are 'CANCEL' and 'CREATE' buttons at the bottom. The status bar at the bottom shows the date and time as 01-May-21 9:16 AM.

Sign In to Auth0.



Accept the connection to connect to our App.



Login is successful.

The screenshot shows a web browser window with three tabs open: "Auth0 PHP SDK Quickstarts: Log", "User Details", and "localhost/auth0-app/?code=REsQAFLQH0FwVB_c&state=ac7d8bc5098fcbdbdbfb261712cf9902". The main content area displays "Login SuccessHello sankara@student.tce.edu!" followed by the Auth0 logo and a dark grey square containing the letters "SA". Below this is the text "Welcome sankara" and a blue "LOGOUT" button.



Result

Thus, application is connected to auth0 and authentication is performed.

Ex.No : 7
28.04.2021

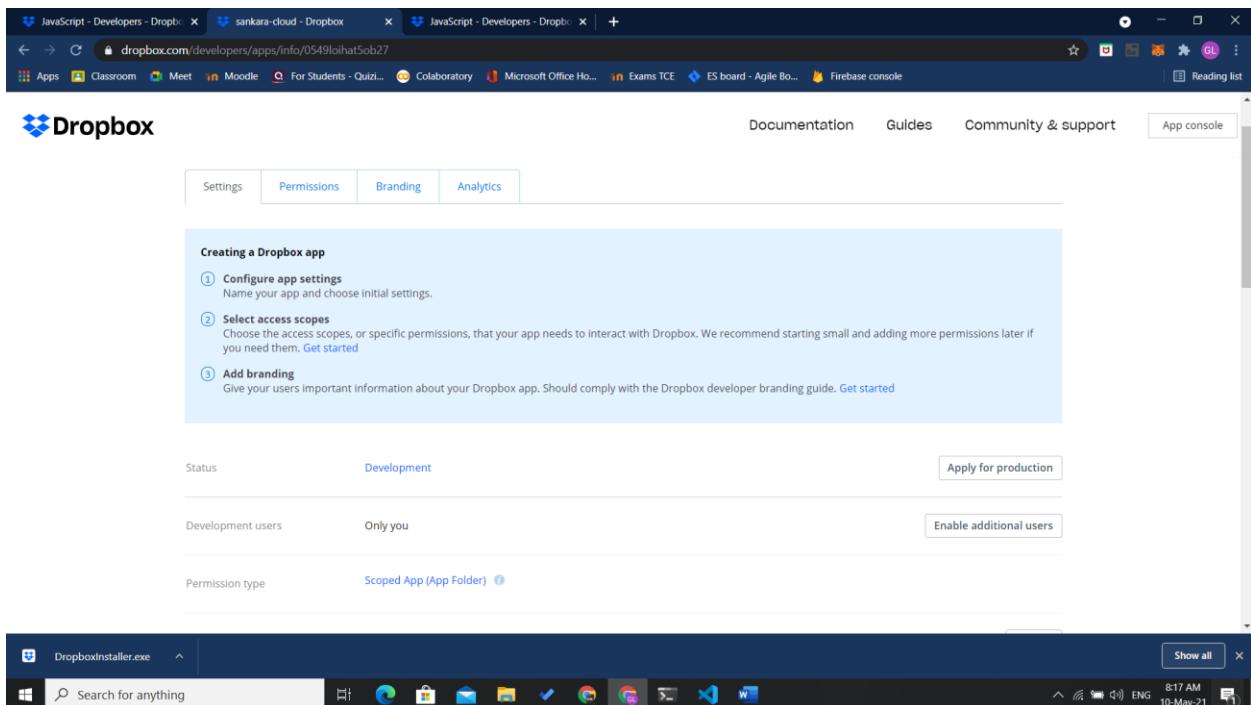
Implementation of Storage as a Service using DropBox

Aim

To create and deploy our node js application for accessing the dropbox api.

Procedure

1. Login to Dropbox account.
2. Create our application in dropbox.



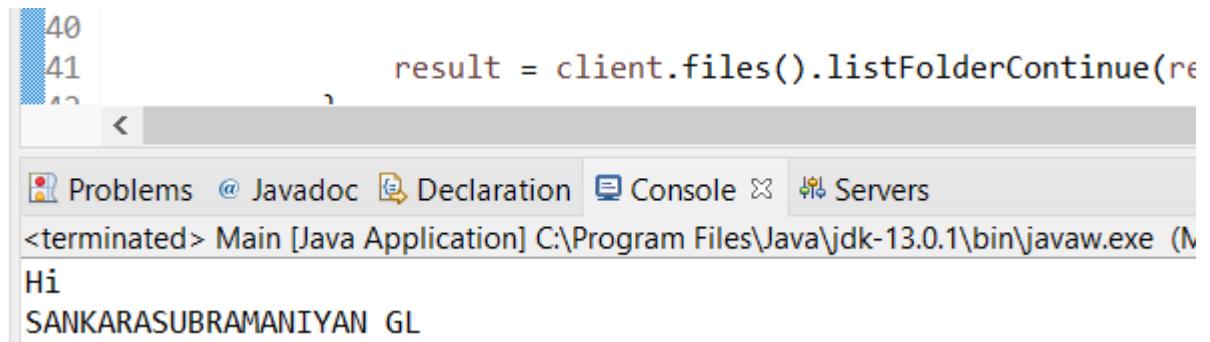
The screenshot shows the 'My apps' section of the Dropbox App Console. A single app, 'sankara-cloud', is listed. The app icon is a blue puzzle piece. The app details show: Status: Development and Permission type: Scoped App (App Folder). The user is logged in as SANKARASUBRAMANIYAN GL.

3. Generate Access Token.

4. Import the java project as an Existing Maven Project.

The screenshot shows the Eclipse IDE interface with the 'Import' dialog box open. The 'Select' tab is selected, displaying the 'Existing Maven Projects' option under the 'Existing Maven' wizard. The 'Outline' view on the right indicates there is no active editor. The desktop taskbar at the bottom shows various application icons.

5. Access the folder location.



The screenshot shows a Java code editor and a terminal window. The code editor has three lines of code visible:

```
40         result = client.files().listFolderContinue(re
41
42     ,
```

The terminal window shows the following output:

```
Problems @ Javadoc Declaration Console Servers
<terminated> Main [Java Application] C:\Program Files\Java\jdk-13.0.1\bin\javaw.exe (N
Hi
SANKARASUBRAMANIYAN GL
```

6. Generate Access

Result

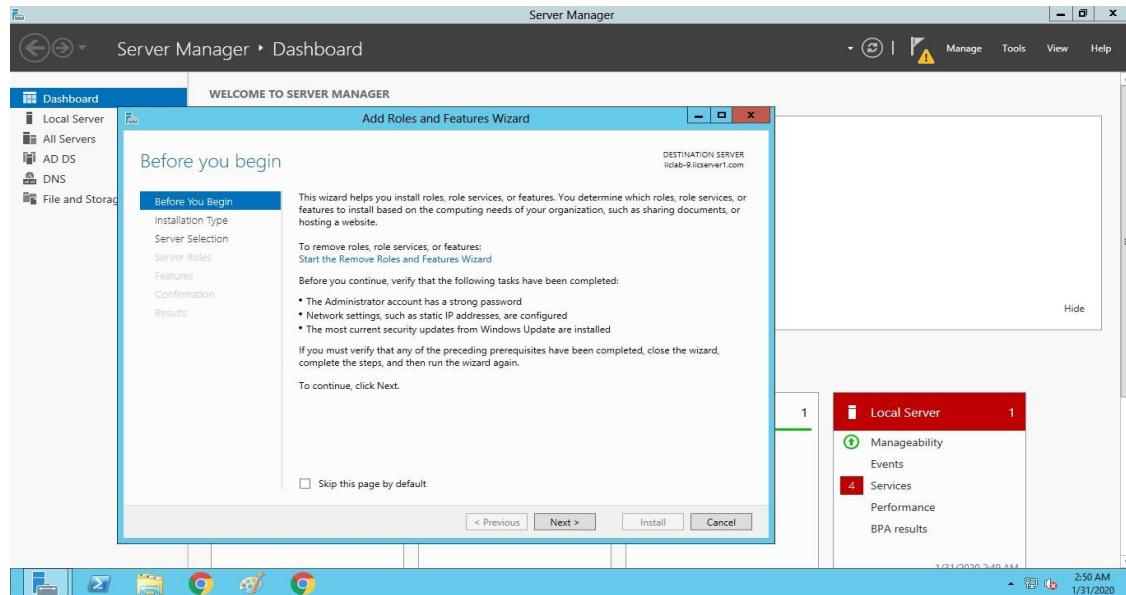
Thus, the dropbox service is accessed and the contents are retrieved.

Aim:

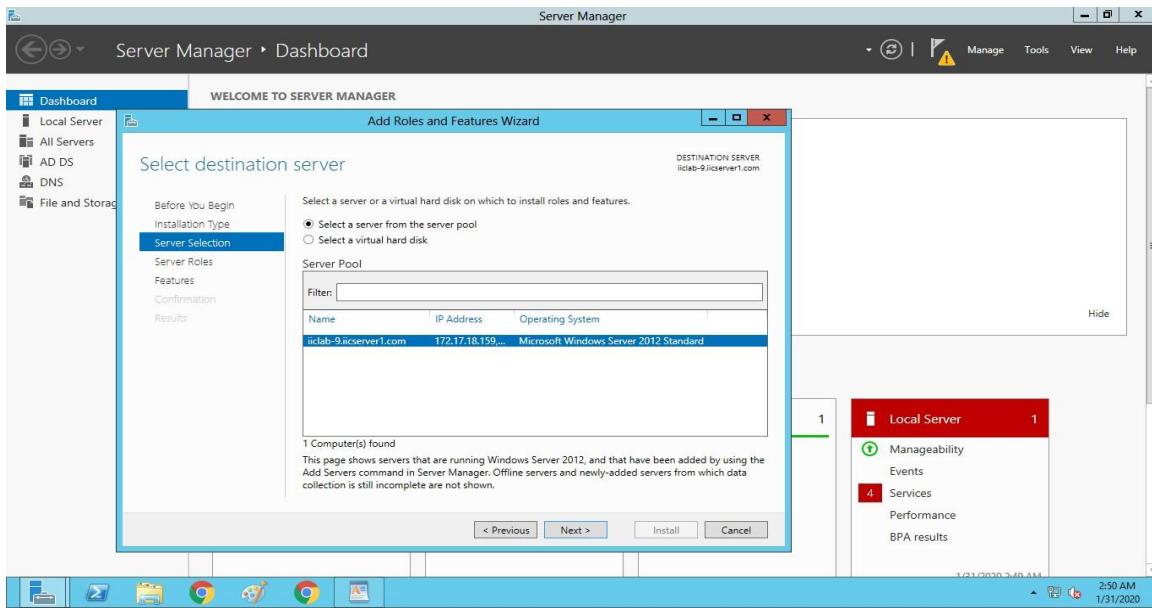
To implement the Migration of VM to another Host by using Hyper-V tool.

VM MIGRATION PROCEDURE :

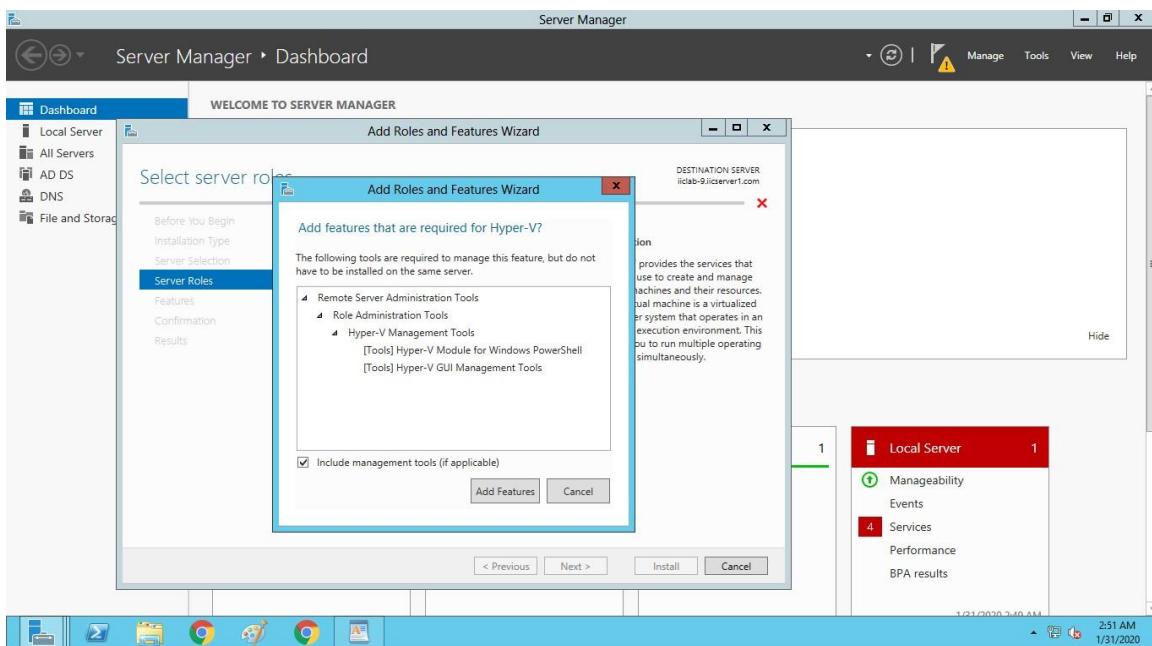
It is the process of moving a running virtual machine or application between different physical machines without disconnecting the client or application. Memory, storage and network connectivity of the virtual machine are transferred from the original guest machine to the destination.



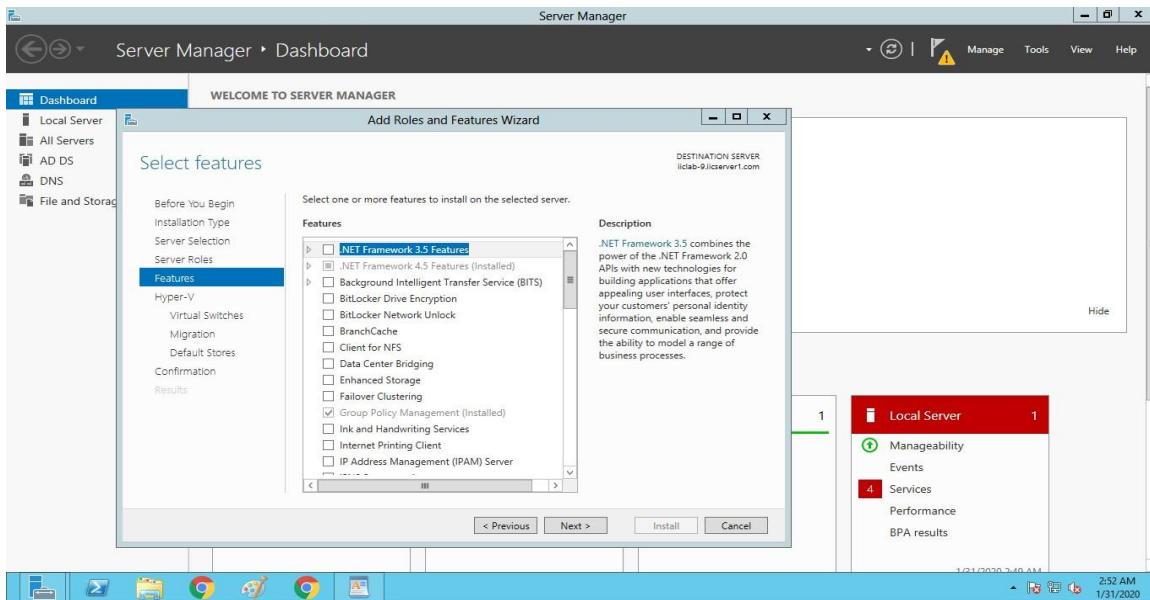
- Adding roles and features through the server manager for Hyper V services.



- Selecting the available server from the server pool

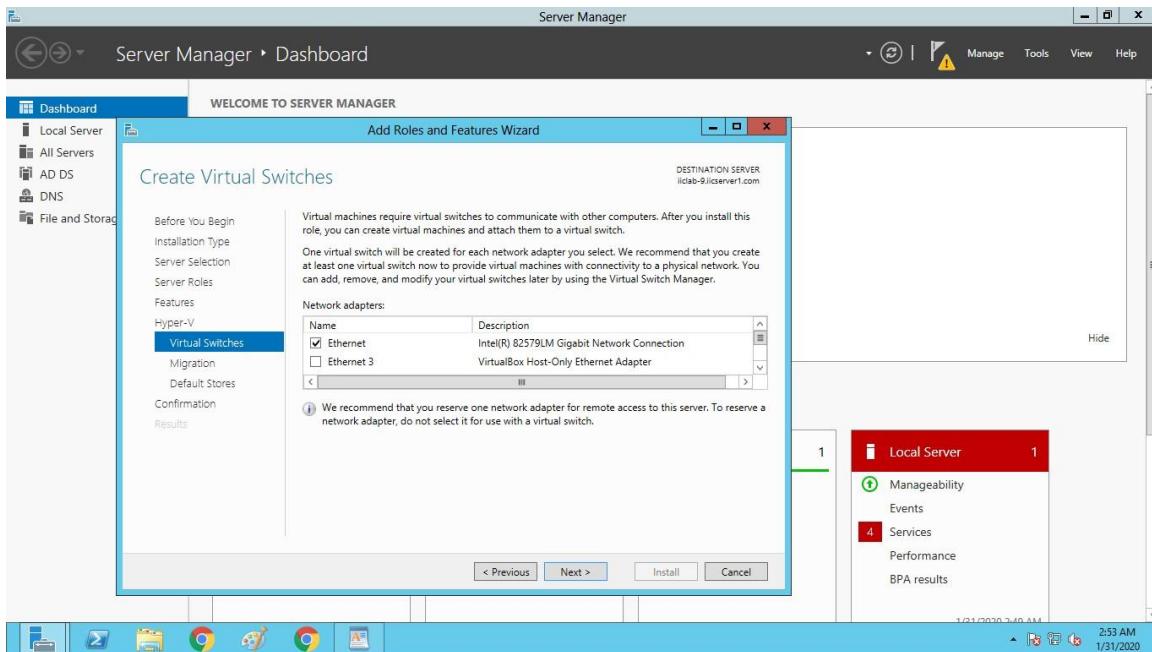


- Hyper V features will be added inorder to avail migration service from the server roles panel.

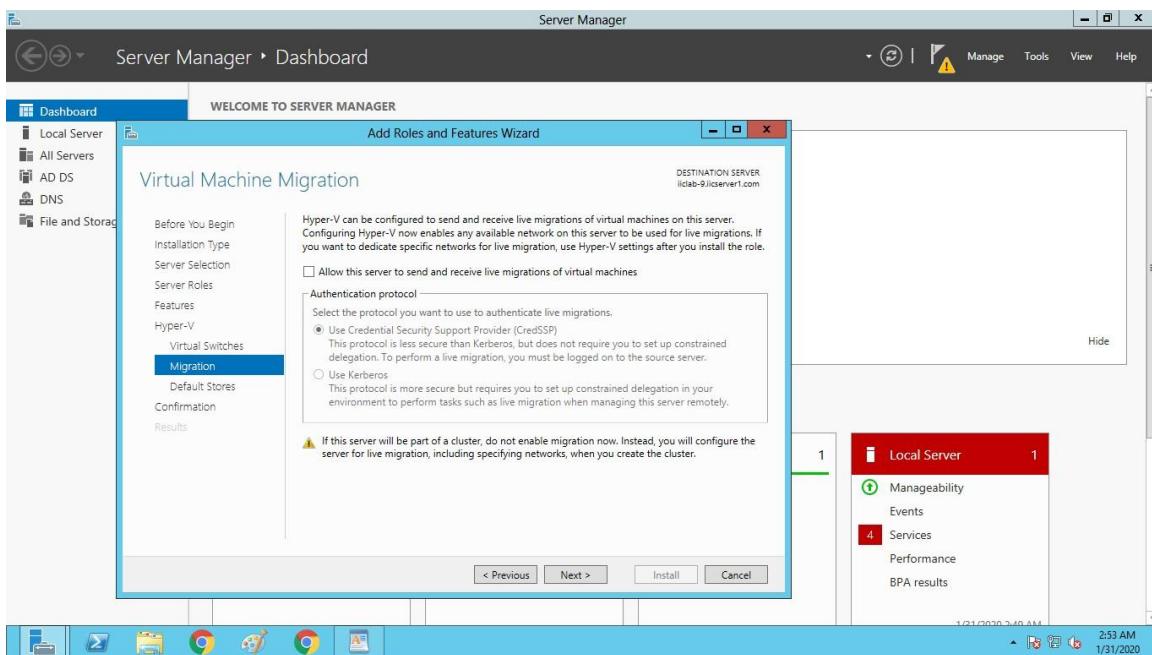


- From the Features panel, select the Net.Framework 3.5 which combines the power of the Net.Framework 2.0 with application.

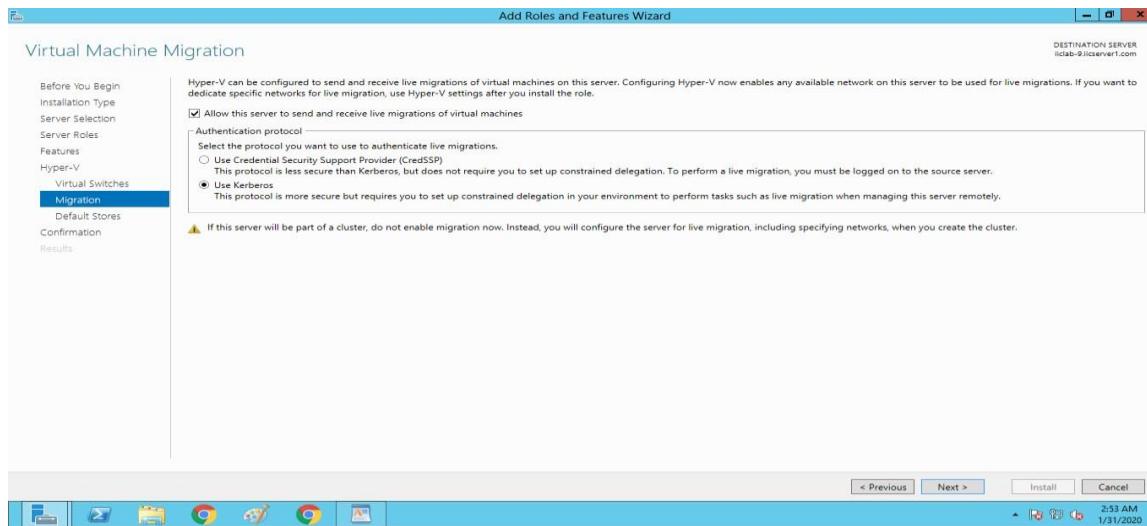
- Then, the Hyper V panel will be availed below the features panel.



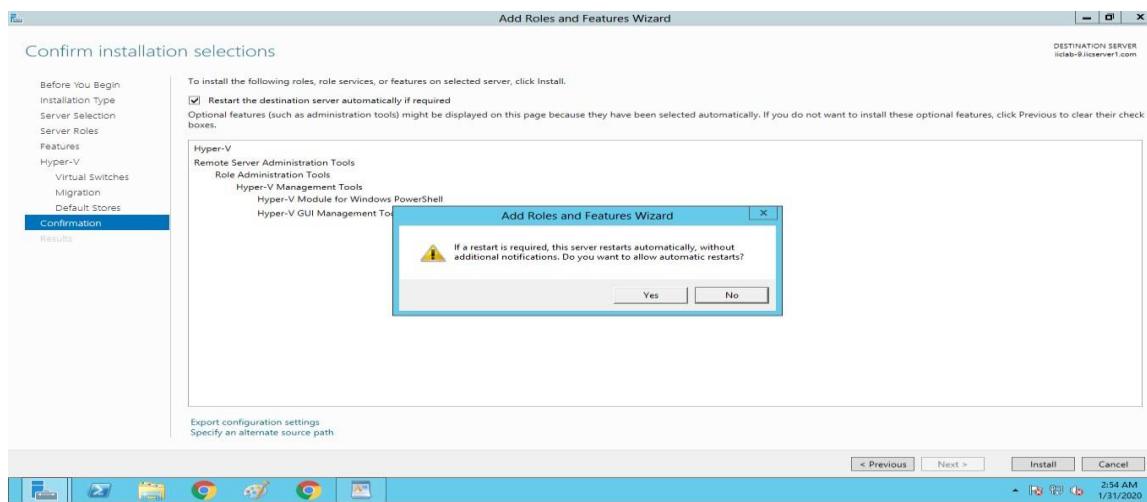
- From the virtual switches option, select the ethernet which helps to communicate with the other computers.



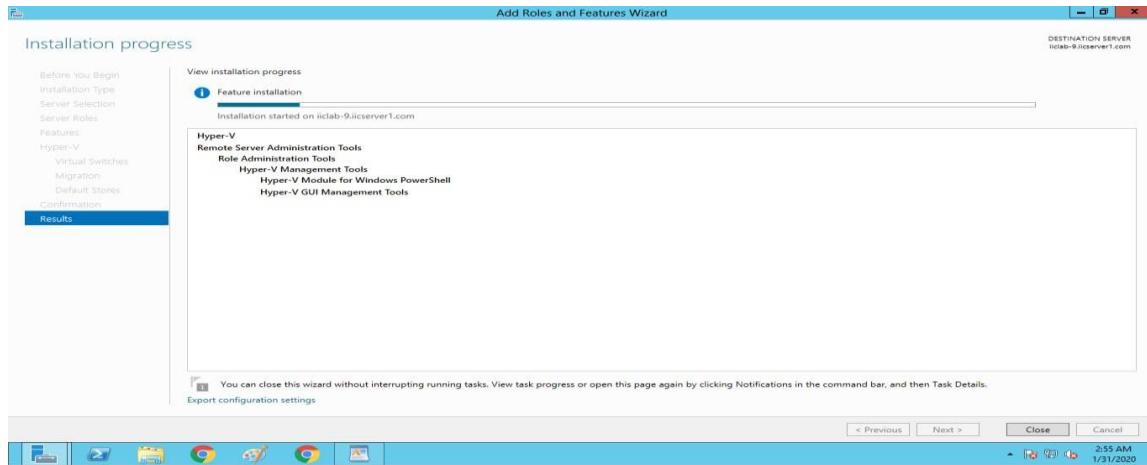
- Enable the migration services by checking the box for activating the live migrations of virtual machines.



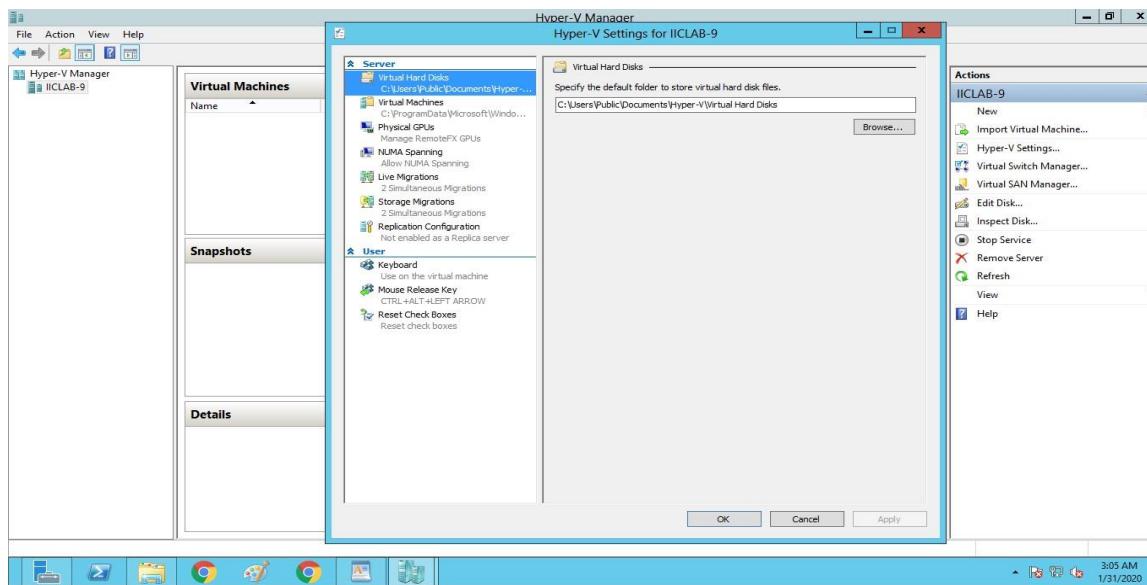
- Select Kerberos for authentication protocol



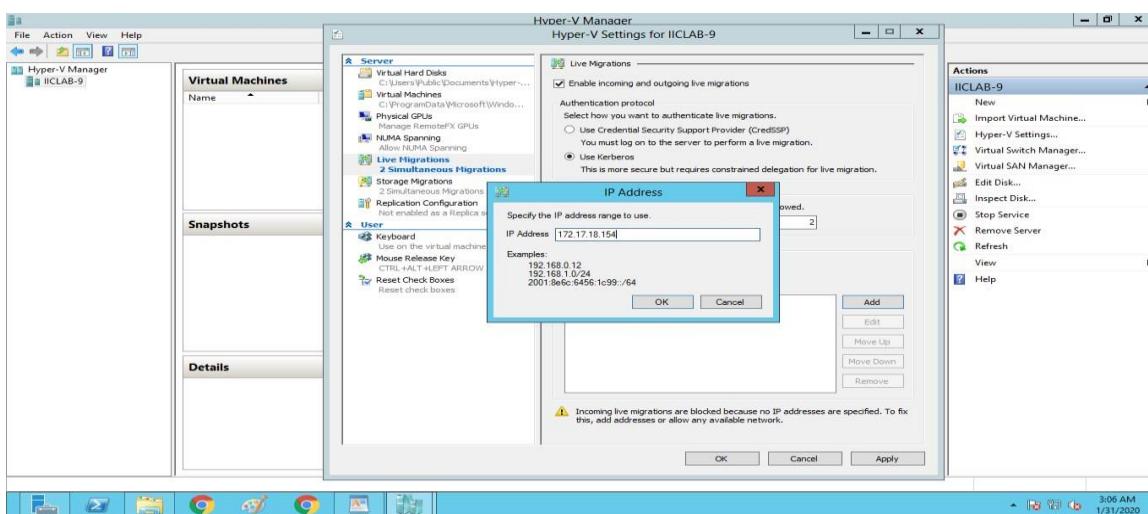
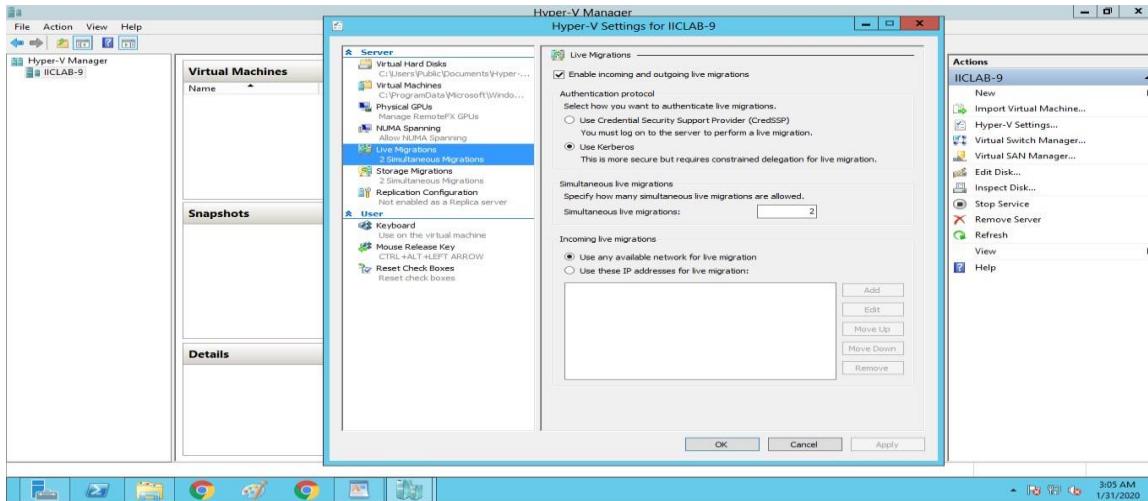
- Confirm the settings and the server will be restarted automatically.

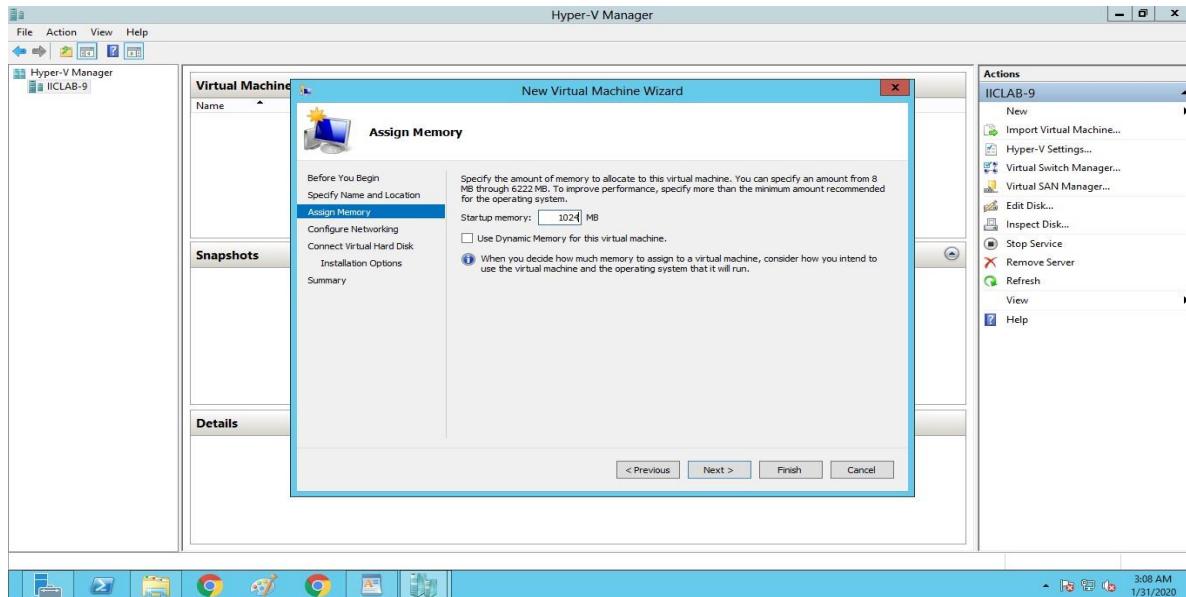


- Installation in progress.

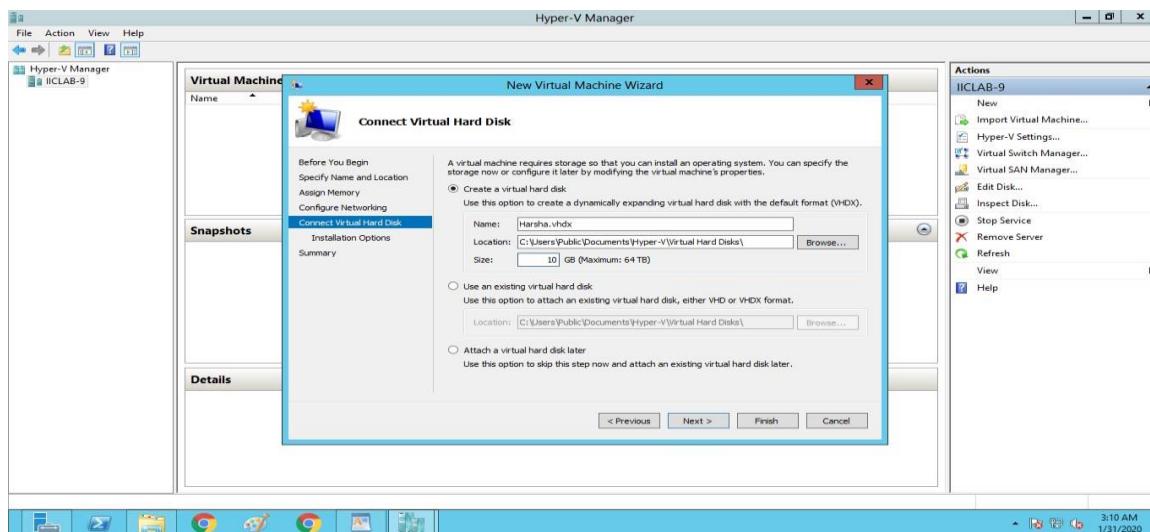


- From the Hyper V settings, Go to the live migration Select the Use IP address operation

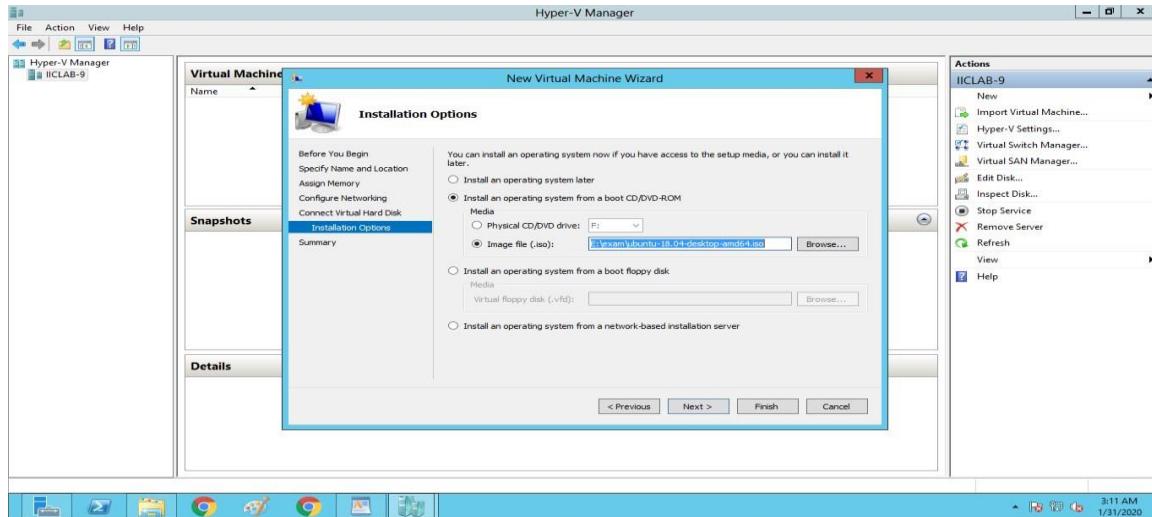




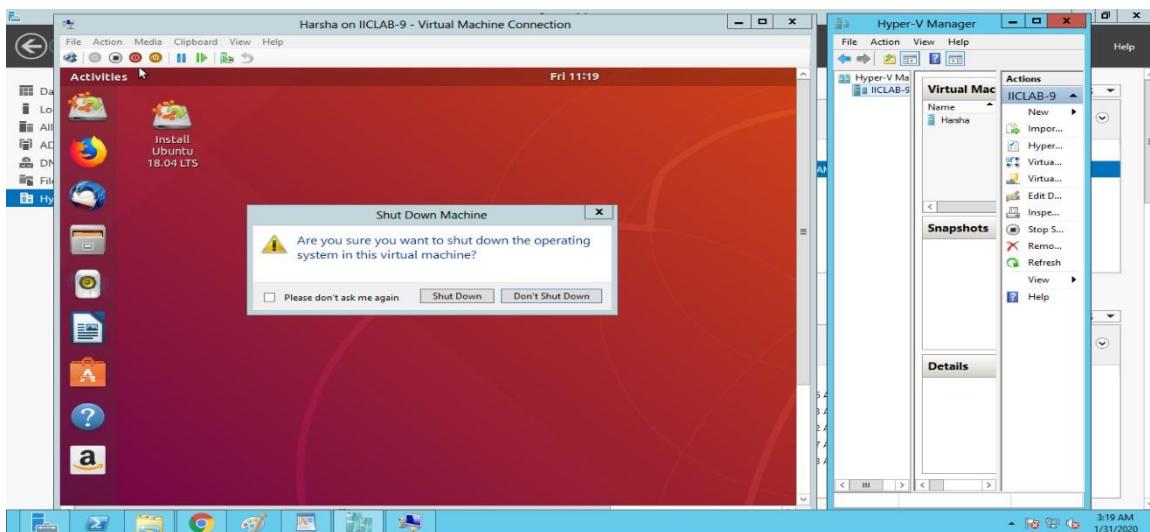
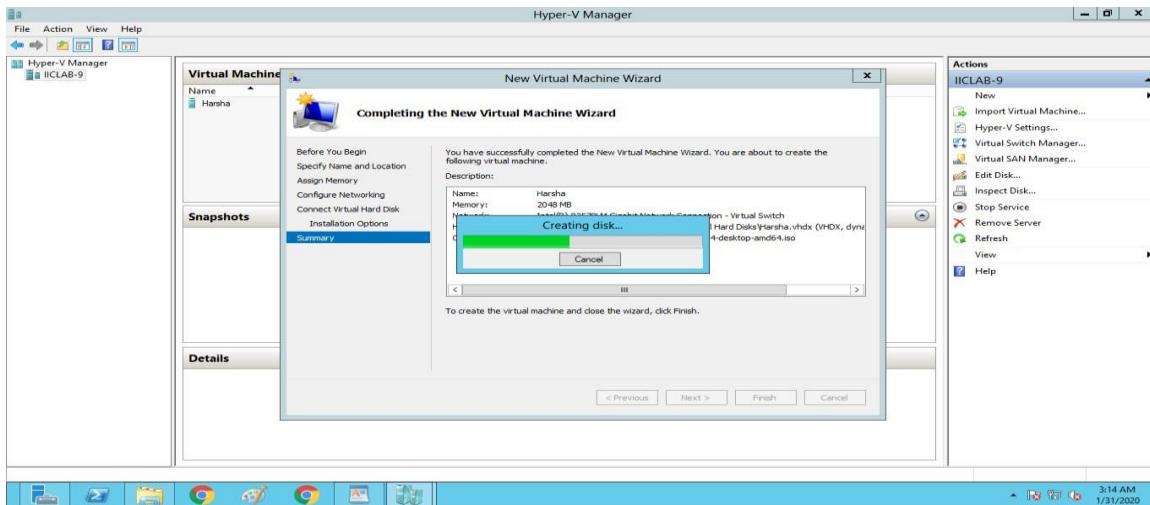
- Create a new virtual machine and assign the RAM memory as 1GB



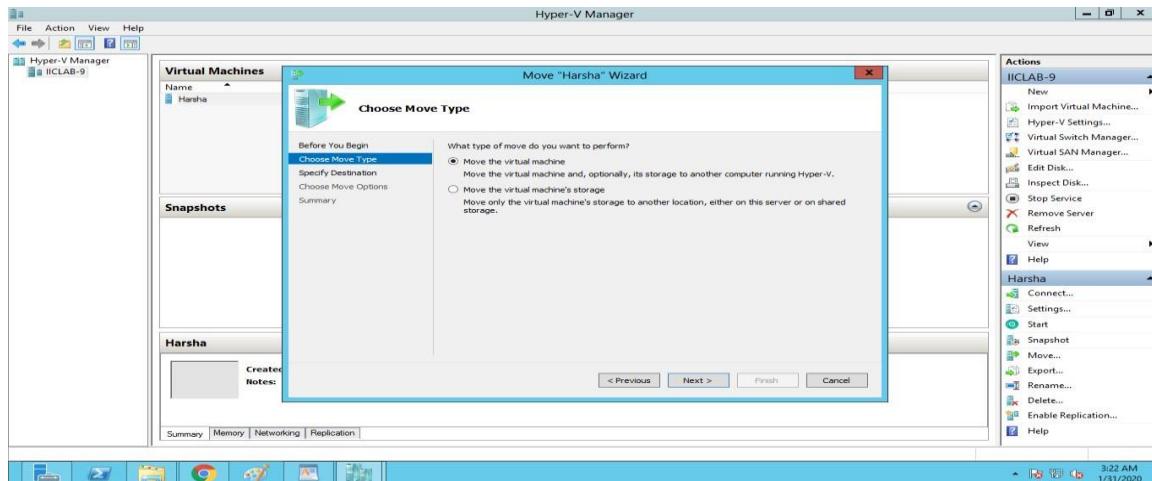
- Connect virtual hard disk size as 10GB and set the document path



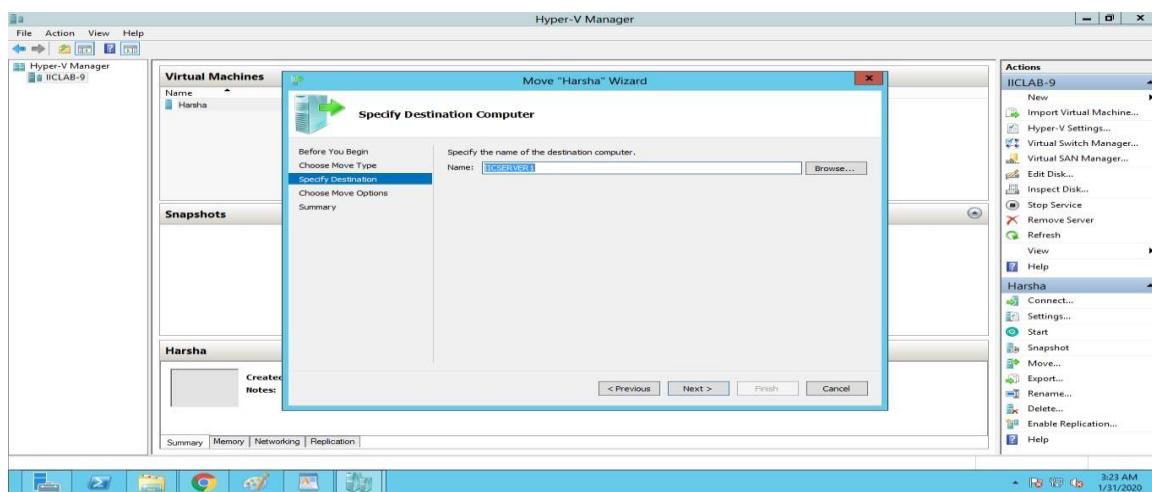
- Select ISO file to install the Ubuntu as Guest OS



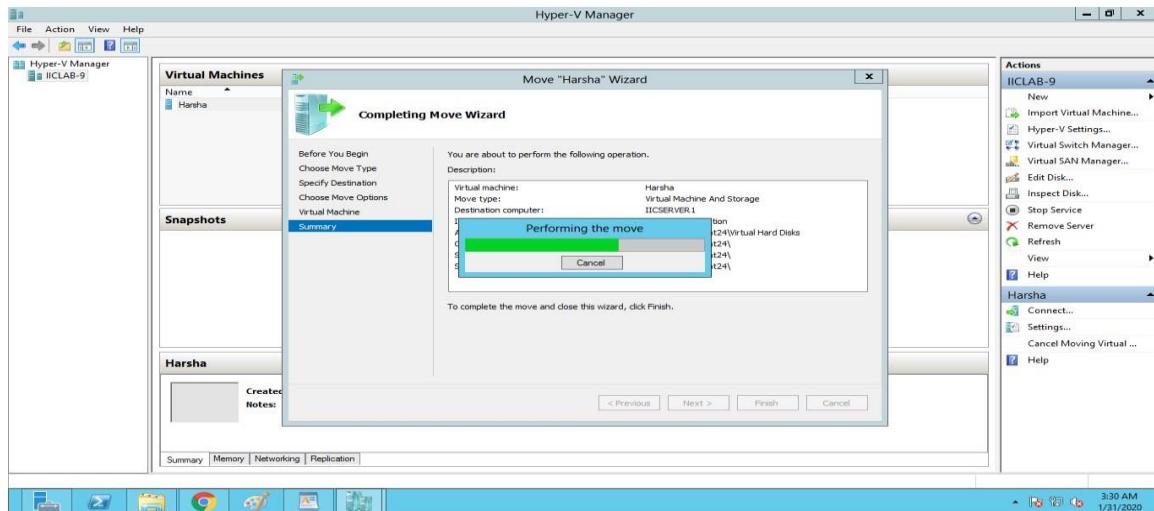
- Open the virtual machine and then Shutdown the virtual machine.



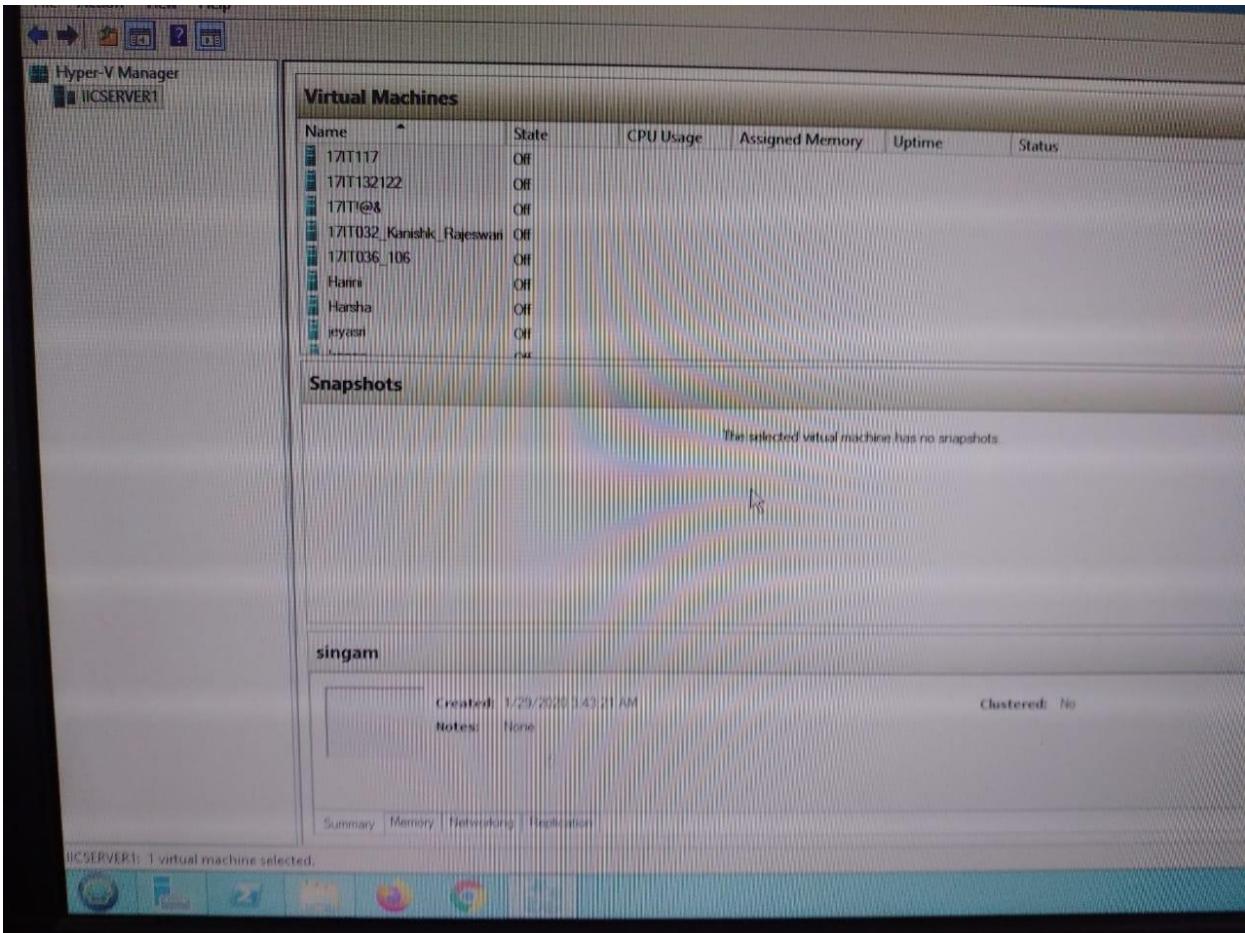
- From the settings, select move virtual machine



- Specifying the destination server



- Moving the Virtual Machine to the server
- Migrated Virtual Machines on the Destination computer..



Description:

Paravirtualization is a virtualization technique that provides an interface to virtual machines that are similar to their underlying hardware. In paravirtualization, the guest operating system is explicitly ported before installing a virtual machine because a non-tailored guest operating system cannot run on top of a virtual machine monitor (VMM). Xen is a [hypervisor](#) using a [microkernel](#) design, providing services that allow multiple computer [operating systems](#) to execute on the same [computer hardware](#) concurrently. The Xen Project is the leading open source virtualization platform that is powering some of the largest clouds in production today. Amazon Web

Services, Aliyun, Rackspace Public Cloud, Verizon Cloud and many hosting services use Xen Project software.

System Requirements:

XenServer requires at least two separate physical x86 computers: one to be the XenServer host and the other to run the XenCenter application or the XenServer Command-Line Interface (CLI). The XenServer host computer is dedicated entirely to the task of running XenServer and hosting VMs," and is not used for other applications.

To run XenCenter use any general-purpose Windows system that satisfies the hardware requirements. This Windows system can be used to run other applications. When you install XenCenter on this system, the XenServer CLI is also installed.

A standalone remote XenServer CLI can be installed on any general purpose Linux system.

i) Xen Server

Host: 64-bit x86 server

CPUs: One or more 64-bit x86 CPUs, 1.5 GHz minimum, 2 GHz or faster multicore CPU recommended.

RAM:2 GB minimum, 4 GB or more recommended

Disk space: 46 GB of disk space minimum, 70 GB of disk space recommended

ii) XenCenter

Operating System :

- Windows 10

- Windows 8.1
- Windows 7 SP1
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2 SP1
- Windows Server 2008 SP2 (see Note)
- Windows Server 2016

RAM: 1 GB Minimum, 2 GB Or More Recommended

Disk Space: 100 MB Minimum Network: 100 Mbit/S Or Faster NIC

Hardware requirements:

- The CPU vendor (Intel, AMD) must be the same on all CPUs on all servers.
- To run HVM virtual machines, all CPUs must have virtualization enabled.

a) Procedure to install Citrix Xenserver 6.5

1. The first step in the installation is to download the **XenServer ISO** file. This can easily be accomplished by visiting the link above or using the ‘**wget**’ utility on a Linux system.

```
# wget -c http://downloadns.citrix.com.edgesuite.net/10175/XenServer-6.5.0-xenserver.orginstall-cd.iso
```

Now burn the ISO to a CD or using ‘**dd**’ to copy the ISO to a flash drive.

```
# dd if=XenServer-6.5.0-xenserver.org-install-cd.iso of=</path/to/usb/drive>
```

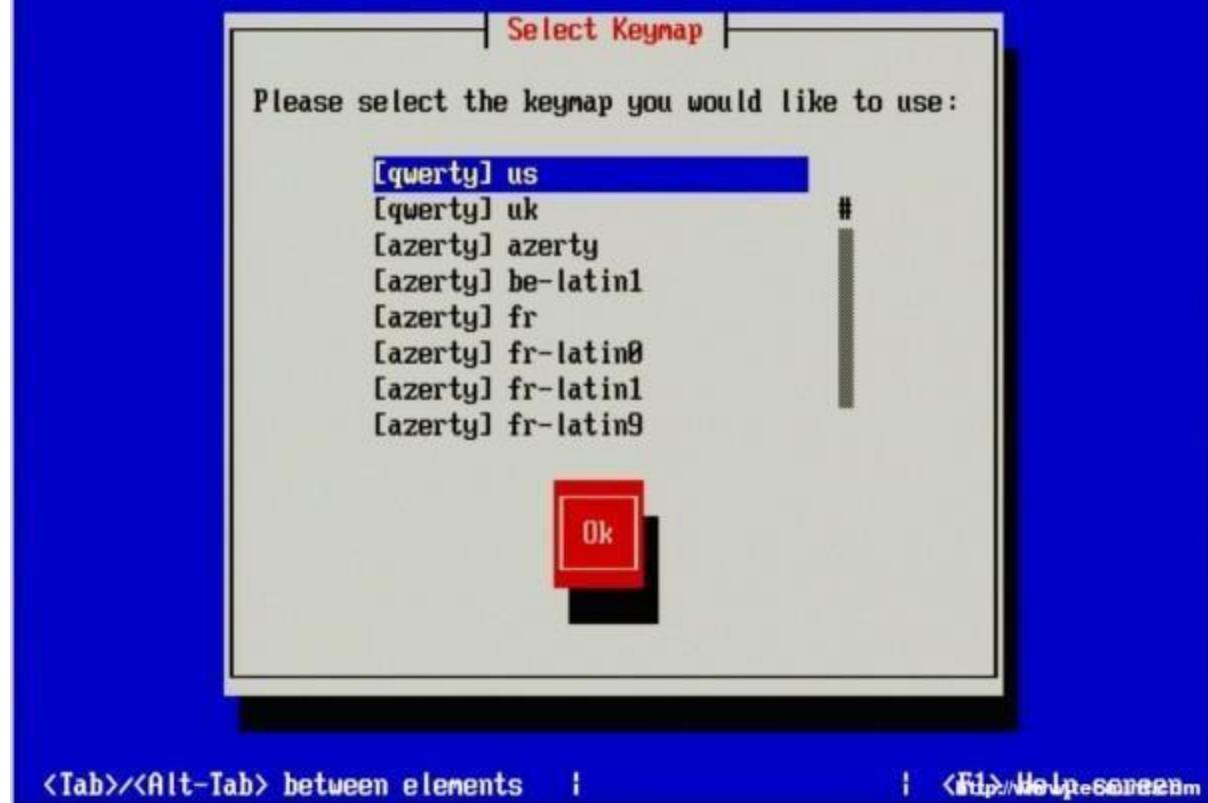
2. Now place the media into the system that **XenServer** will be installed and boot to that media. Upon successful boot the user should be greeted by the wonderful **Citrix XenServer** boot splash.



XenServer Boot Menu

3. At this point simply press **enter** to begin the booting process. This will boot the user into the XenServer installer. The first screen will ask the user to provide a **language** selection.

Welcome to XenServer - Version 6.5.0 (#90233c)
Copyright (c) 2014 Citrix Systems, Inc.



Select XenServer Installation Language

4. The next screen asks the user to confirm the reason for booting to this media as well as provide the option to load extra hardware drivers if needed. In this particular case, it is to install XenServer to the machine so it is safe to click "**OK**".

Welcome to XenServer Setup

This setup tool can be used to install or upgrade XenServer on your system or restore your server from backup. Installing XenServer will erase all data on the disks selected for use.

Please make sure you have backed up any data you wish to preserve before proceeding.

To load a device driver press <F9>.

OK

Reboot

<Tab>/<Alt-Tab> between elements | <F9> load driver | <F1> Help screen

Load XenServer Device Driver

5. The next prompt is the obligatory **EULA** (End User License Agreement). Feel free to read the whole thing, as you're supposed to anyways right, otherwise using the keyboard arrows move the cursor over to the "**Accept EULA**" button and hit enter.

Please view the accompanying source code (including LICENSE, COPYING, and README files, if any) for the applicable license agreements. To the maximum extent permitted by applicable law, the accompanying software is provided "as is" WITHOUT WARRANTIES OF ANY KIND, including the implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

Accept EULA

Back

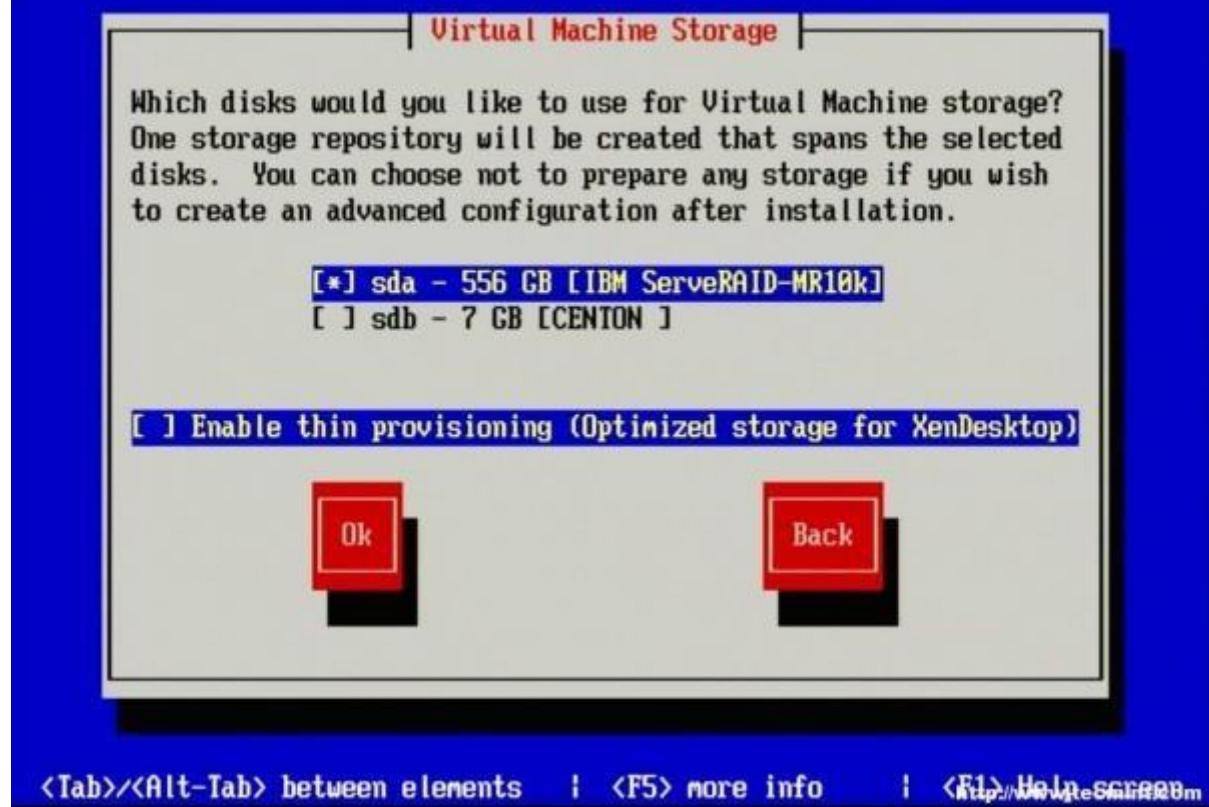
<Tab>/<Alt-Tab> between elements |

| <F1>/Help <esc>|

Accept License Agreement

6. The next screen requests the installation device. In this example the RAID setup on the server is where XenServer will be installed.

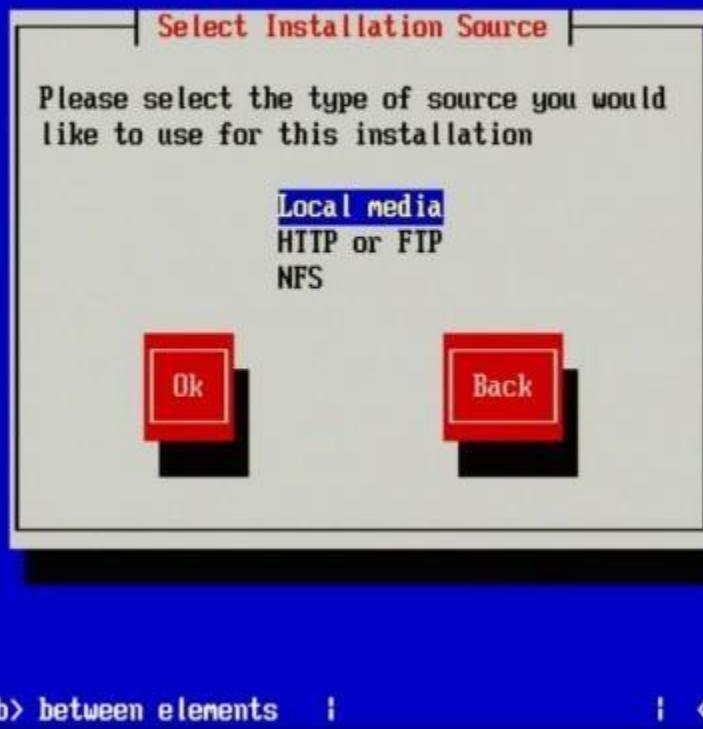
The RAID system is reflected as "**sda - 556 GB [IBM ServeRAID-MR10k]**" For this guide, thin provisioning is not necessary. Make sure the asterisk (*) character is next to the hard drive selection to install XenServer and tab to the "OK" button.



Select XenServer Virtual Machine Storage

7. The next screen will prompt the user for the location of the installation files. Since the installer was boot locally with a CD/DVD/USB, make sure to select the **“Local Media”** option.

Welcome to XenServer - Version 6.5.0 (#90233c)
Copyright (c) 2014 Citrix Systems, Inc.



Select XenServer Installation Source

8. The next step allows for the installation of **Supplemental Packs** (SP) at the time of install. For this guide, none of the supplemental packs available will be installed at this point but will be covered later once XenServer is up and running.



Select Supplemental Packs

9. The next screen will ask if the user wishes to verify that the installer media is not corrupt. Generally this is a good idea but is a personal choice. All in all the verification on this test server took about 3 minutes from a CD.

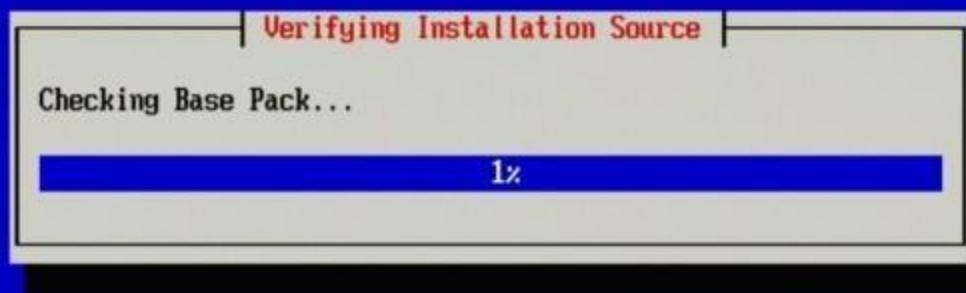


<Tab>/<Alt-Tab> between elements |

| <F1> Help screen

Verify XenServer Installation Media

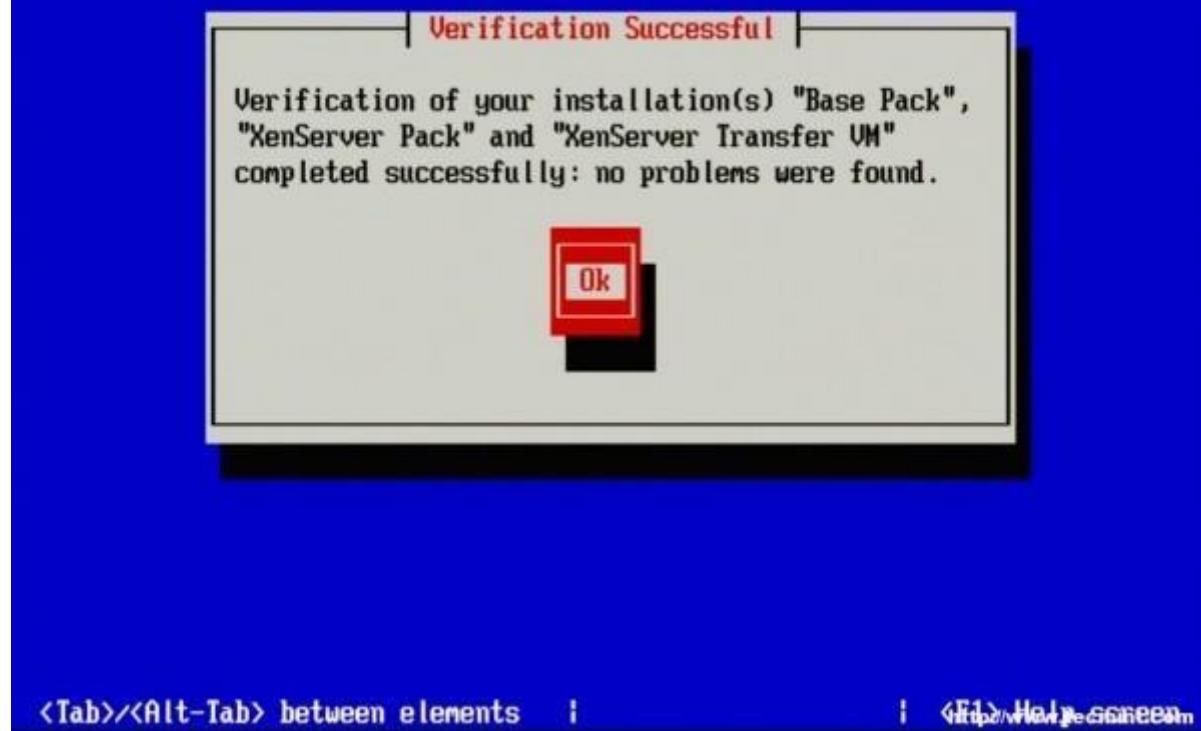
Welcome to XenServer - Version 6.5.0 (#90233c)
Copyright (c) 2014 Citrix Systems, Inc.



Working: Please wait...

<http://www.tecmint.com>

Checking Base Pack



Verification Successful

10. Once the verification is completed, if selected during install, the XenServer installer will ask the user to setup some system information.

The first prompt will be to set the root user's password. Now, since XenServer will be the underlying system to potentially several important virtualized servers, it is imperative that the password be secured as well as sufficiently complex!

Important: Do not forget this password either as there will not be any other users on the system once the installer finishes!



Set XenServer Root Password

11. The next screen will ask the user to chose a management interface. The number of interfaces that shows up will vary from system to system and the management interface doesn't have to be a separate interface. A separate interface can be used to secure administrative access to the hypervisor itself.



Configure XenServer Network

12. Once the management interface is selected, the system will prompt for how to obtain an IP address for the management interface. This step will obviously vary from site to site as well.



Set XenServer IP Address

13. After determining the IP address information for the XenServer, the system will ask for a **hostname** to be set as well as **DNS** server configuration.



Hostname and DNS Configuration

14. The next three screens will go through the steps of setting up time zones as well as the method for keeping track of time. Generally, **Network Time Protocol** is suggested for keeping system time synchronized on systems but again this will vary from location to location.

If a network time server isn't available or the XenServer doesn't have access to the Internet time servers, the installer will allow for manual time setting.

Please select the geographical area
that your XenServer Host is in:

Africa

America

Antarctica

Arctic

Asia

Atlantic

Australia

Brazil

Ok

Back

<Tab>/<Alt-Tab> between elements

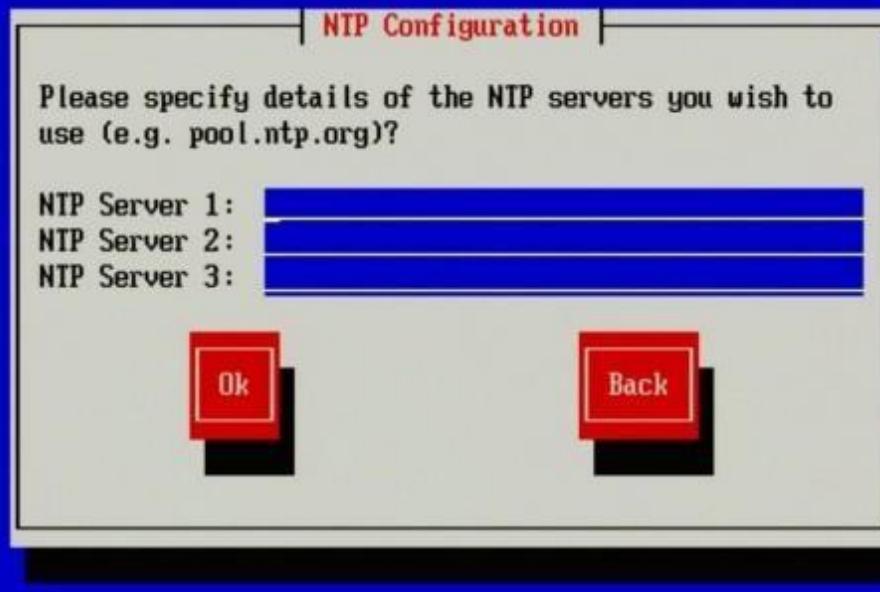
| <F1> Help screen

Set Time Zone

Welcome to XenServer – Version 6.5.0 (#90233c)
Copyright (c) 2014 Citrix Systems, Inc.



Set Time Using NTP



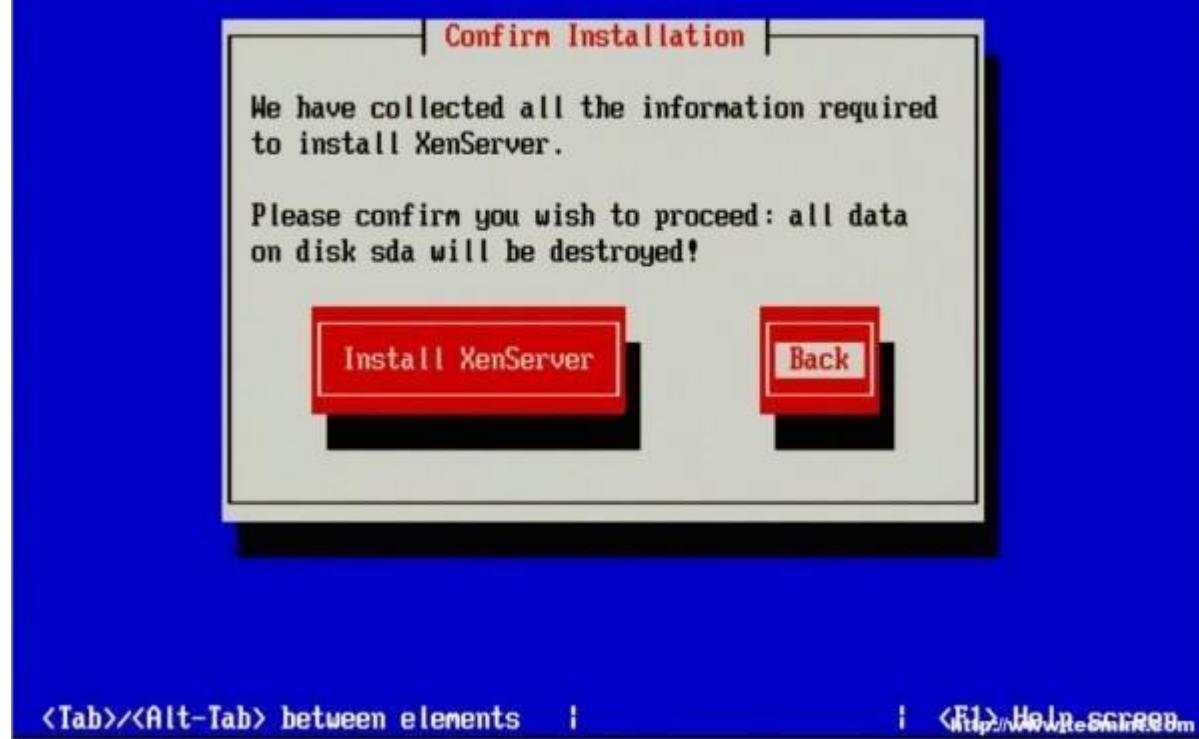
<Tab>/<Alt-Tab> between elements

| <F1> Help

Add NTP Servers

15. At this point all of the initial configuration parameters will be complete and the XenServer installer will be ready to begin the installation of the hypervisor.

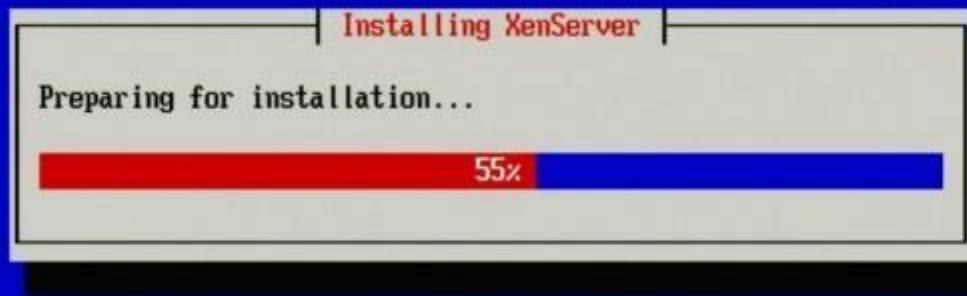
WARNING – Continuing at this point **WILL ERASE ALL DATA** on the target disks!



Confirm XenServer Installation Drive

16. The installation will take some time but a progress bar will be displayed. Once the installer is done, it will prompt the user to restart the system to boot to the newly installed hypervisor (be sure to remove the XenServer installation disk upon reboot).

Welcome to XenServer - Version 6.5.0 (#90233c)
Copyright (c) 2014 Citrix Systems, Inc.



Working: Please wait...

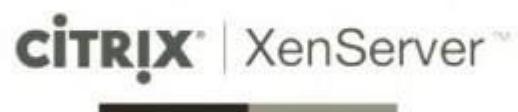
<http://www.tecmint.com>

Preparing XenServer Installation



XenServer Installation Complete

17. Upon removing the installation media and rebooting the system, the user should be presented with the **Citrix XenServer** splash page. Allow the system to continue booting.



Citrix XenServer Booting

18. Once the booting has completed, the system will present the **XenServer control** page.



XenServer Configuration Page

b) Procedure to install XenCenter

1. Before installing XenCenter, be sure to uninstall any previous version.
2. Launch the installer.

If installing from the XenServer installation CD:

Welcome to Citrix XenServer ® 7.1.0

Thank you for using Citrix XenServer. This page contains important information about the XenServer 7.1.0 release.

Read Me First

For information about getting started with XenServer and for links to product documentation and other useful resources, see [Read Me First](#).

XenCenter

The links below can be used to download and install XenCenter, the graphical, Windows-based user interface for XenServer, from this host.

- [XenCenter CD image](#)
- [XenCenter installer](#)

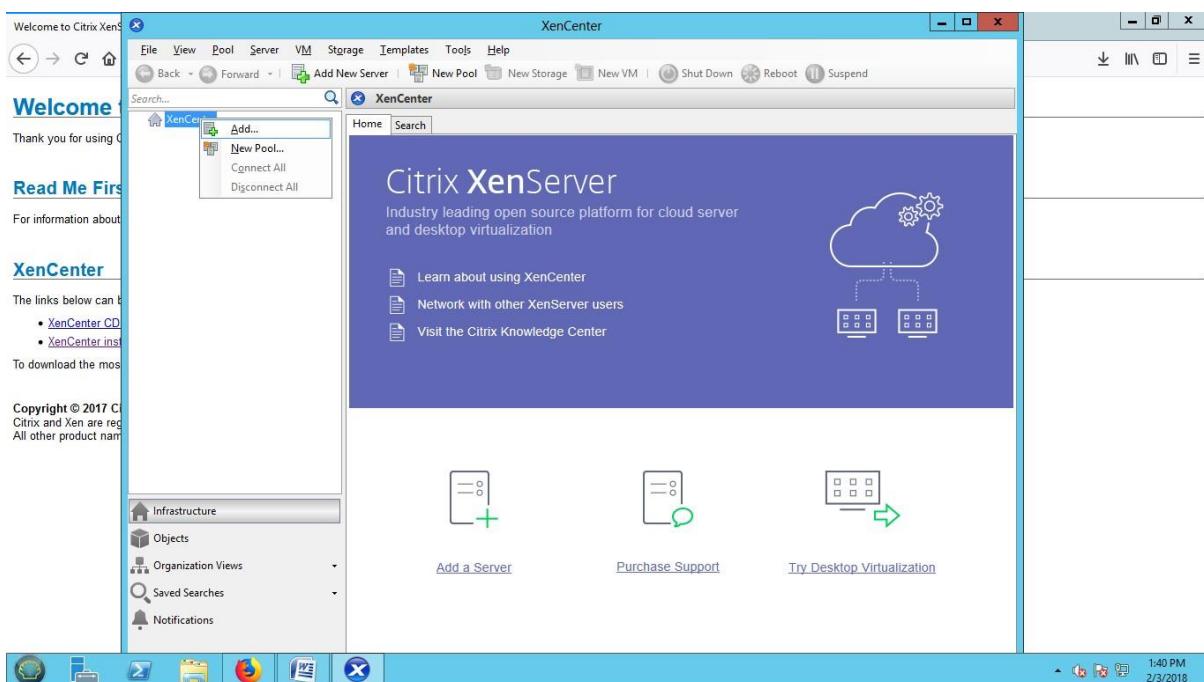
To download the most up-to-date version go to [How to Download and Install a New Version of XenCenter](#).

Copyright © 2017 Citrix Systems, Inc. All rights reserved.
Citrix and Xen are registered trademarks. XenServer and XenCenter are trademarks of Citrix Systems, Inc. in the United States and other countries.
All other product names, company names, marks, logos, and symbols are trademarks of their respective owners.

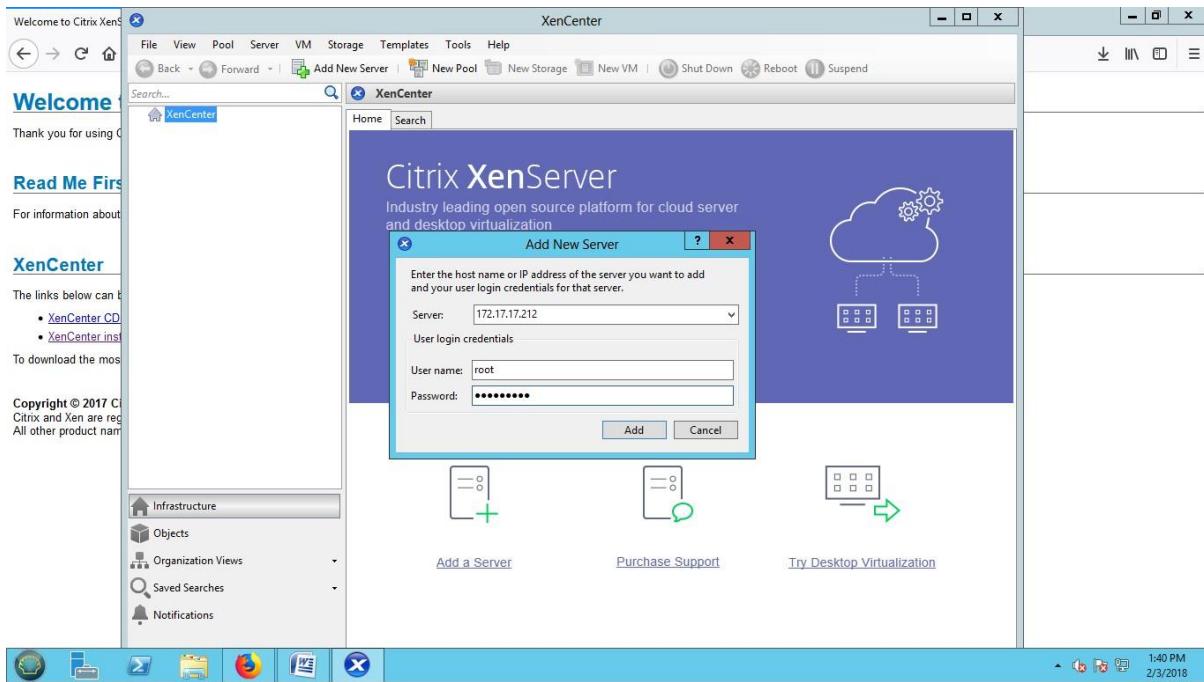
-
1. Insert the CD into the DVD drive of the computer which you want to run XenCenter.
 2. Open the client_install folder on the CD. Double-click XenCenter.msi to begin the installation.
 3. Follow the Setup wizard, which allows you to modify the default destination folder and then to install XenCenter.
- c) **Procedure to manage and migrate VMs in XenServer**
1. Launch XenCenter. The program opens to the **Home** tab..



2. Click the Add New Server icon.

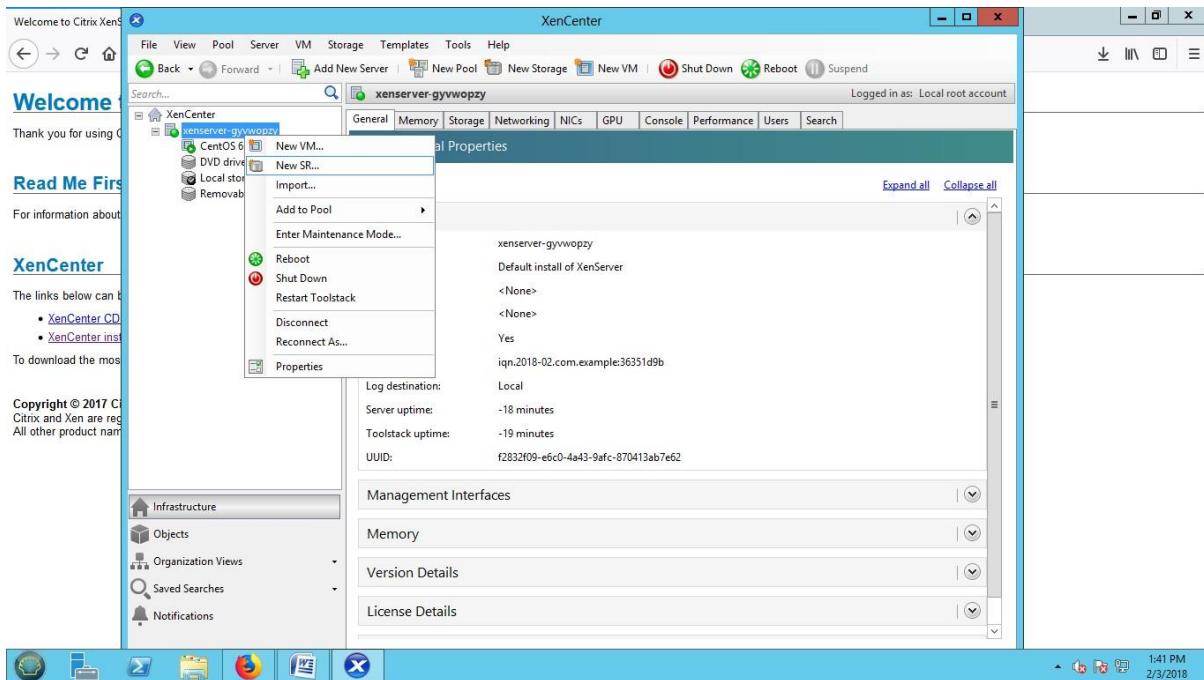


3: Enter the IP address of the XenServer host in the **Server** field. Type the root user name and password that you set during XenServer installation. Click **Add**.

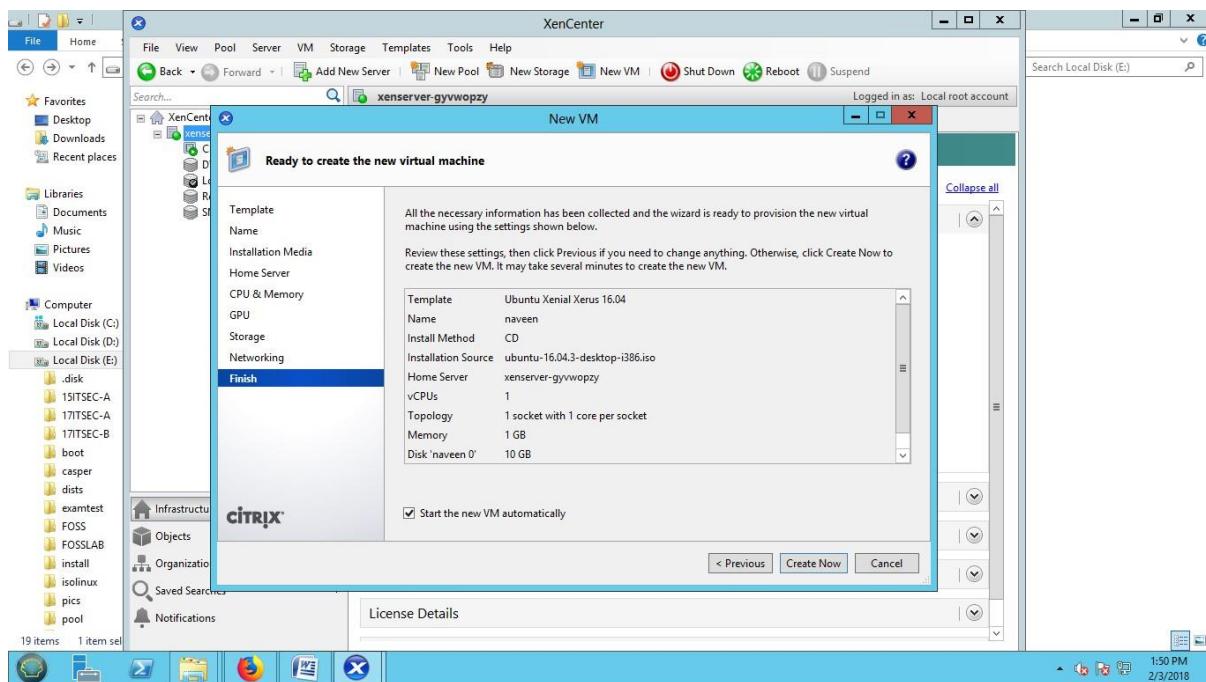
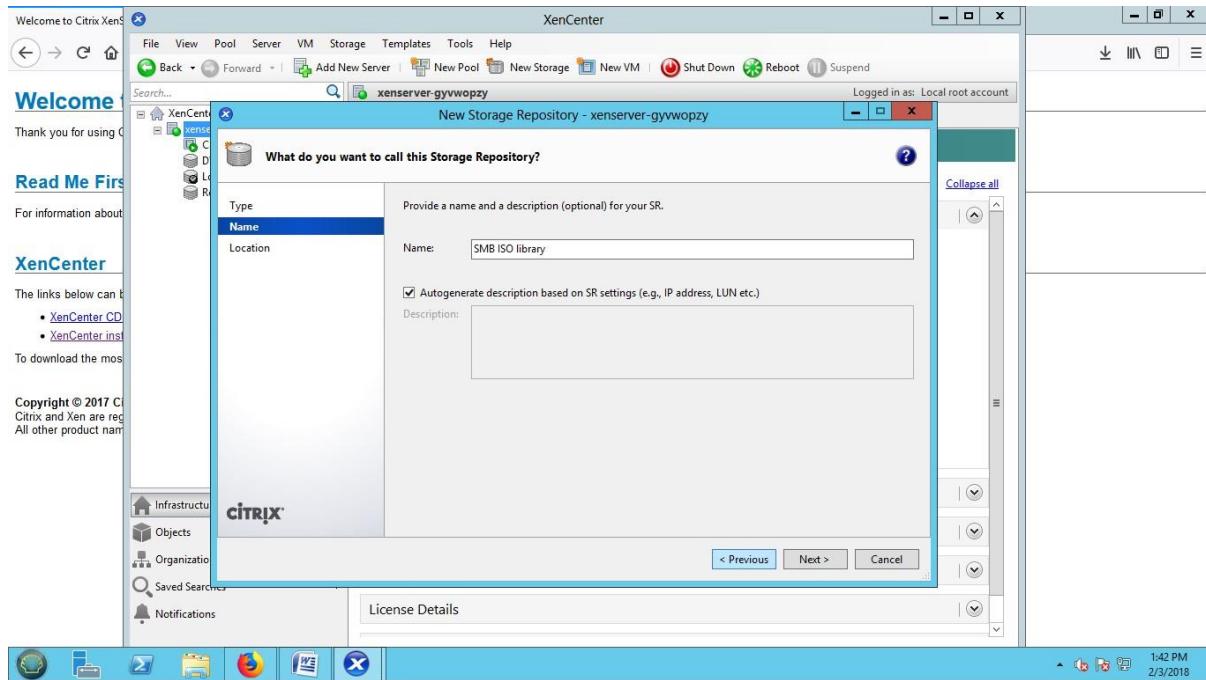


4: Add a host.

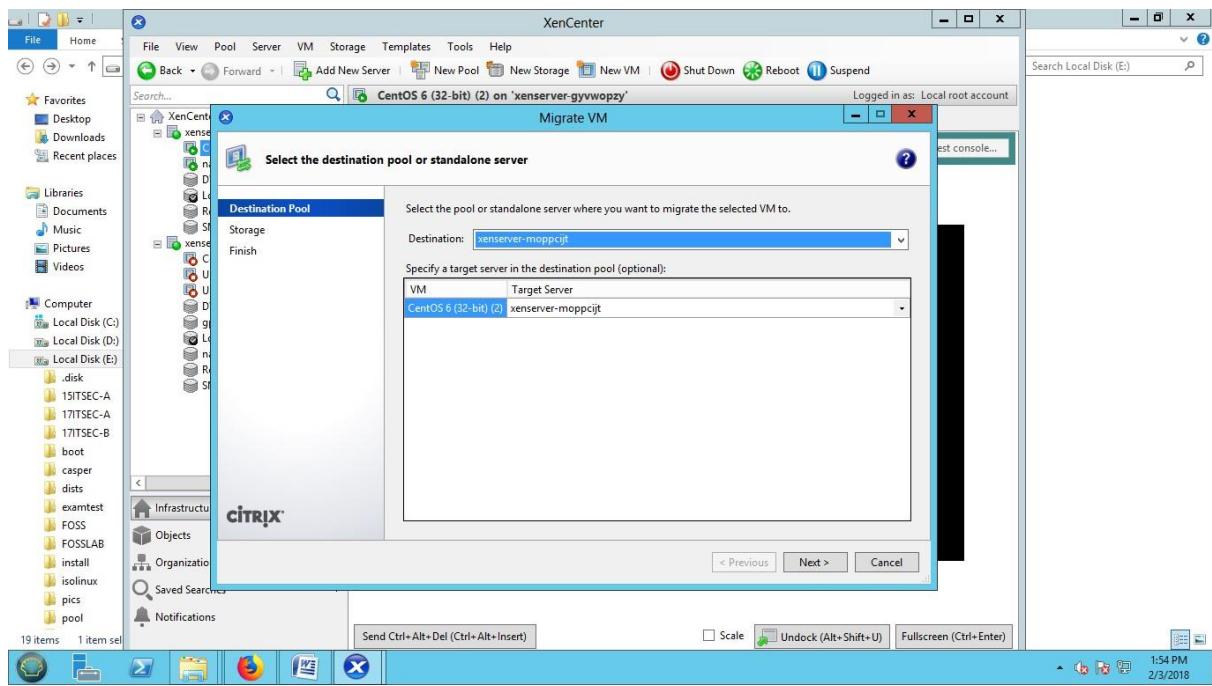
The first time you add a host, the **Save and Restore Connection State** dialog box appears. This dialog enables you to set your preferences for storing your host connection information and automatically restoring host connections.

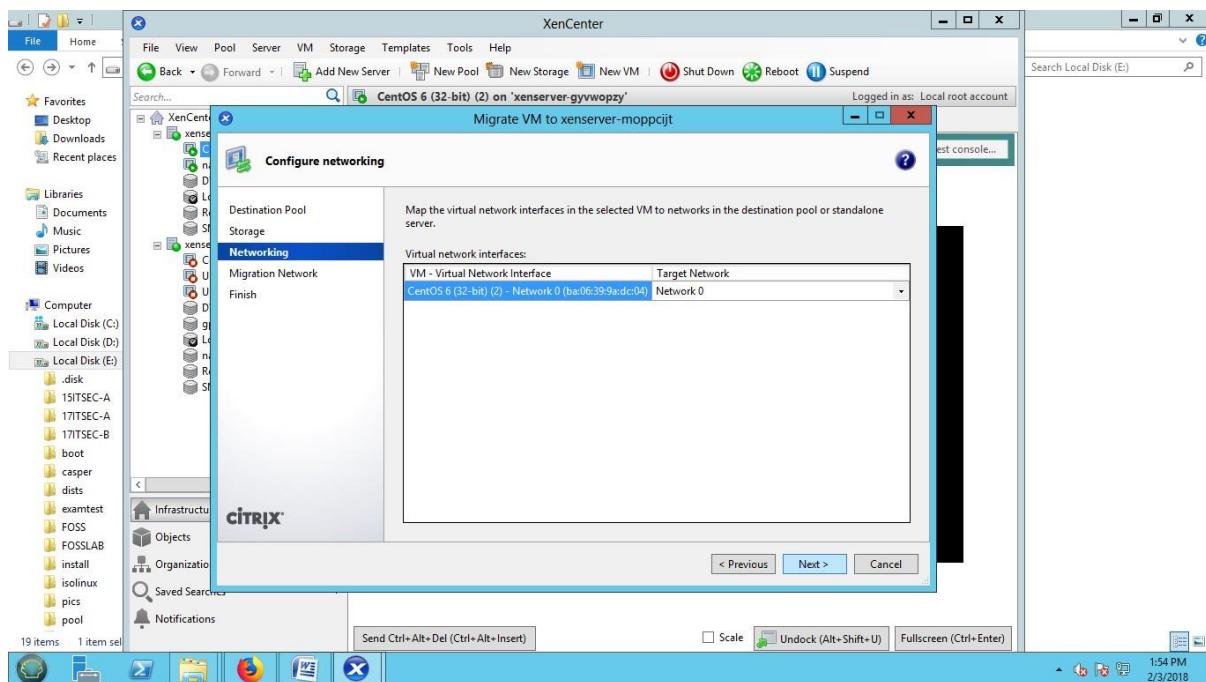
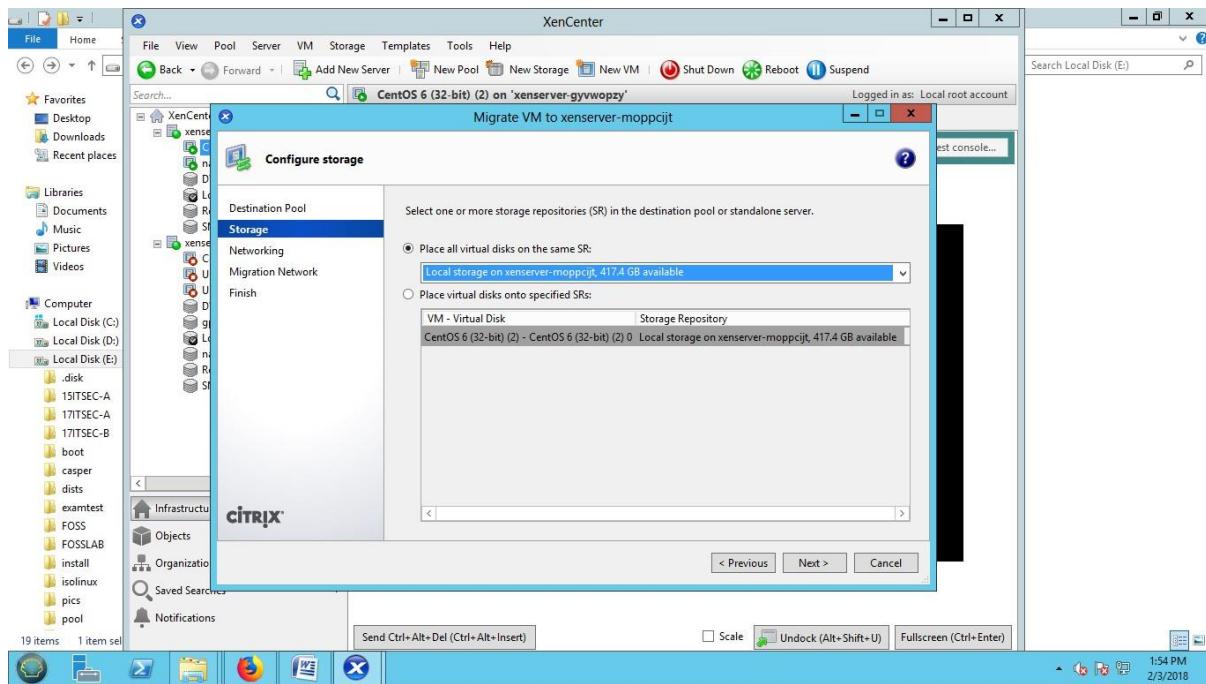


5: Specify the OS and click on next.

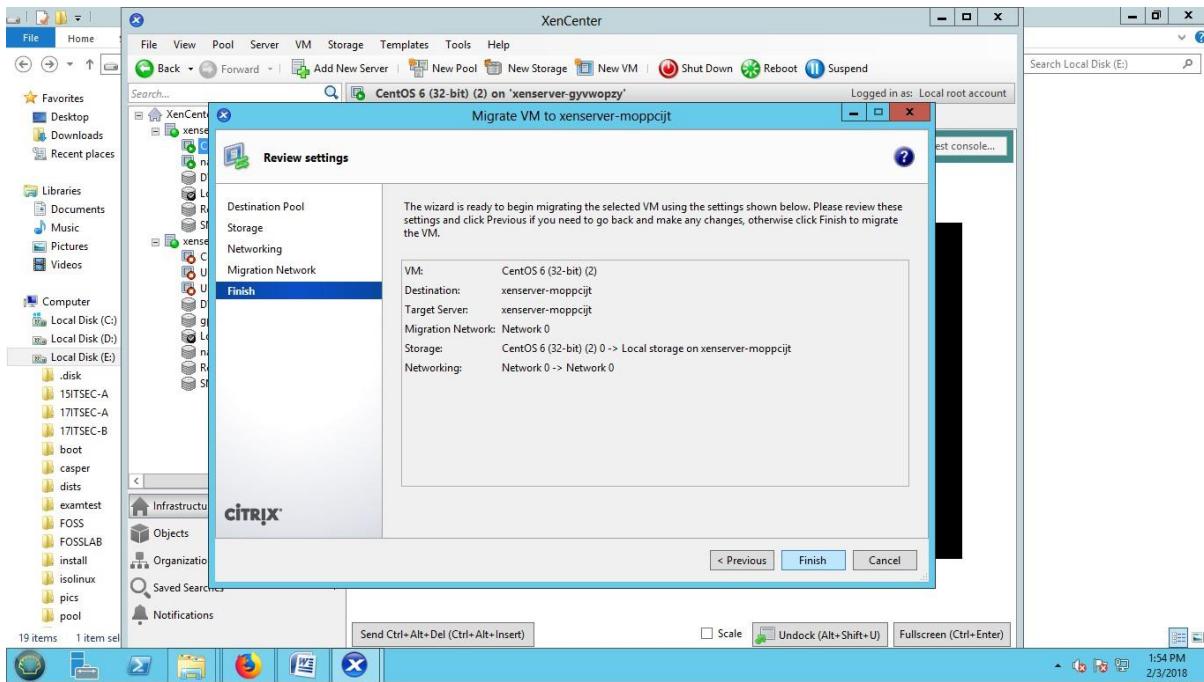


6:Create another server and move the server





7:Vm moved to another server.



Result:

Thus, the VM Migration using Microsoft Hyper-V tool and Xen is implemented Successfully.

Ex.No : 9

03.05.2021

Study on AR-VR as a service from Cloud

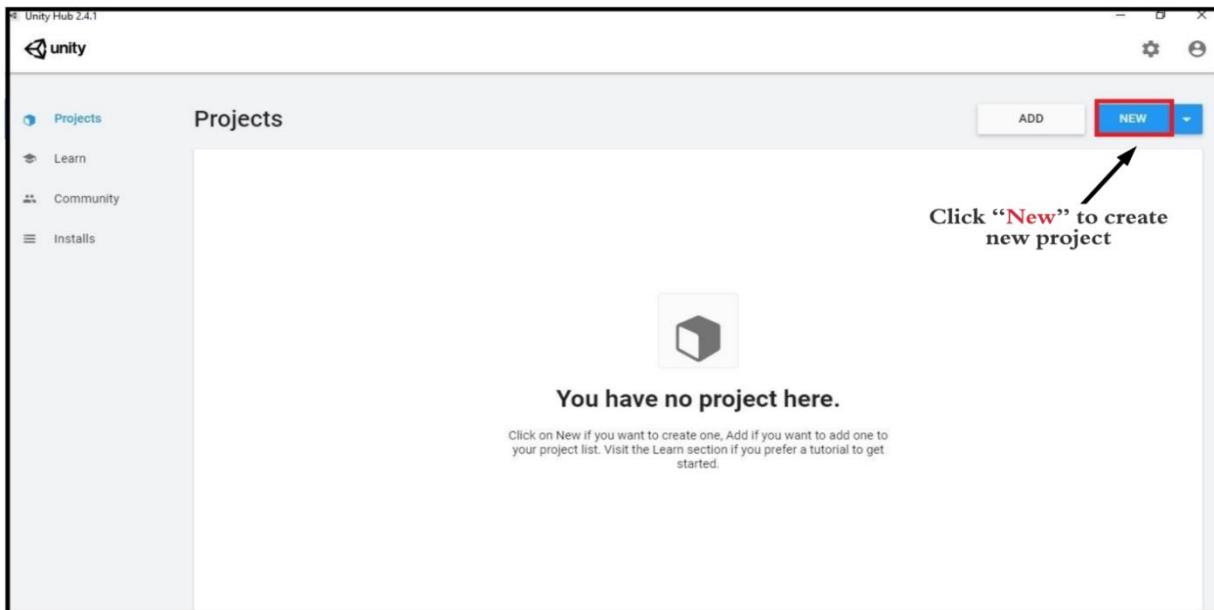
AIM:

To study AR VR as a service from cloud

PROCEDURE:

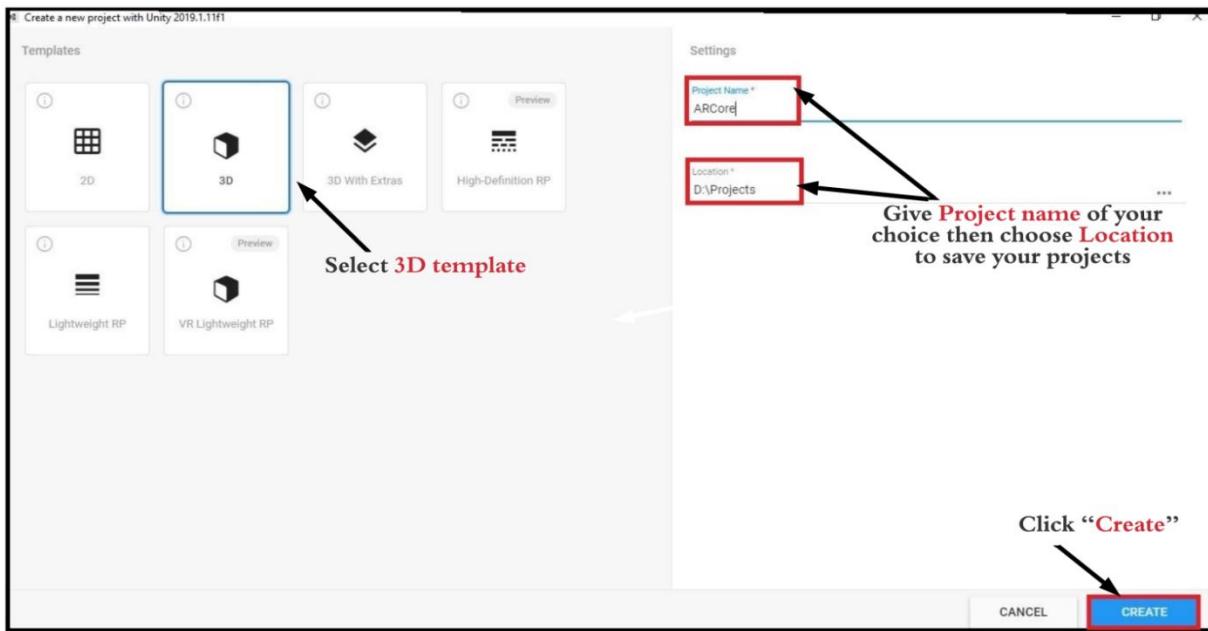
STEP 1:

By clicking “New” option, you can create new project in unity.



STEP 2:

Select 3D option and you can give any name of your choice. For example, as given below “ARCore” and click create button.



STEP 3:

Go through this link

<https://developers.google.com/ar/develop/unity/quickstart-android>

The screenshot shows the ARCore Unity SDK Quickstart page. A red box highlights the download link "ARCore SDK for Unity 1.20.0 or later". A callout bubble points to it with the text "Choose ARCore unity sdk package of your choice". Another callout bubble points to the main content area with the text "To develop ARCore app, you need to go through this manual step".

Get the ARCore SDK for Unity

1. Download [ARCore SDK for Unity 1.20.0 or later](#).
The SDK is downloaded as `arcore-unity-sdk-1.20.0.unitypackage`.

Note: Google collects and processes data from your use of the ARCore SDK Policy and the [Google APIs Terms of Service](#). This data may include, for example, the version of the ARCore SDK.

To develop ARCore app, you need to go through this manual step

Create a new project and import the SDK

1. Open Unity and create a new 3D project.
2. Unity 2019 only: Select Window > Package Manager and install the following packages:
 - Multiplayer HLAPI (required by the CloudAnchors sample)
 - XR Legacy Input Helpers (required by Instant Preview, which uses the TrackedPoseDriver)
3. Import the ARCore SDK for Unity:
 - a. Select Assets > Import Package > Custom Package.
 - b. Select the `arcore-unity-sdk-1.20.0.unitypackage` that you downloaded.
 - c. In the Importing Package dialog, make sure that all package options are selected and click Import.

STEP 4:

Download ARCore SDK Unity package.

The screenshot shows the GitHub release page for ARCore SDK v1.18.0. A red box highlights the download link "[arcore-unity-sdk-1.18.0.unitypackage](#)". A callout bubble points to it with the text "Click to Download arcore unity sdk Package".

Other changes

- Added `CloudServiceResponse.ErrorTooManyCloudAnchors` enum value.
- The HelloAR sample now includes support for occlusion using the ARCore Depth API.
- Updated C# style for non-public variables and methods. For example, `private float m_Foo` now reads `private float _foo` and `private void _Bar()` now reads `private void Bar()`.

Bug fixes

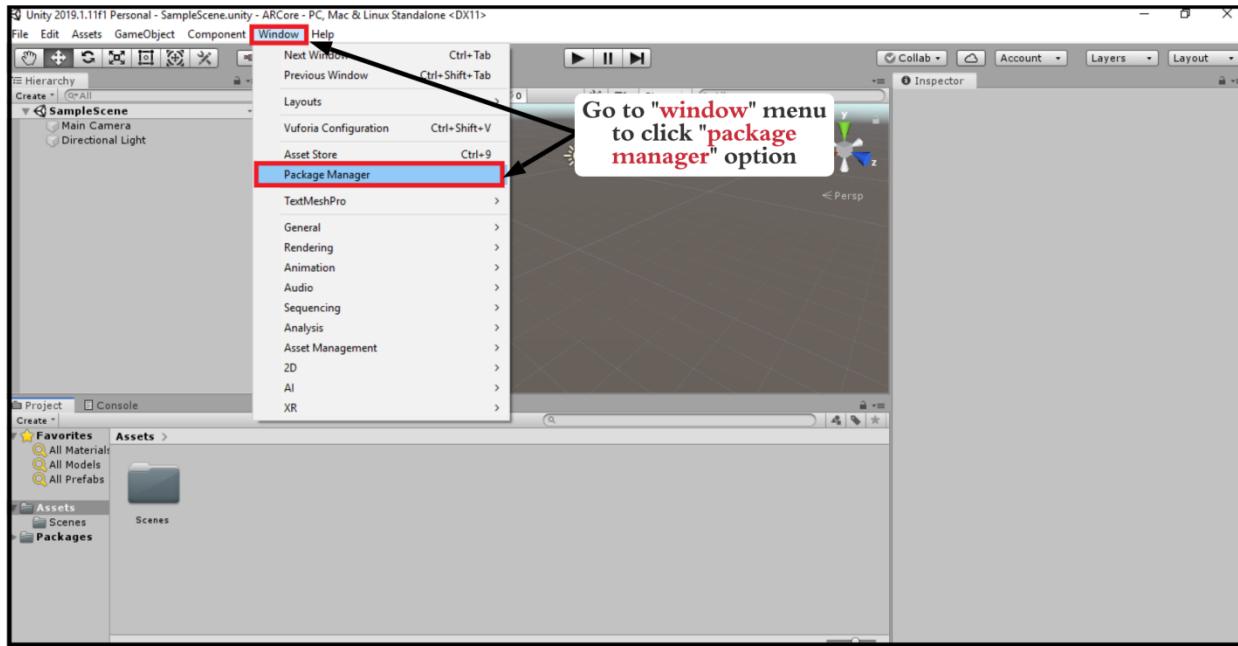
- Fixed Cloud Anchors Privacy link in the `CloudAnchors` sample app. The incorrect link now redirects to the correct link, so existing apps with the incorrect link don't need to be updated.

Assets 3

- [arcore-unity-sdk-1.18.0.unitypackage](#) (selected)
- [Source code \(zip\)](#)
- [Source code \(tar.gz\)](#)

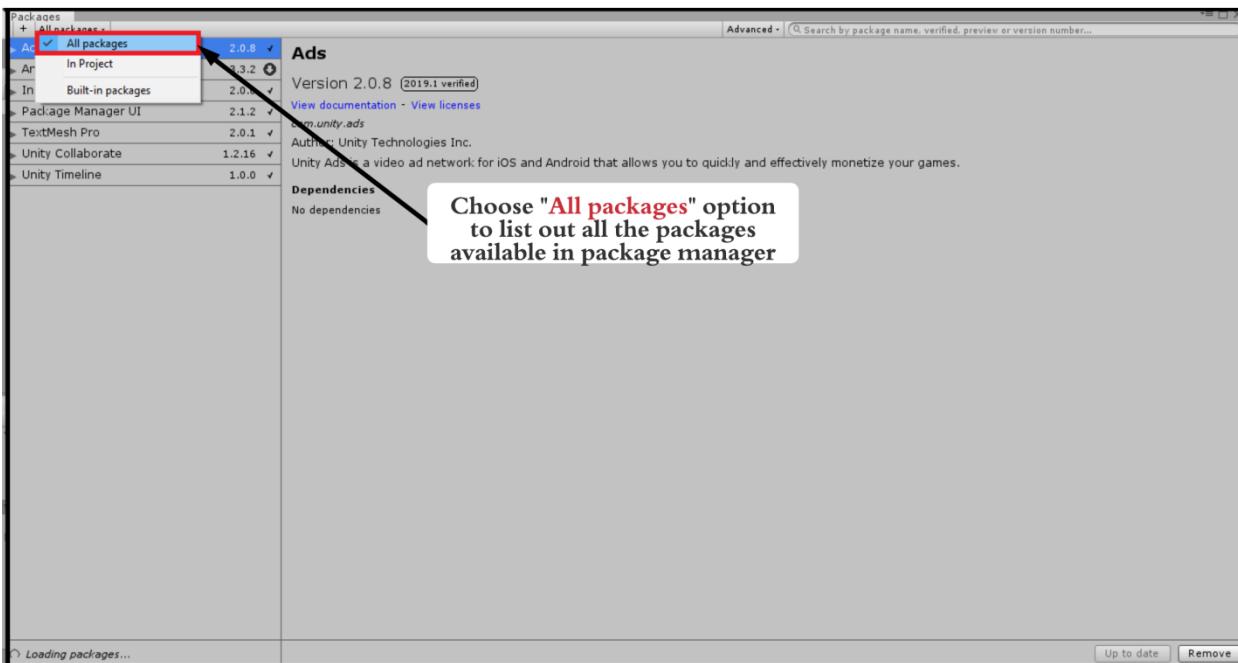
STEP 5:

ARCore SDK needs some supporting packages or it may cause error, so go to Windows → Package Manager.



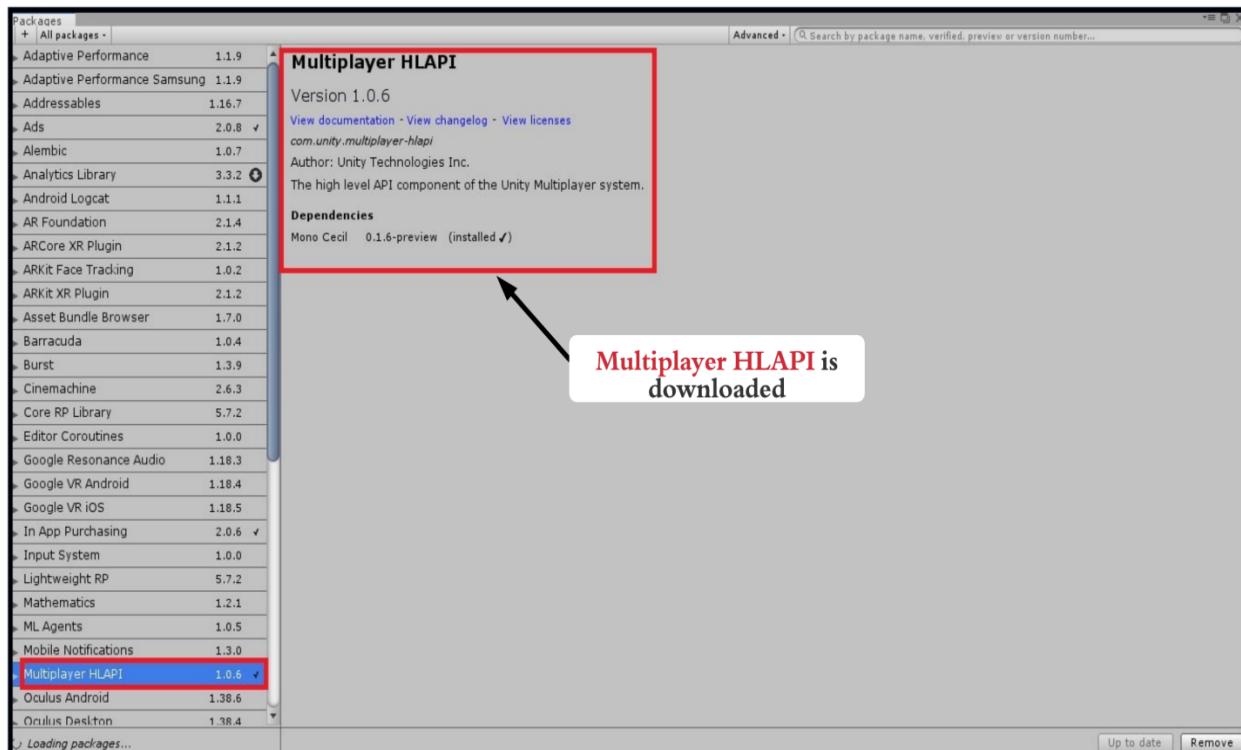
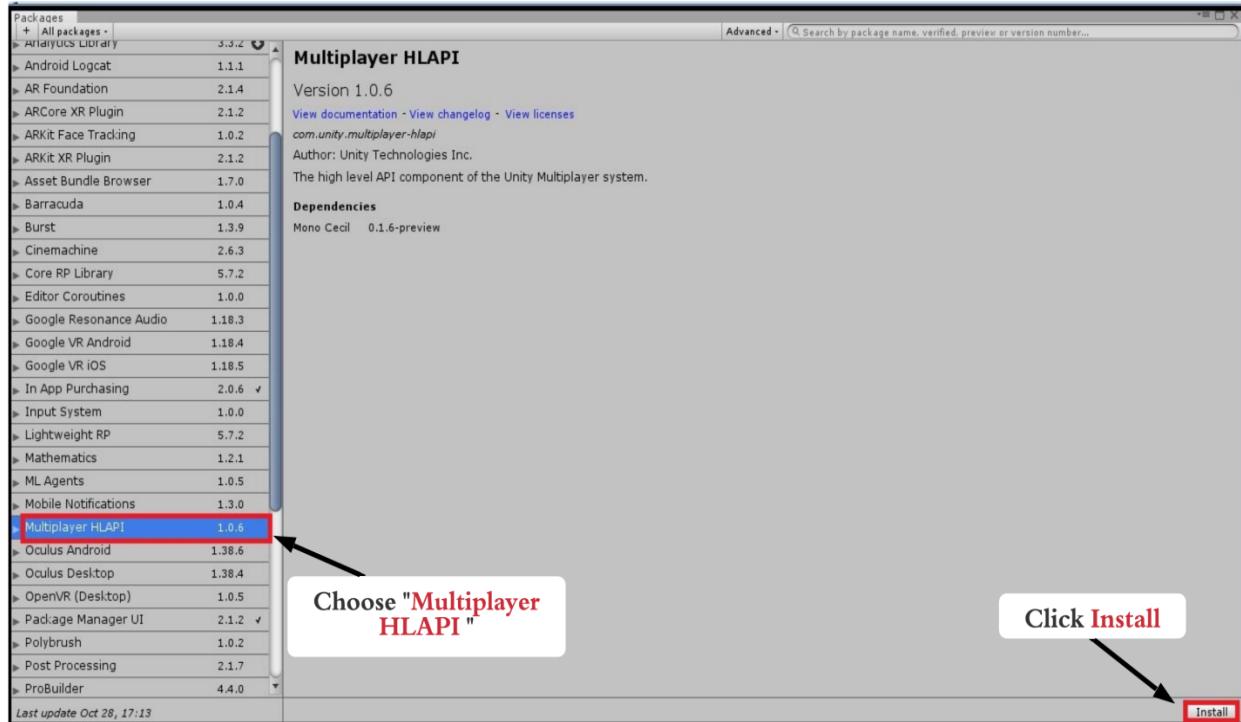
STEP 6:

There may be default packages in Package Manager (In Project), excluding this you can install packages.



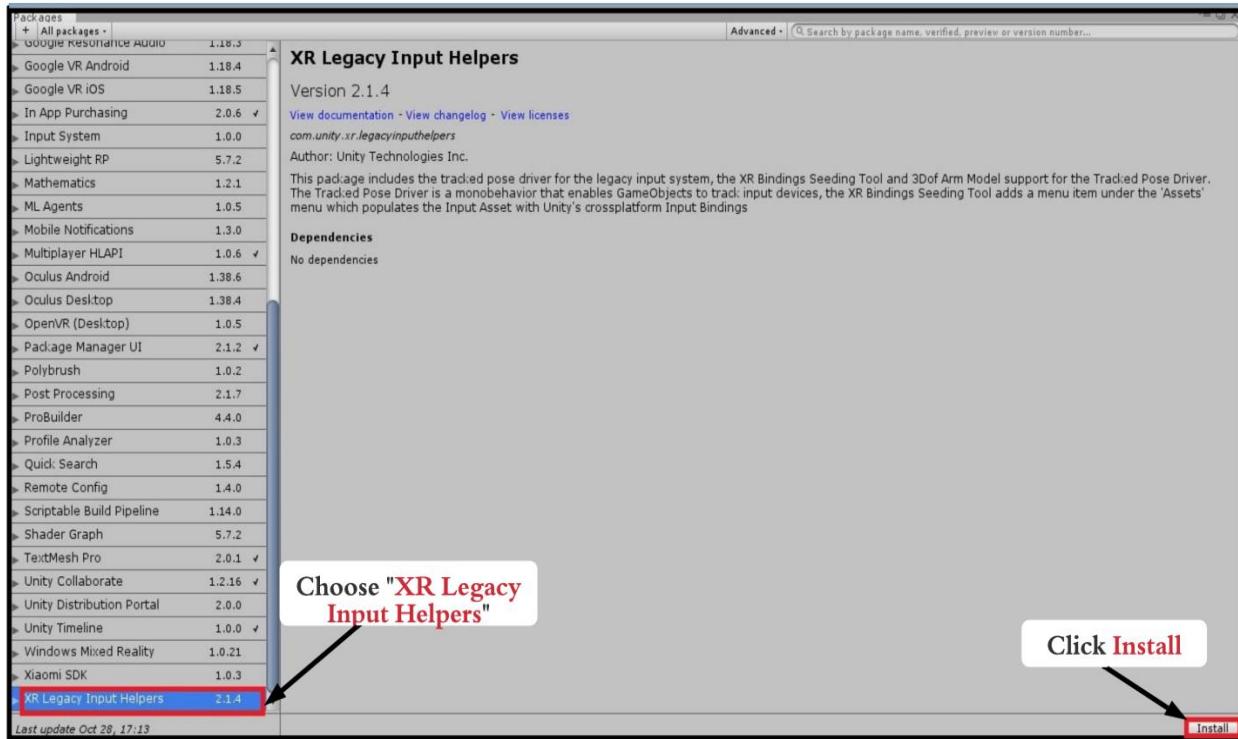
STEP 7:

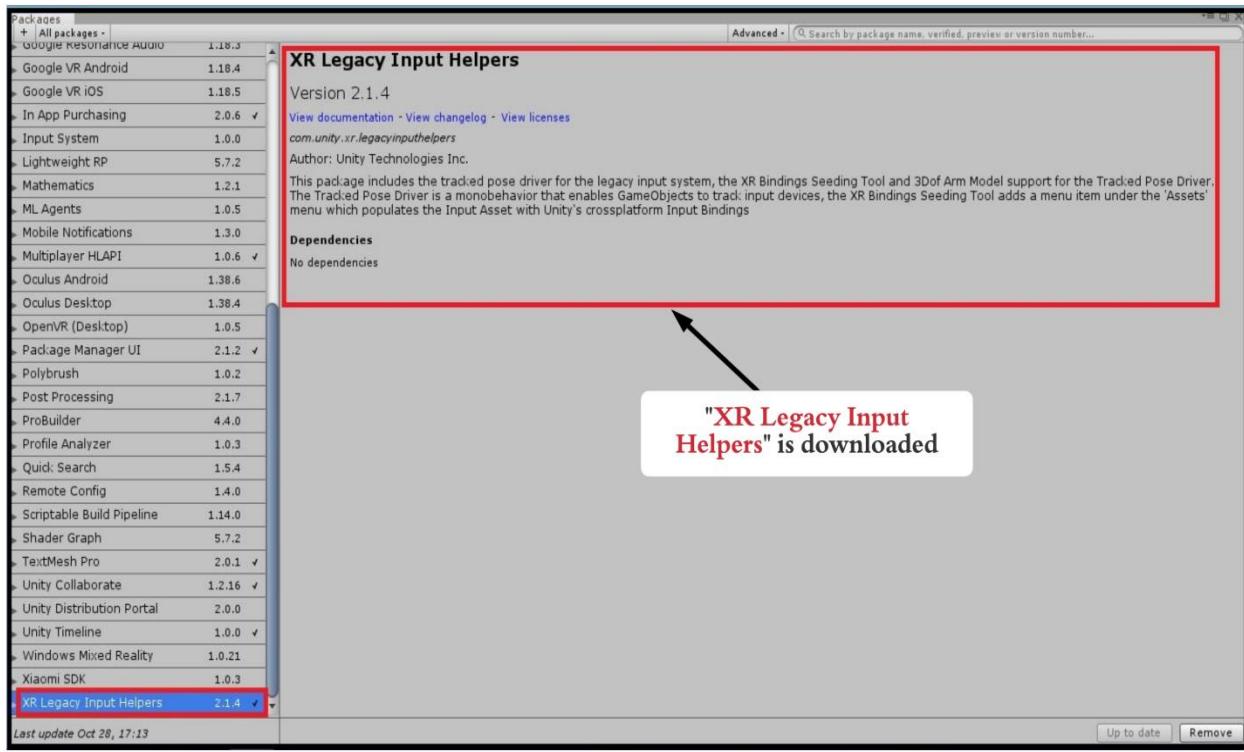
Multiplayer HLAPI needed to be downloaded.



STEP 8:

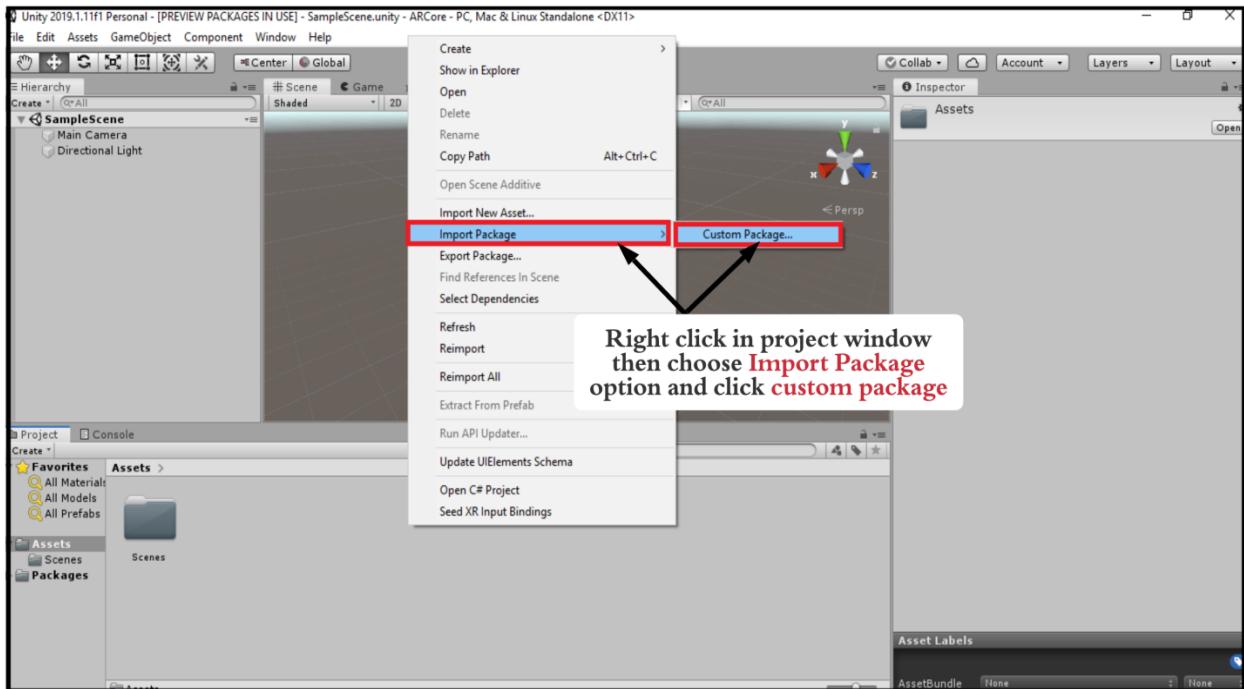
XR Legacy Input Helpers needed to be downloaded.





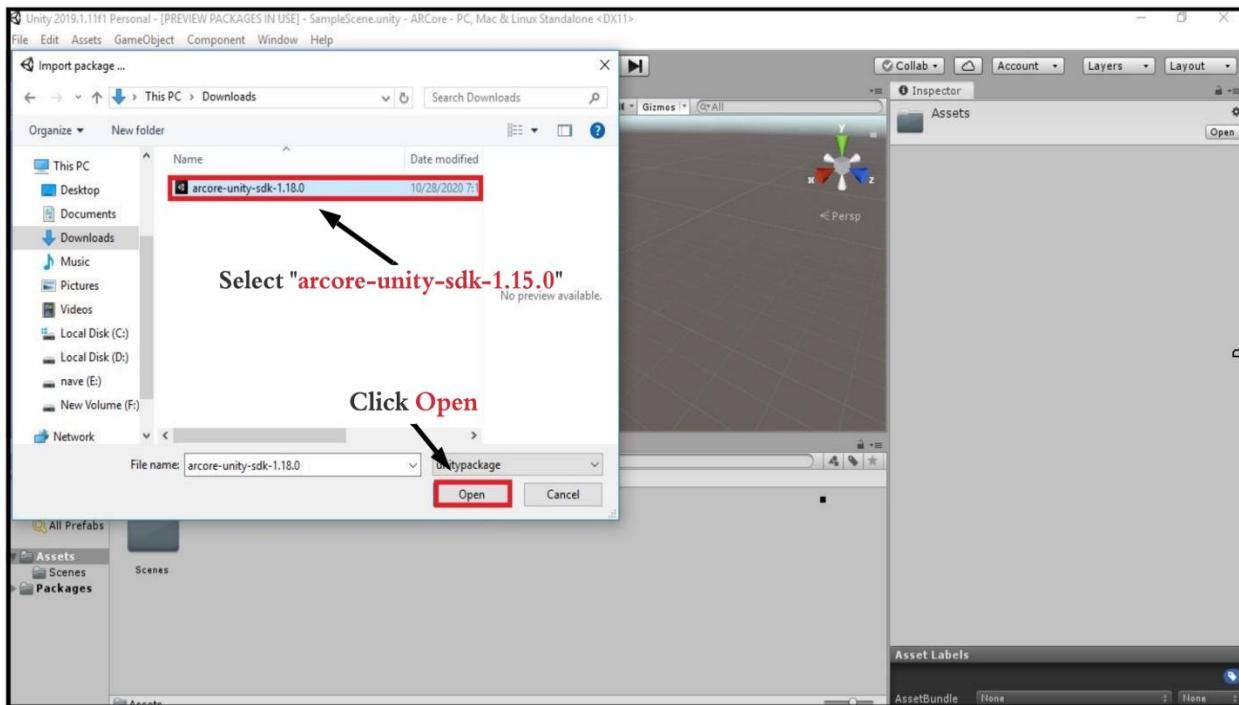
STEP 9:

In Project window right-click to import the package.



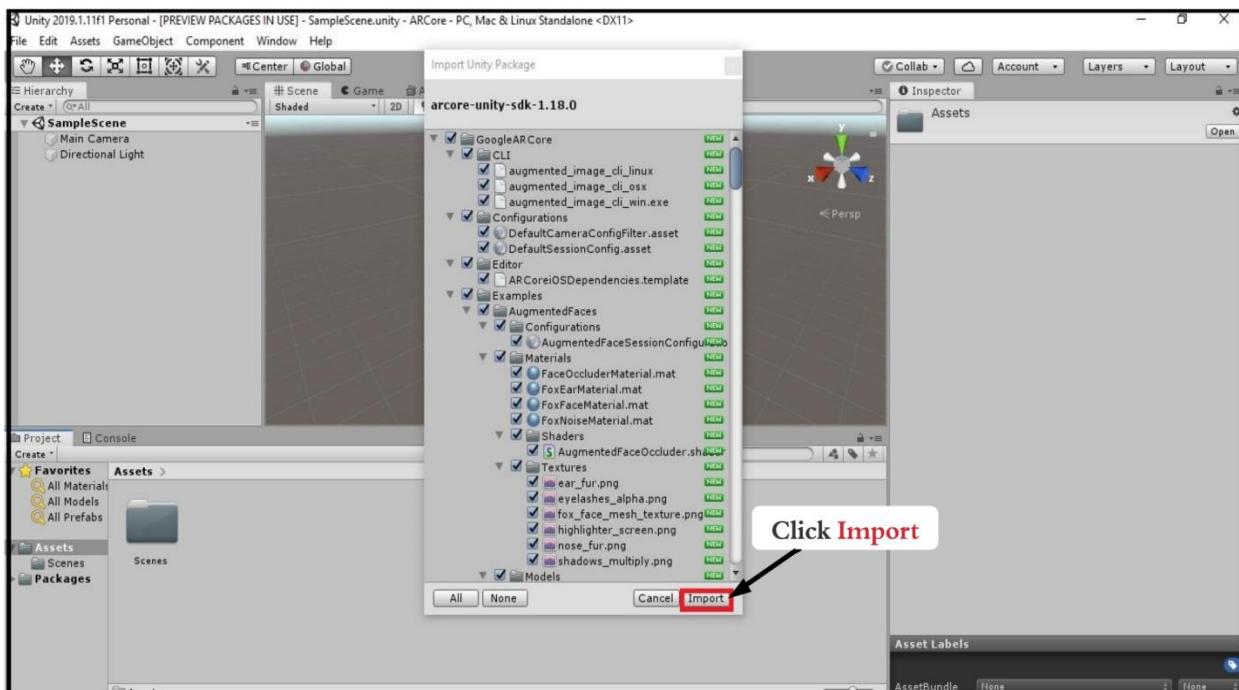
STEP 10:

Import the ARCore SDK package.



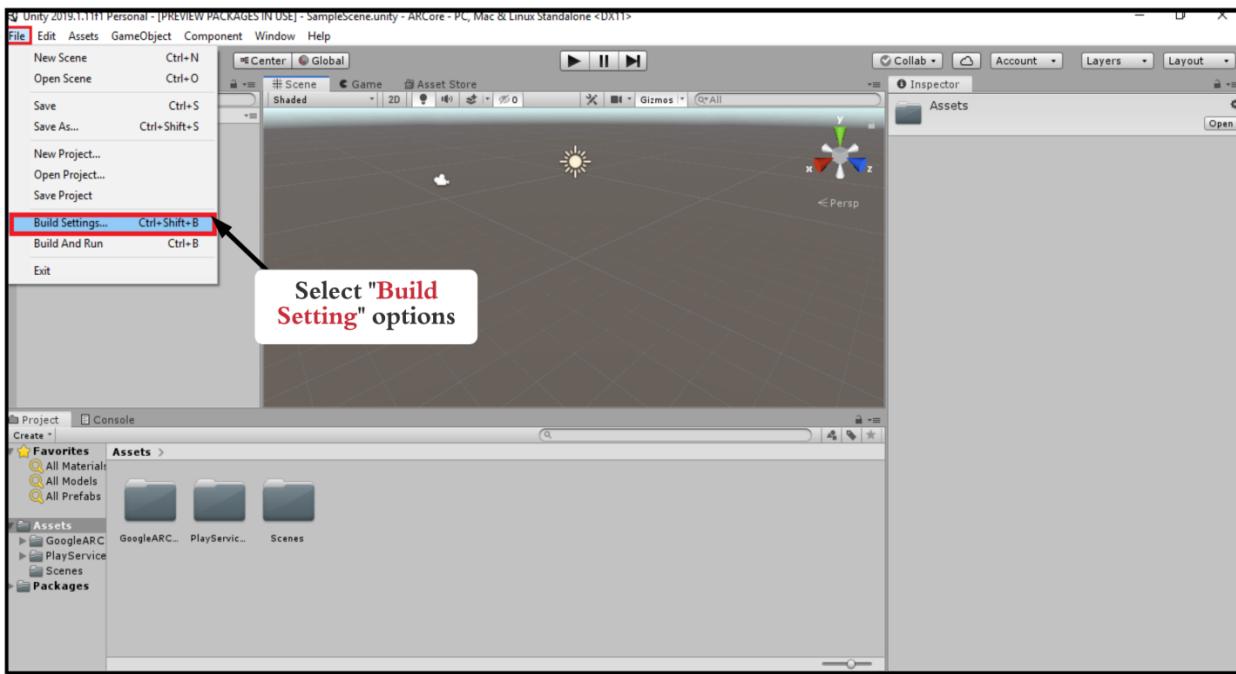
STEP 11:

Click Import.



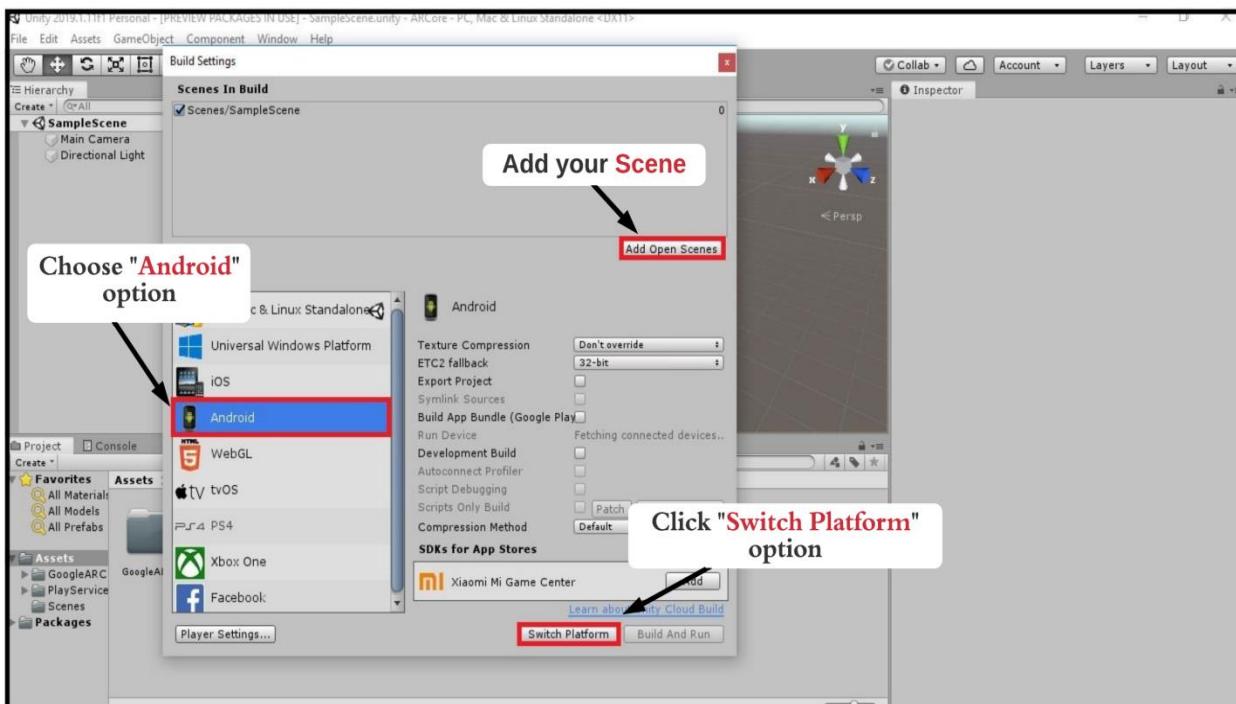
STEP 12:

Go to File menu → Build Settings.



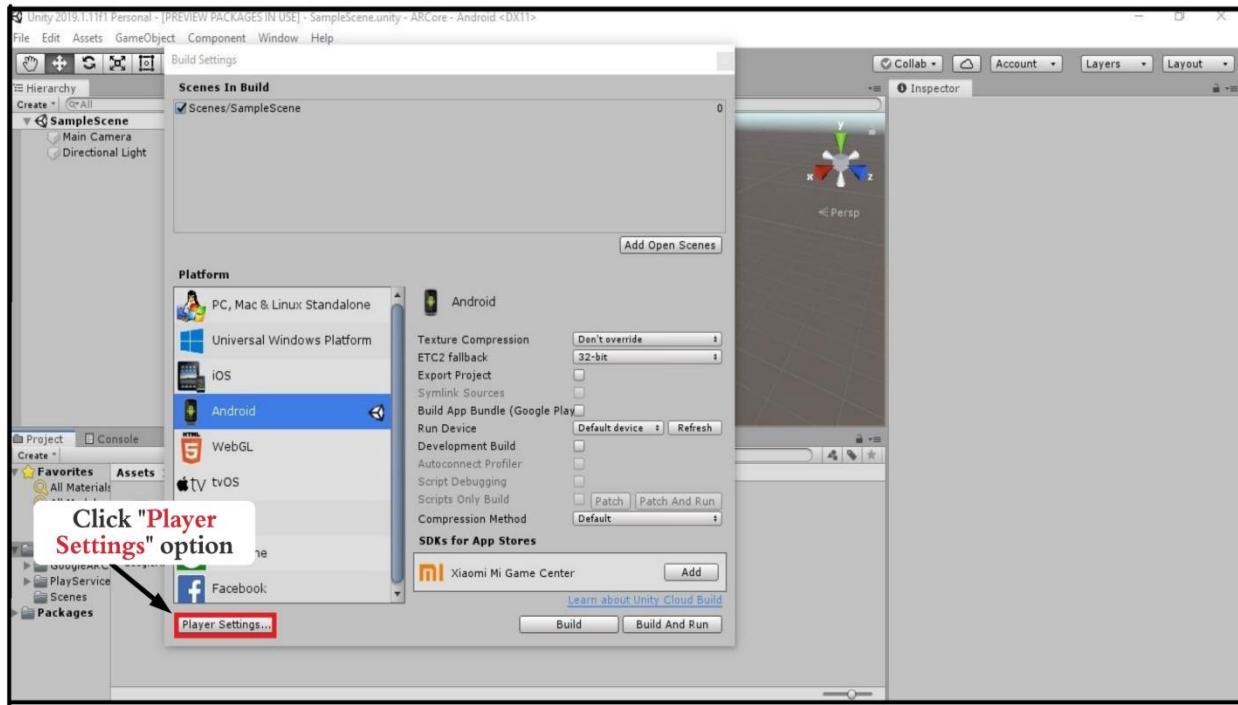
STEP 13:

Switch your Platform to “Android” to run it in your mobile phone.



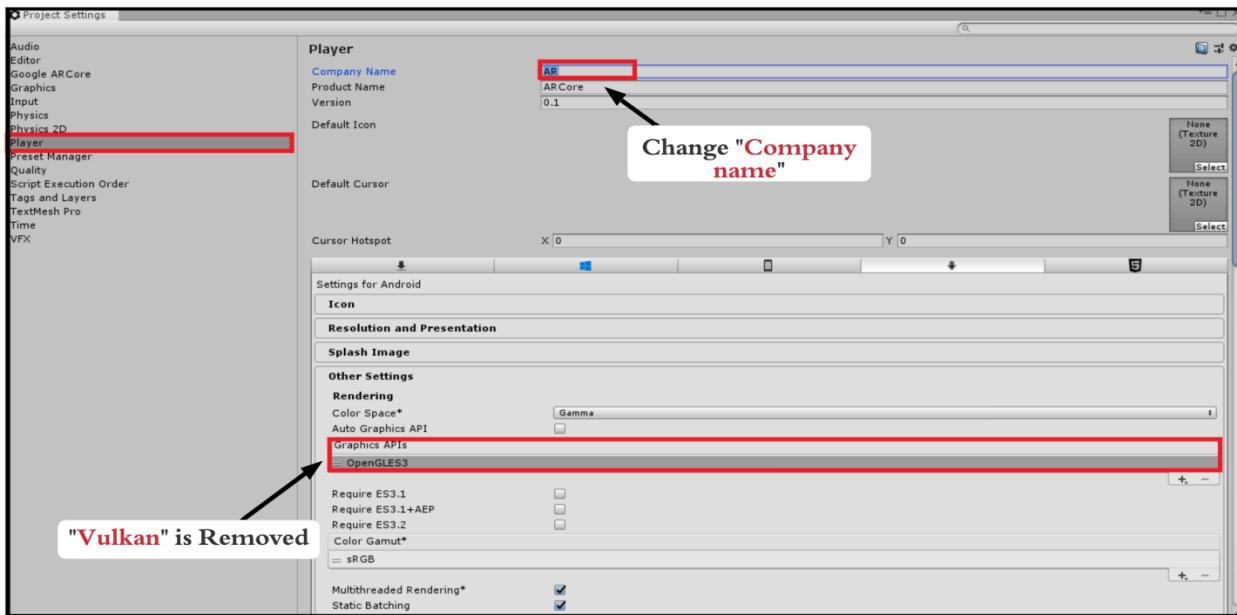
STEP 14:

Go to Player settings to make some changes for supporting ARCore plugin.



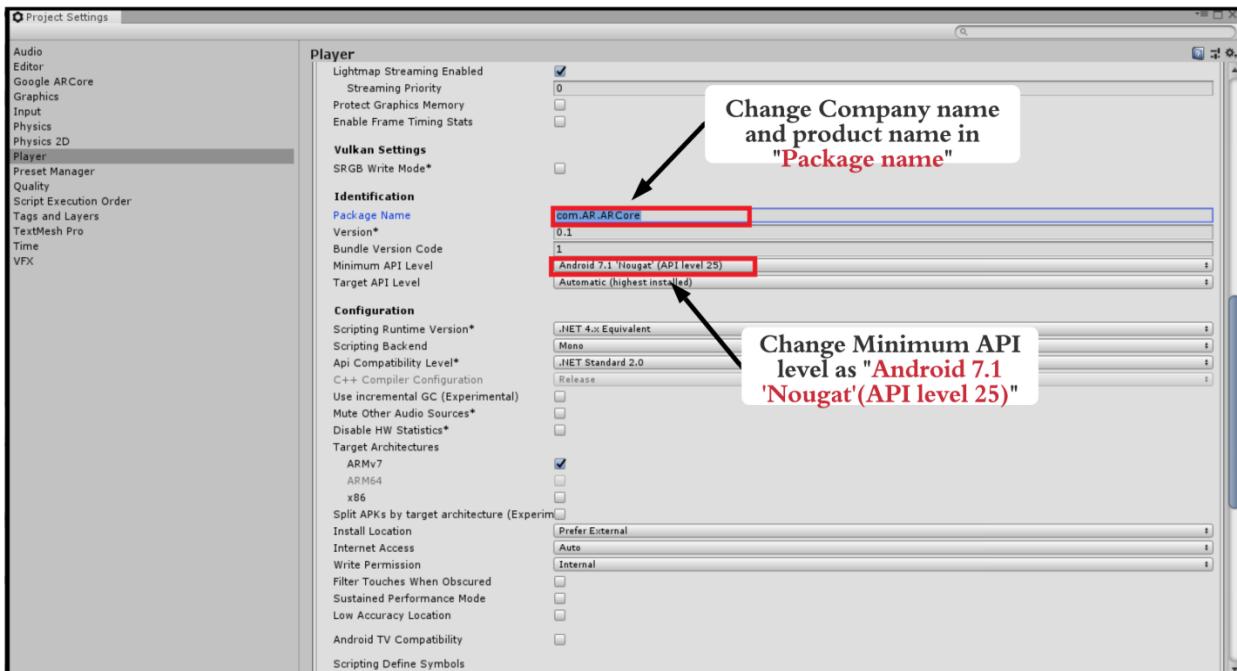
STEP 15:

In player, you need to change company name as of your wish. For example "AR" and product name will be taken by itself that's your Unity project name and remove "Vulkan".



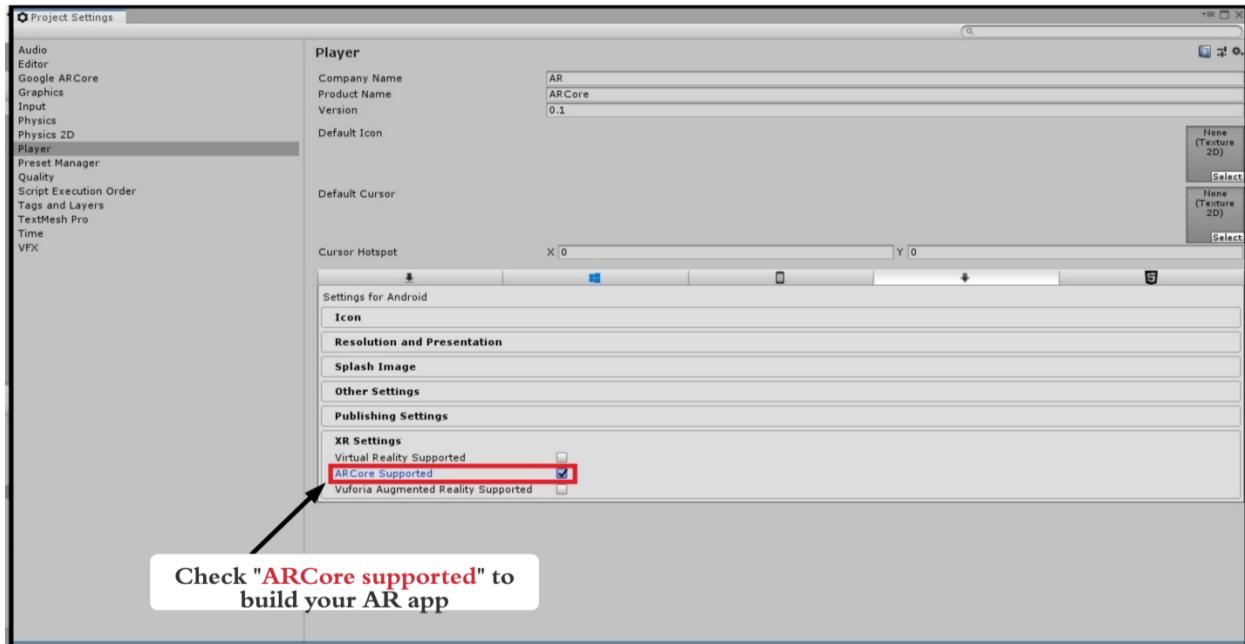
STEP 16:

Select Other settings → Bundle identifier and change as of given below same as you gave for above company and product name. → Minimum API Level must be above 25 and Maximum API Level may be of your choice.



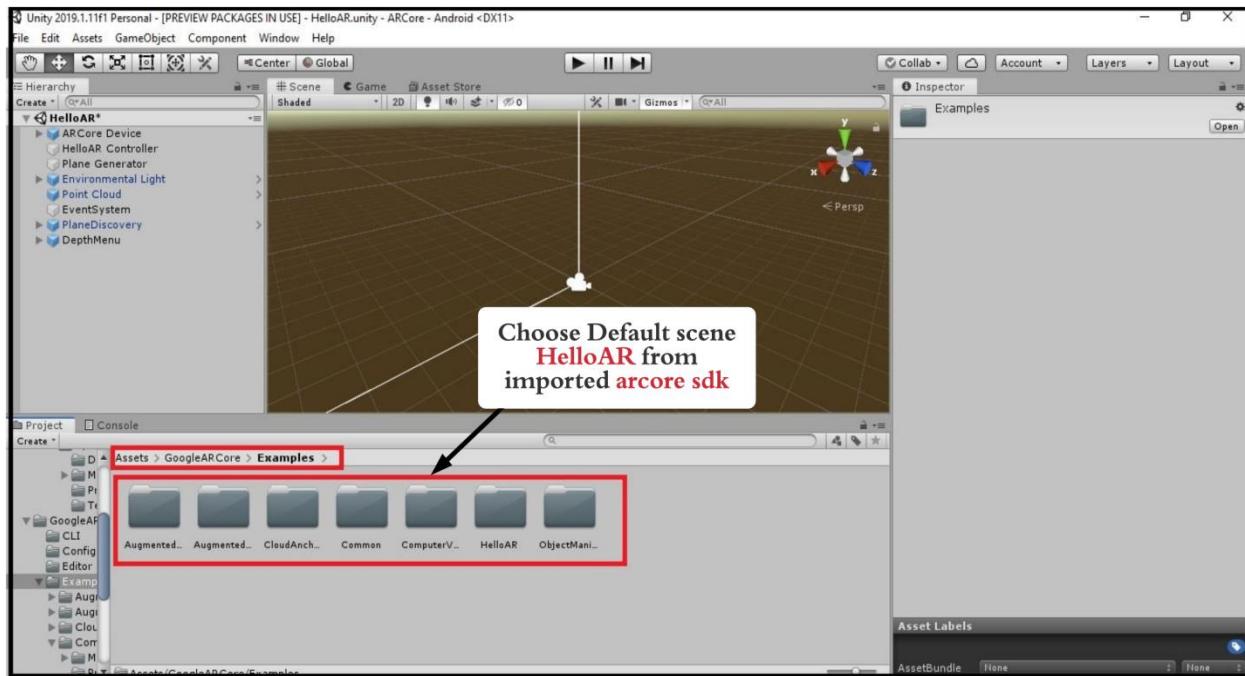
STEP 17:

Go to Player Settings → XR Settings, Check ARCore supported for it creates ARCore plugin supporter.



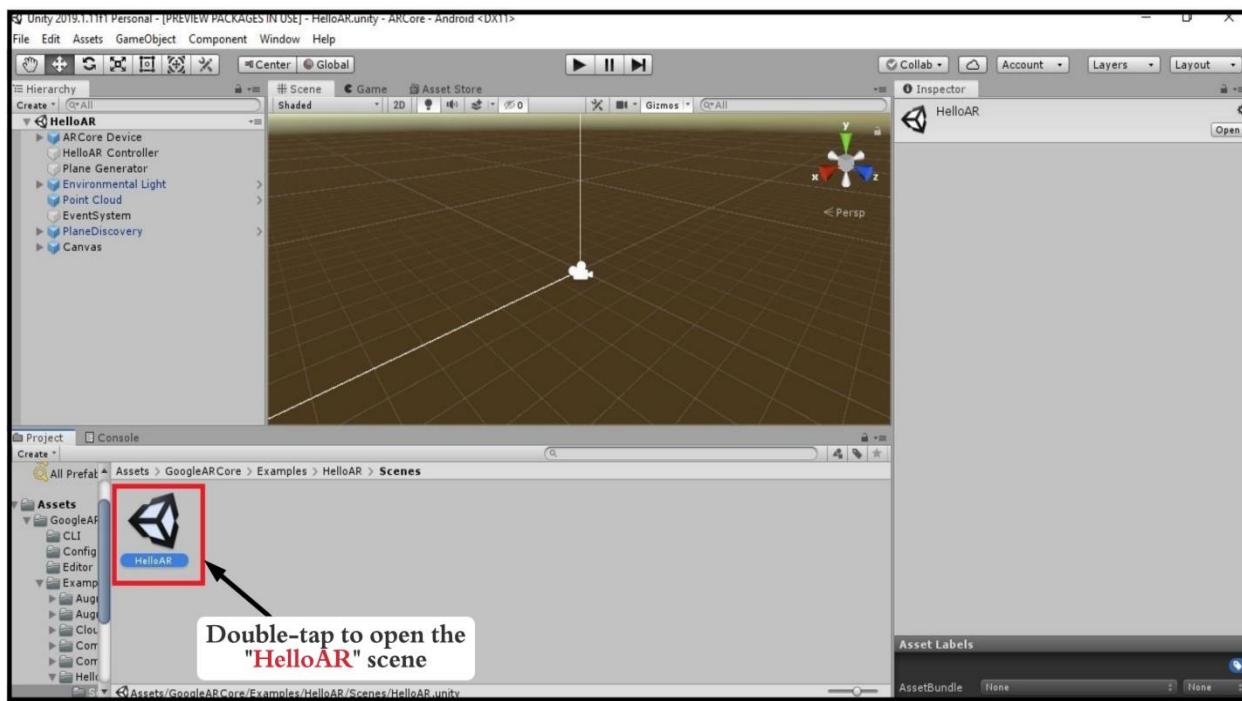
STEP 18:

In Project window, go to Assets → GoogleARCore → Examples. Choose default scene “HelloAR”.



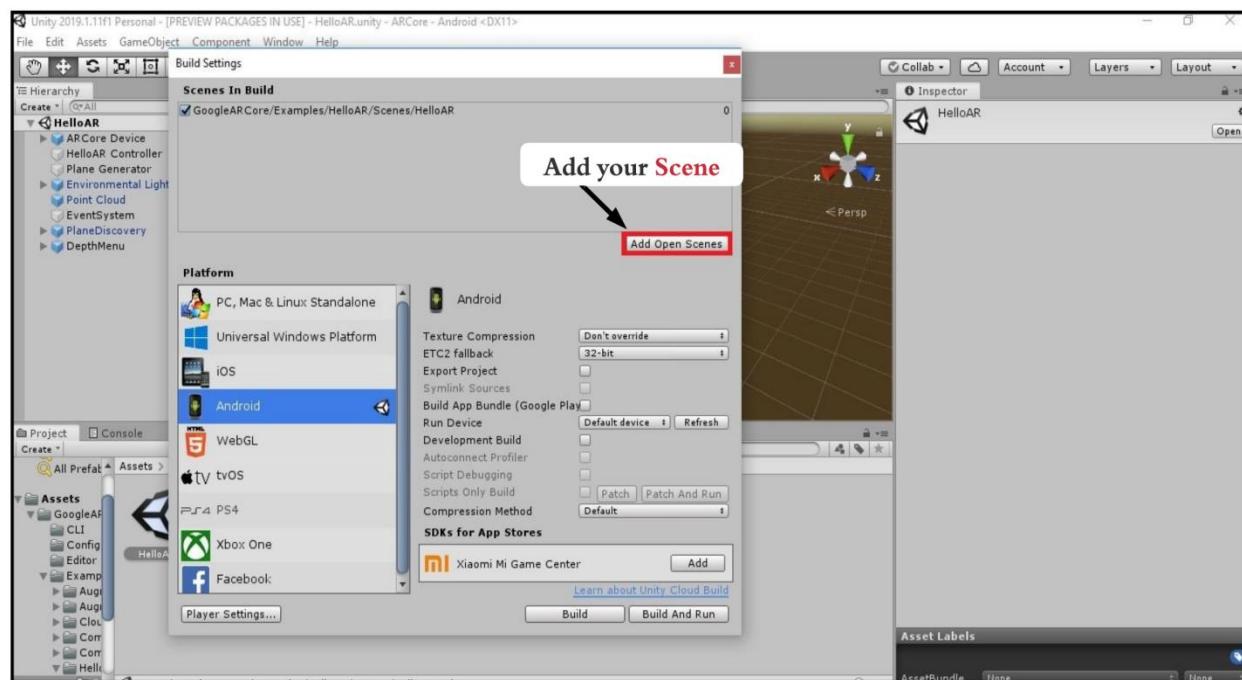
STEP 19:

Double-tap to open HelloAR scene.



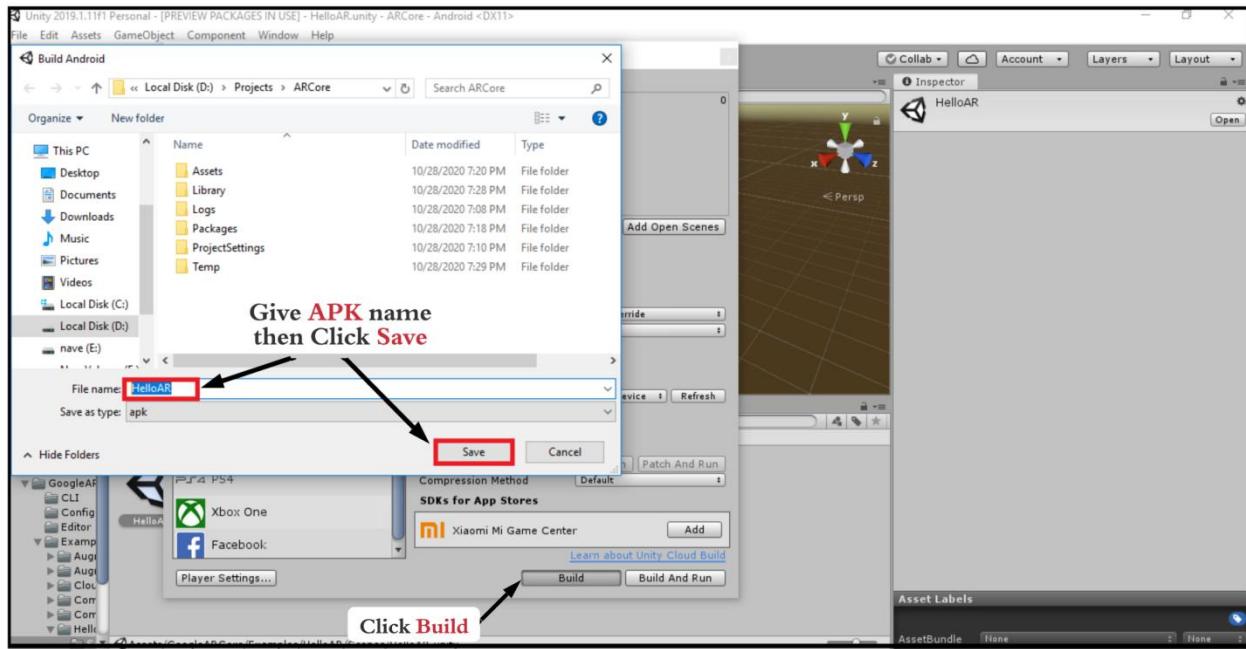
STEP 20:

Add your scene "HelloAR" to your Scenes in Build.



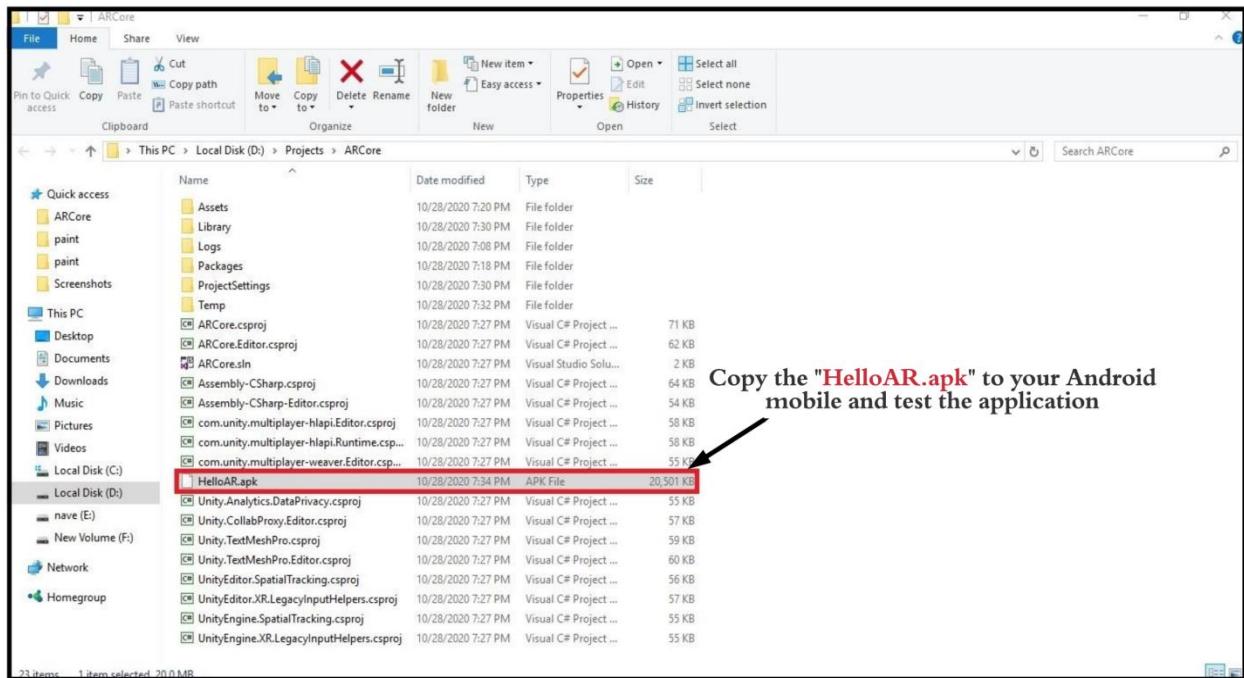
STEP 21:

Click Build, Give Apk name of your choice. For example “HelloAR” and click Save.



STEP 22:

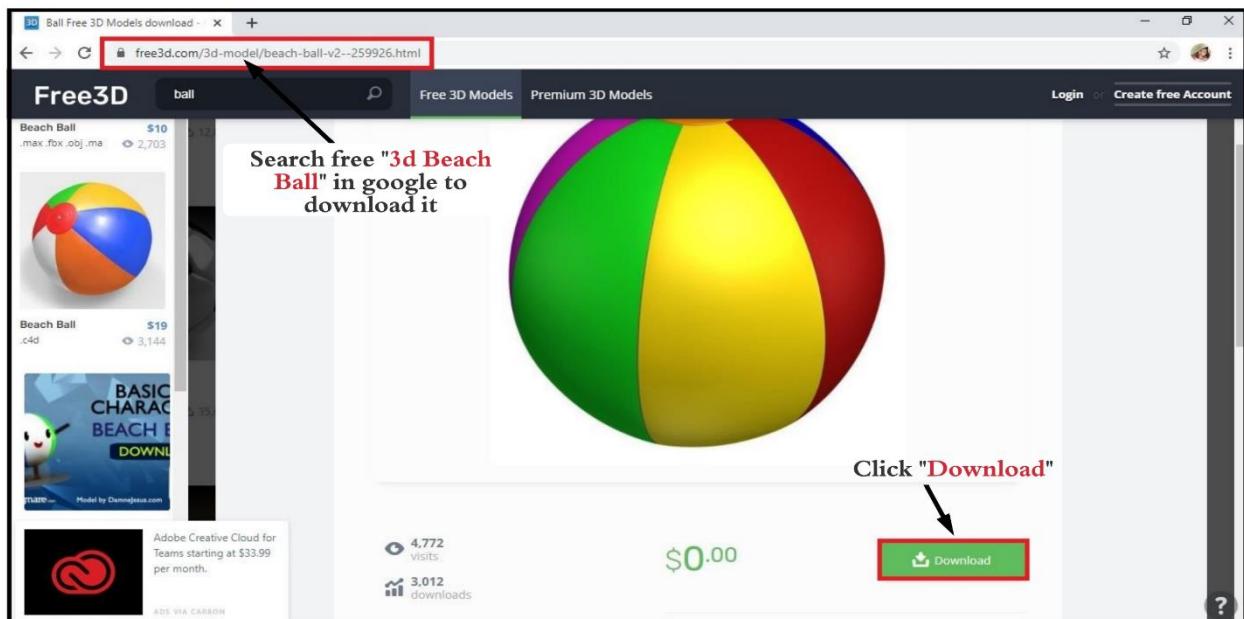
Apk is built now , copy the apk to your Android mobile and test the application.



Replacing Default 3D model in ARCore

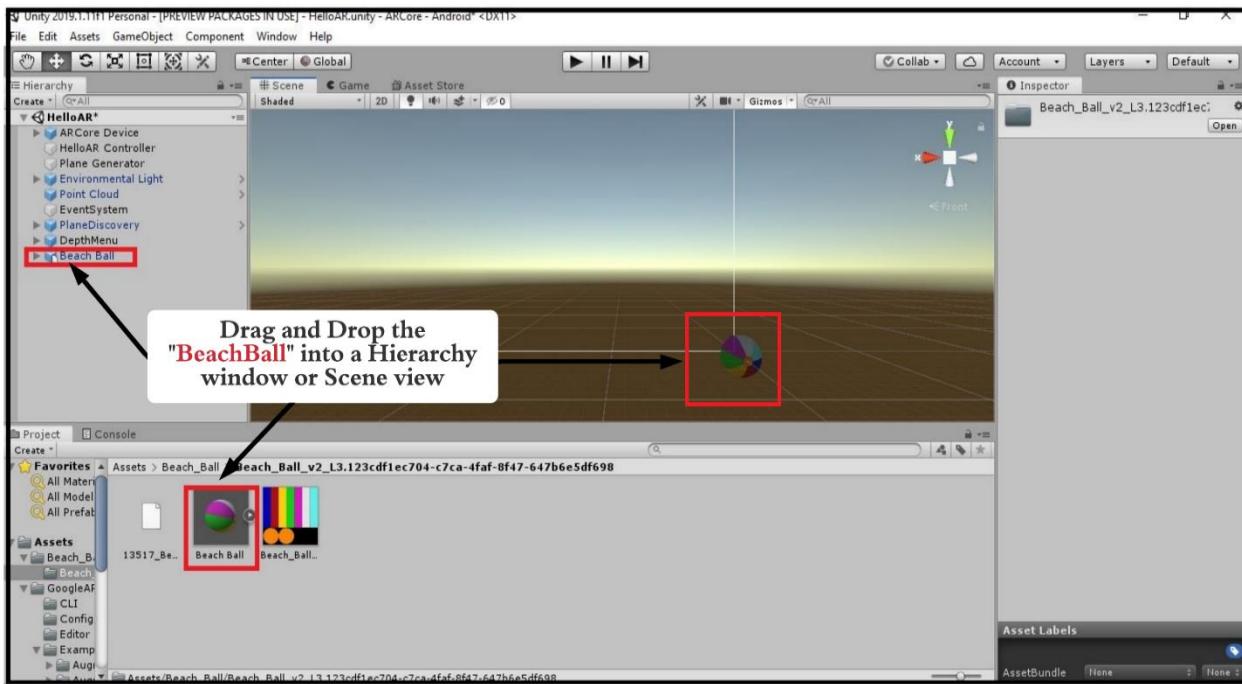
STEP 23:

Go to Google and search for any 3D model you wish. For example “Beach Ball”.



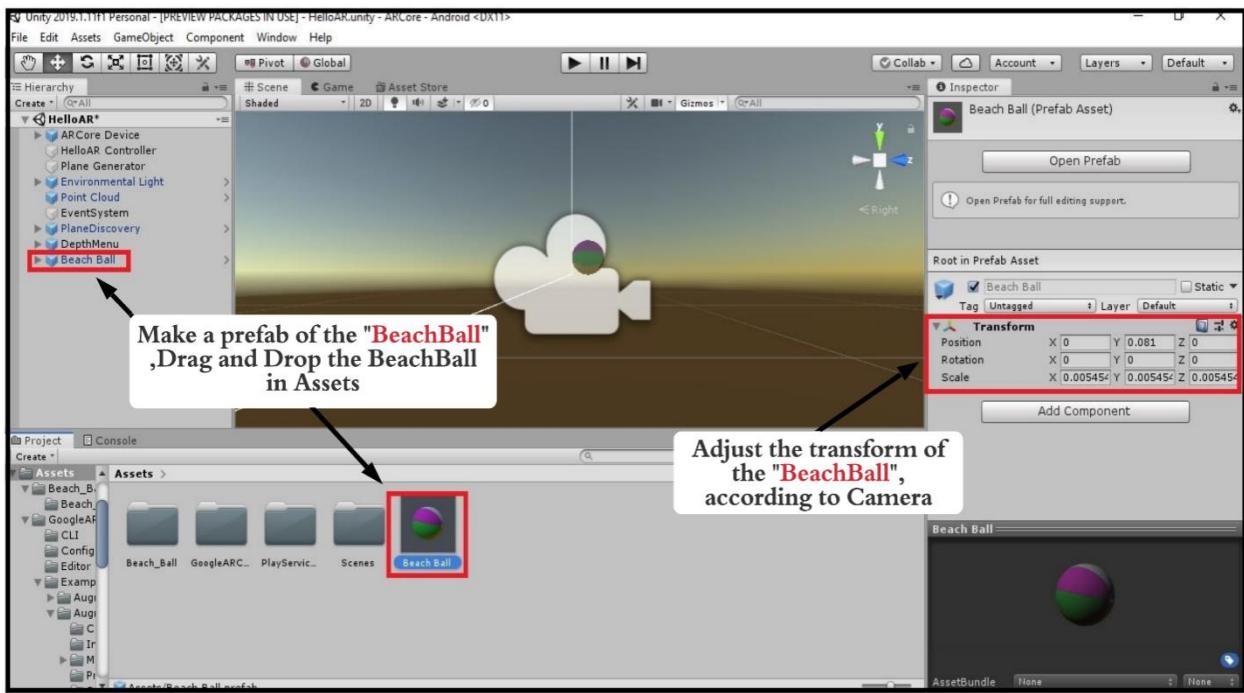
STEP 24:

You need to replace the default gameObject with Beach ball.



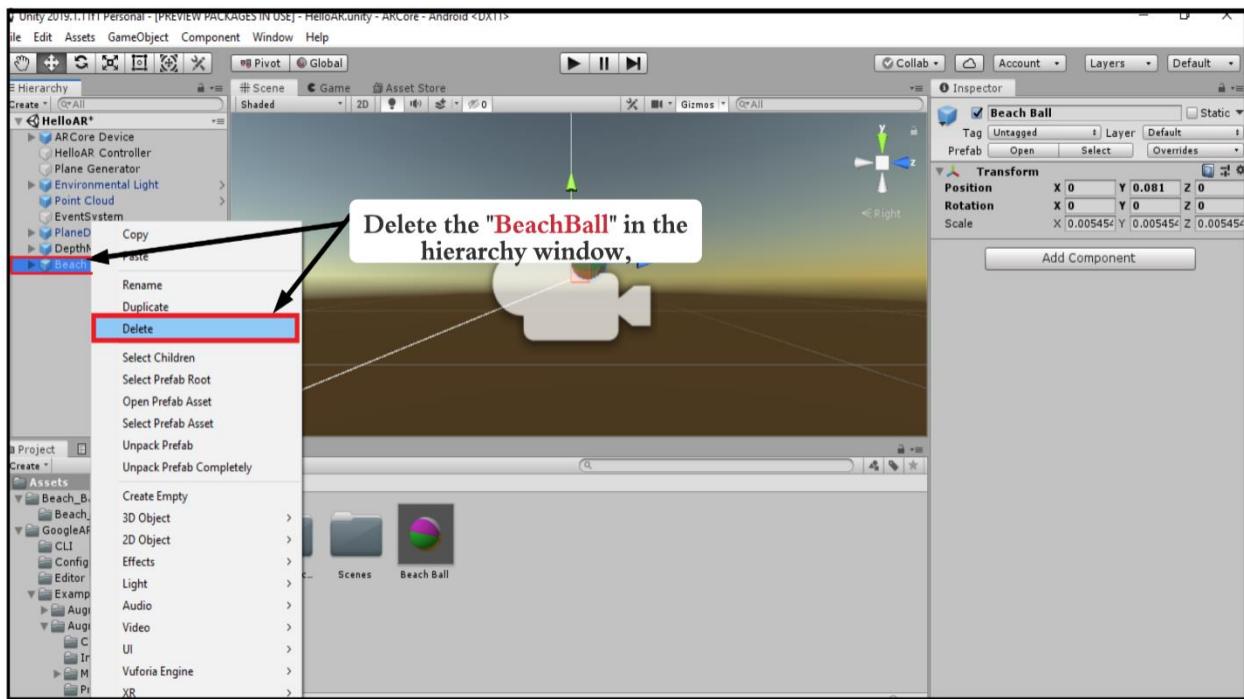
STEP 25:

Check with the transforms of this Beach ball object, and adjust along with the camera.



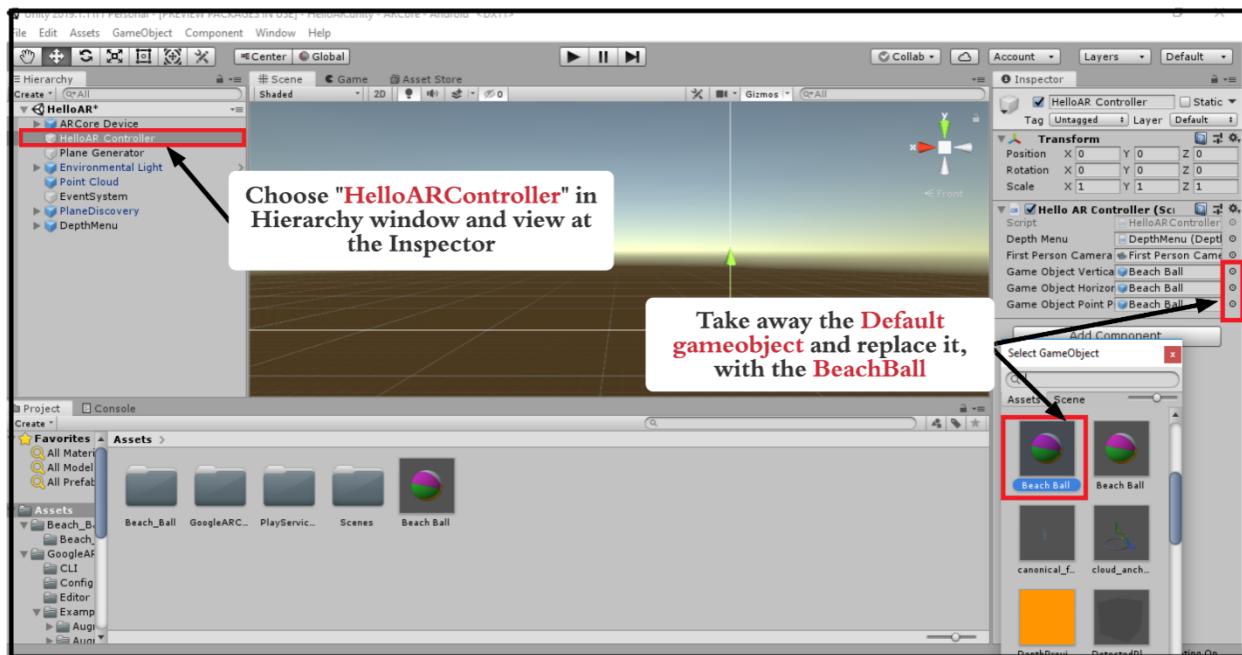
STEP 26:

Delete the Beach ball as it is no longer needed.



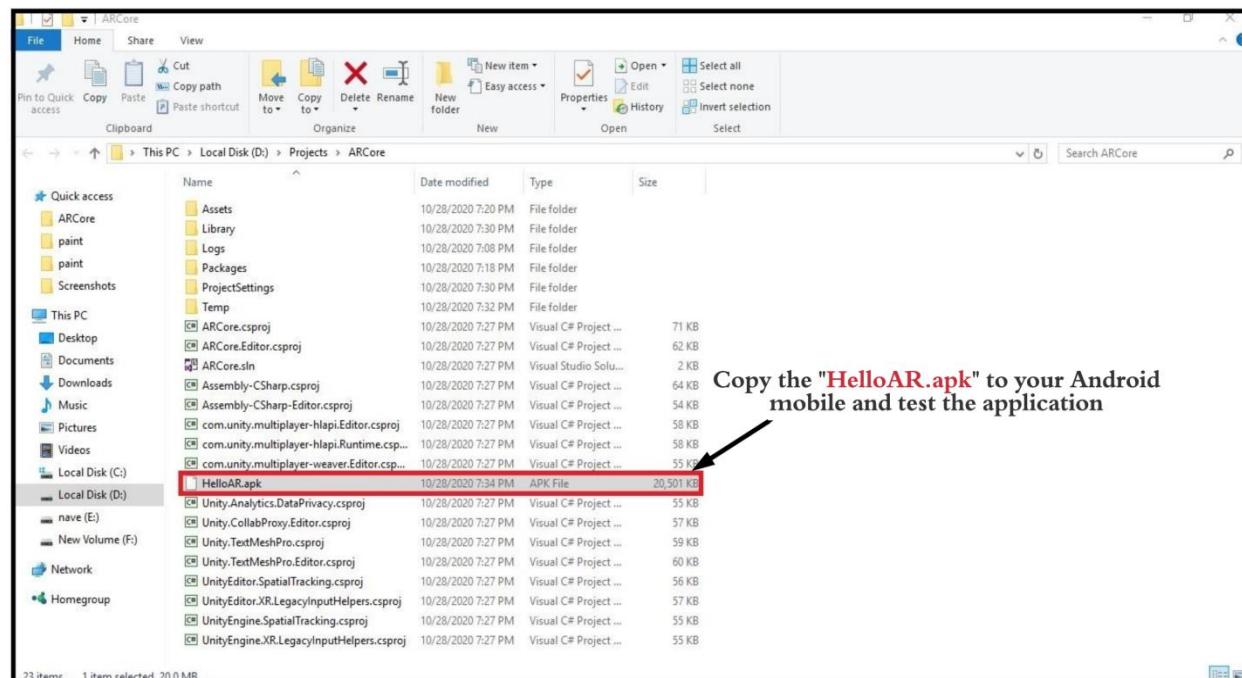
STEP 27:

In Hierarchy window, choose “HelloAR Controller” and view it in the Inspector window, you may find “HelloAR Controller(Script)” and replace Beach ball with the default object.



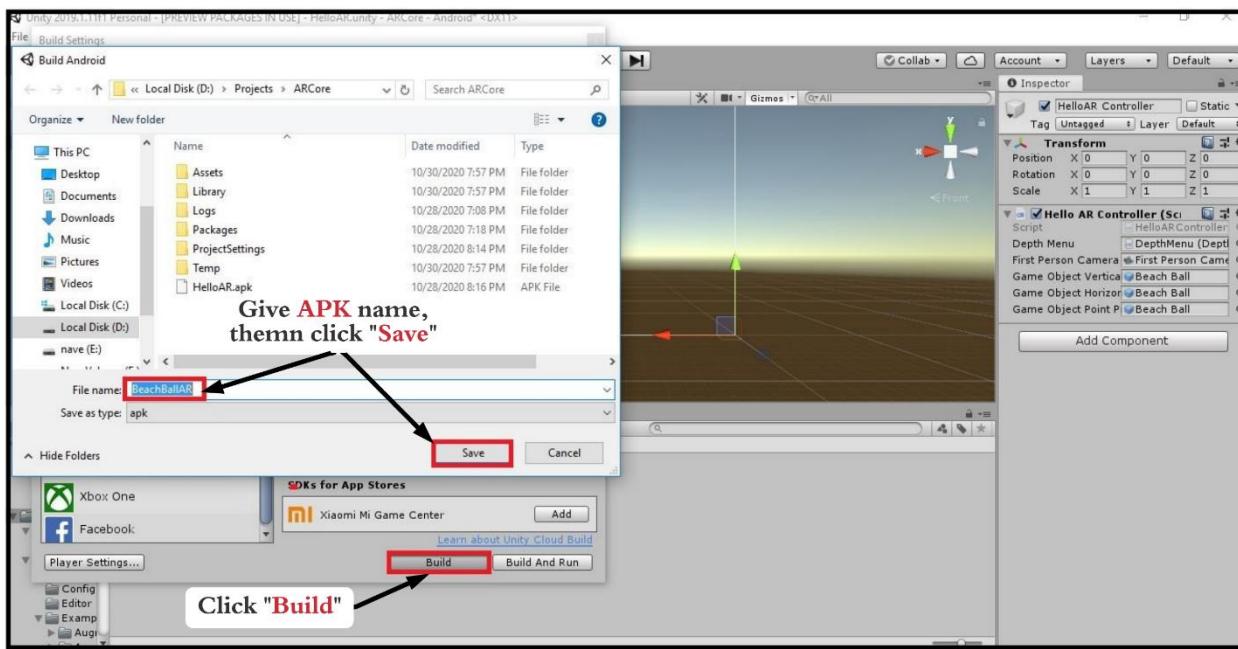
STEP 28:

Add your scene “HelloAR” to your Scenes in Build.



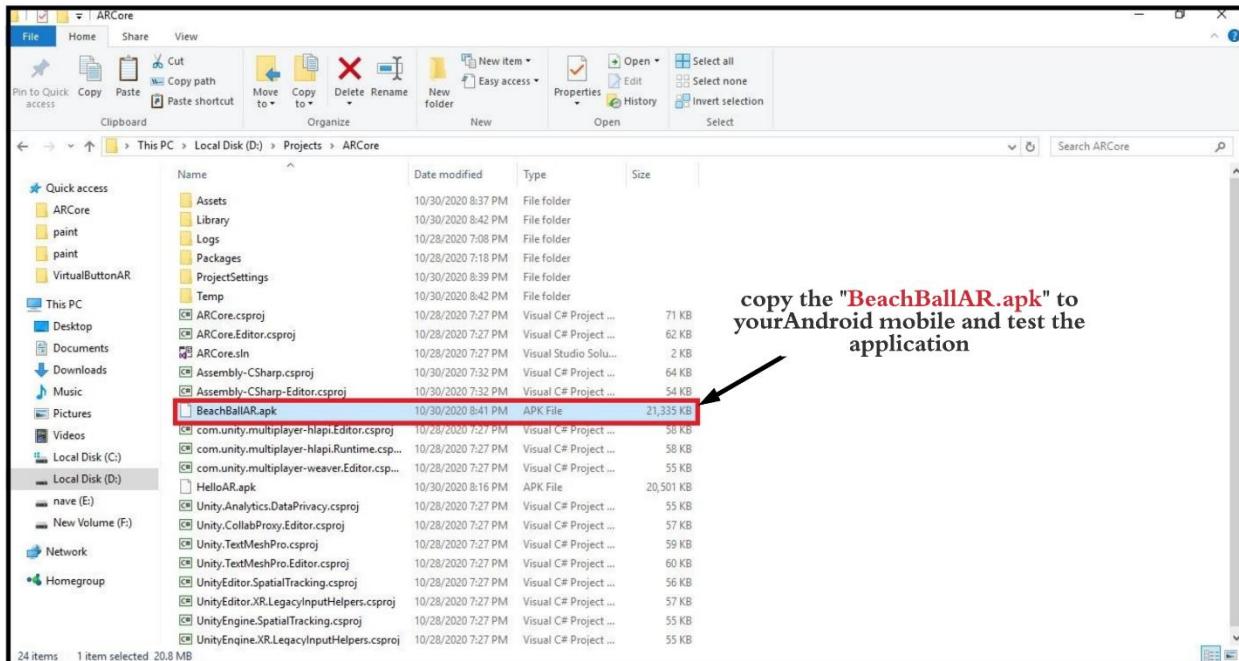
STEP 28:

Click Build, Give Apk name of your choice. For example “BeachBallAR” and click Save.



STEP 29:

Apk is built now , copy the apk to your Android mobile and test the application.



Result:

Thus AR VR as a service in cloud has been visualized successfully

Ex.No : 10

04.05.2021

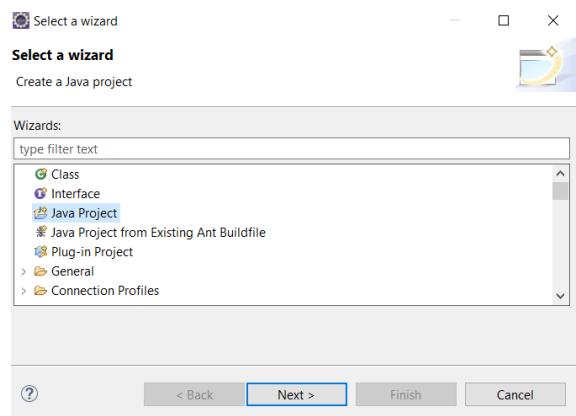
Simulation of cloud scenario using cloudsim

Aim

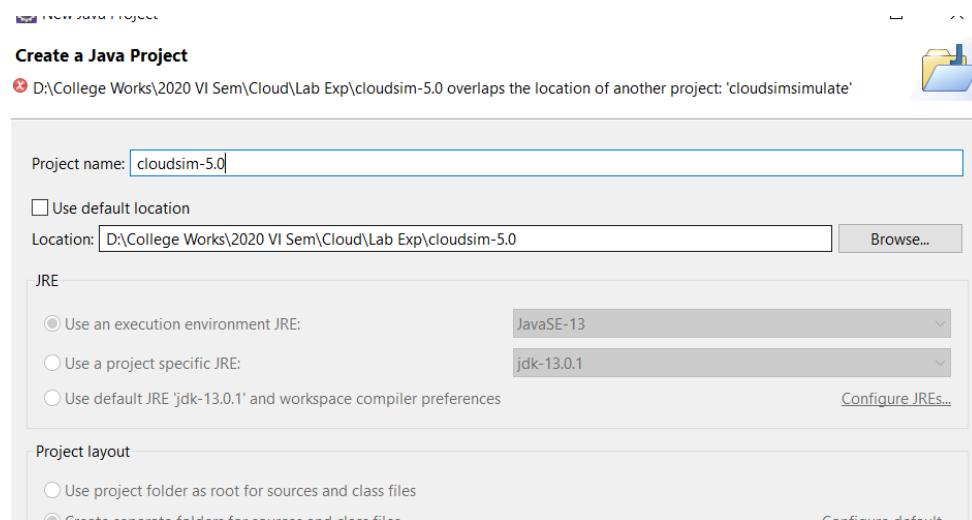
To simulate the scenarios in cloud using cloudsim and common math libraries.

Procedure

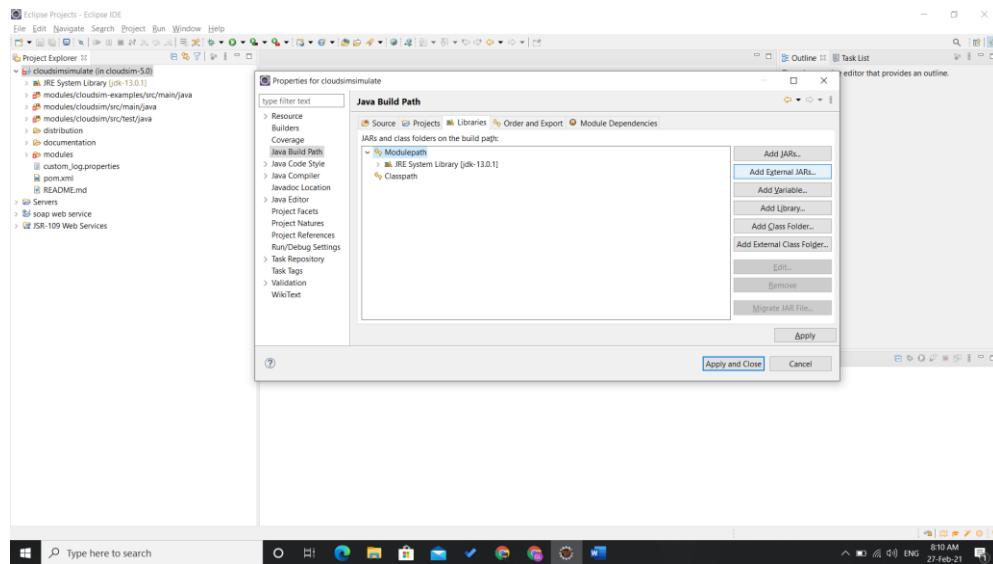
1. Create a new Java project in the cloudsim directory.



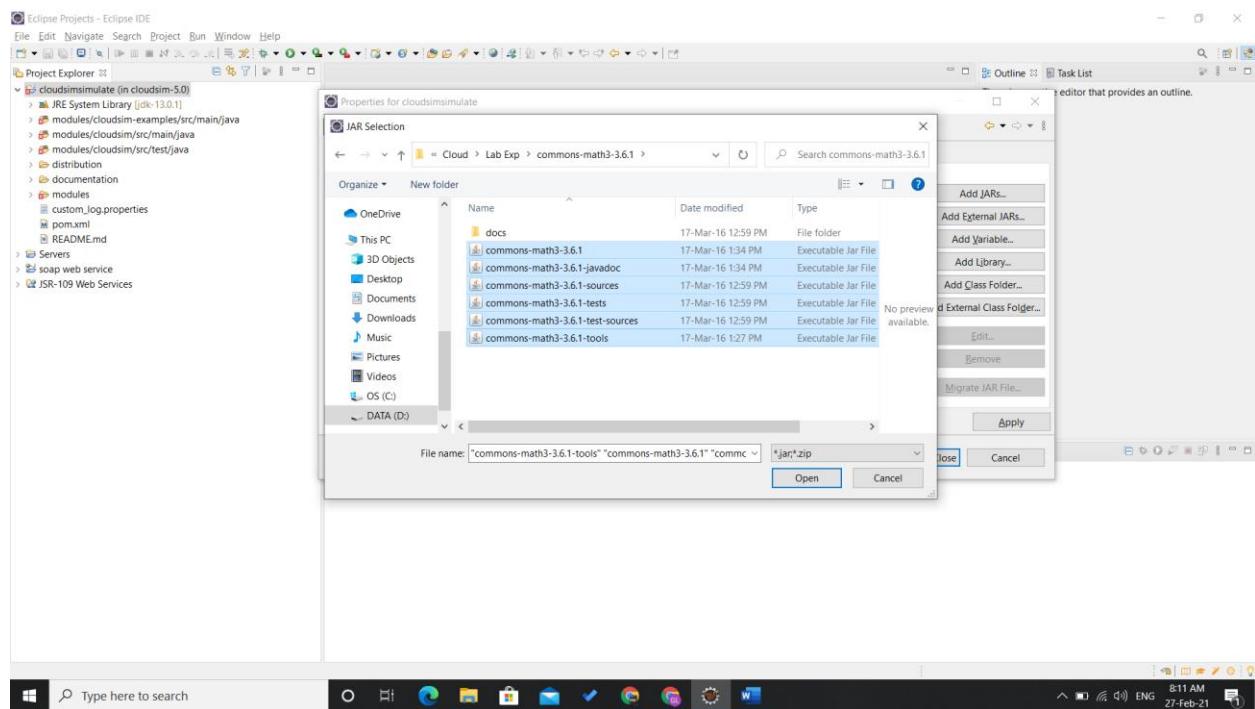
2. Give a suitable project name and select the location as same as the cloudsim location.

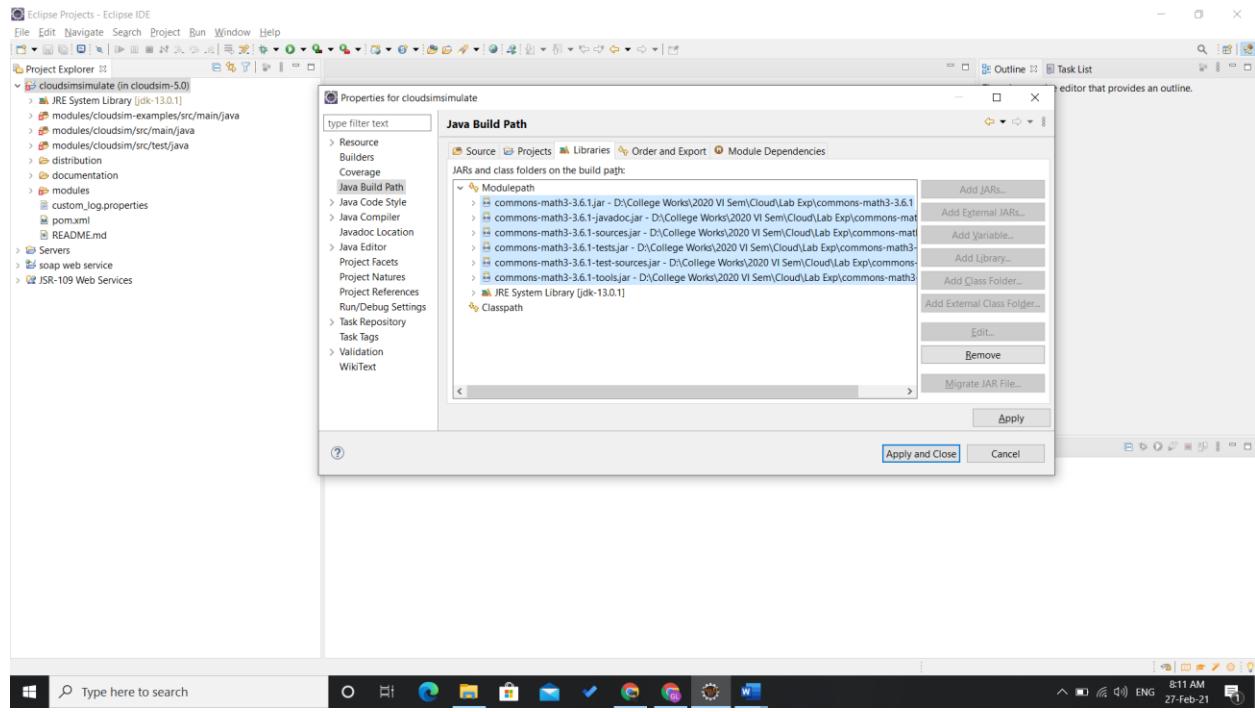


3. Go to properties and navigate to module path.

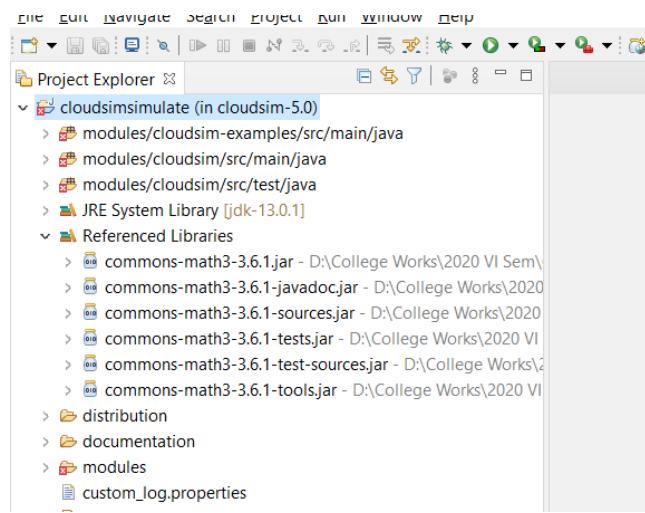


4. Add common math jar files.





5. The jar files are referenced.



6. Execute the example program under cloudsim example package.

```
<terminated> CloudSimExample1 [Java Application] C:\Program Files\Java\jdk-13.0.1\bin\javaw.exe (Feb 27, 2021, 8:14:56 AM – 8:15:03 AM)
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

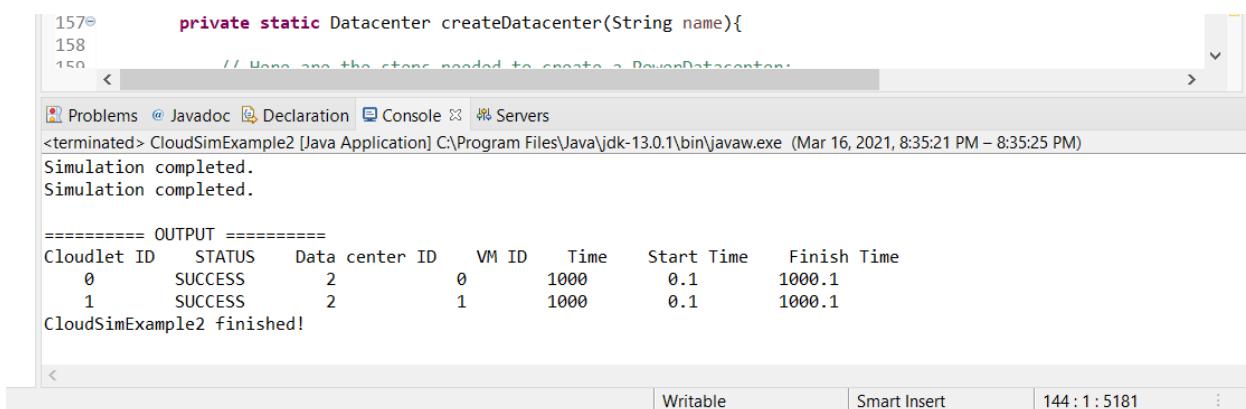
===== OUTPUT =====
Cloudlet ID    STATUS    Data center ID    VM ID    Time    Start Time    Finish Time
      0        SUCCESS         2            0       400      0.1        400.1
CloudSimExample1 finished!
```

Explanation Program 1:

1. We create two list to store cloudlets and vms.
2. Initialize cloudsim package with parameters like number of users, date and time, events.
3. Create atleast one datacenter to start the simulation. Datacenter class provides with the method createDatacenter() to do this.
4. Next is to create a broker and a vm to the cloud sim. DatacenterBroker class provides with the method createBroker() to do this. Initialize a vm with Vm class and add to the vm list.
5. Give the set of vms to the broker by using submitVmList().
6. Create utilization model with UtilizationModel class and initialize cloudlet by creating an object for the Cloudlet class.
7. Set vmId and userId for the cloudlet using setId() and setVmId() methods. Submit the cloudlet to the broker using submitCloudletList().
8. Start the simulation using startSimulation() method.

Explanation Program 2:

1. Follow the steps as mentioned in the above program upto creation of vm.
2. Add another vm by incrementing the vmId + 1;
3. Submit the vmList and get the cloudlets ready.
4. Here submit each cloudlet to unique vms by bindCloudletToVm() method.
5. Print the cloudlets added using getCloudletReceivedList() method.
6. Start the simulation using startSimulation() and stopSimulation().
7. The results will be printed.



The screenshot shows an IDE interface with a code editor and a console window. The code editor contains Java code for creating a datacenter and defining a cloudlet. The console window displays the output of the simulation, including messages about simulation completion and a detailed table of cloudlet execution results.

```
157     private static Datacenter createDatacenter(String name){  
158         // Here are the steps needed to create a PowerDatacenter:  
159     }  
  
Problems @ Javadoc Declaration Console Servers  
<terminated> CloudSimExample2 [Java Application] C:\Program Files\Java\jdk-13.0.1\bin\javaw.exe (Mar 16, 2021, 8:35:21 PM – 8:35:25 PM)  
Simulation completed.  
Simulation completed.  
  
===== OUTPUT =====  
Cloudlet ID      STATUS      Data center ID    VM ID      Time      Start Time      Finish Time  
      0      SUCCESS          2            0        1000       0.1        1000.1  
      1      SUCCESS          2            1        1000       0.1        1000.1  
CloudSimExample2 finished!
```

Result

Thus, the simulation of cloudsim is analyzed and the result is obtained.

Ex.No : 11
05.05.2021

Study on Private Cloud Implementation using OpenStack

Aim:

To create a private cloud using openstack and configure VM

Description:
Private cloud refers to a model of cloud computing where IT services are provisioned over private IT infrastructure for the dedicated use of a single organization. A private cloud is usually managed via internal resources. OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface.

System Requirements:

Operating system:

Ubuntu

- Ubuntu 16.04 LTS (Xenial Xerus)

CentOS (support is experimental)

- Centos 7, fully updated.

Processor- 1,

Memory -4 GB

Storage- 10 GB

a) Procedure to install openstack single node

1.System with a fresh install of Linux

2.Add your user

We need to add a user to install DevStack. (if you created a user during install you can skip this step and just give the user sudo privileges below) \$ sudo useradd -s /bin/bash -d /opt/stack -m stack

Since this user will be making many changes to your system, it will need to have sudo privileges:

```
$ apt-get install sudo -y || yum install -y sudo  
$ echo "stack ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers
```

3.Download DevStack

We'll grab the latest version of DevStack via https:

```
$ sudo apt-get install git -y || sudo yum install -y git  
$ git clone https://git.openstack.org/openstack-dev/devstack $ cd devstack
```

4.Run DevStack

Now to configure stack.sh. DevStack includes a sample in devstack/samples/local.conf. Create local.conf as shown below to do the following:

- Set FLOATING_RANGE to a range not used on the local network, i.e. 192.168.1.224/27. This configures IP addresses ending in 225-254 to be used as floating IPs.
- Set FIXED_RANGE and FIXED_NETWORK_SIZE to configure the internal address space used by the instances.

- Set FLAT_INTERFACE to the Ethernet interface that connects the host to your local network. This is the interface that should be configured with the static IP address mentioned above.
- Set the administrative password. This password is used for the **admin** and **demo** accounts set up as OpenStack users.
- Set the MySQL administrative password. The default here is a random hex string which is inconvenient if you need to look at the database directly for anything.
- Set the RabbitMQ password.
- Set the service password. This is used by the OpenStack services (Nova, Glance, etc) to authenticate with Keystone.

local.conf should look something like this:

```
[[local|localrc]]
```

```
FLOATING_RANGE=192.168.1.224/27
```

```
FIXED_RANGE=10.11.12.0/24
```

```
FIXED_NETWORK_SIZE=256
```

```
FLAT_INTERFACE=eth0
```

```
ADMIN_PASSWORD=supersecret
```

```
DATABASE_PASSWORD=iheartdatabases
```

```
RABBIT_PASSWORD=flopsymopsy
```

```
SERVICE_PASSWORD=iheartksl
```

Run DevStack:

```
$ ./stack.sh
```

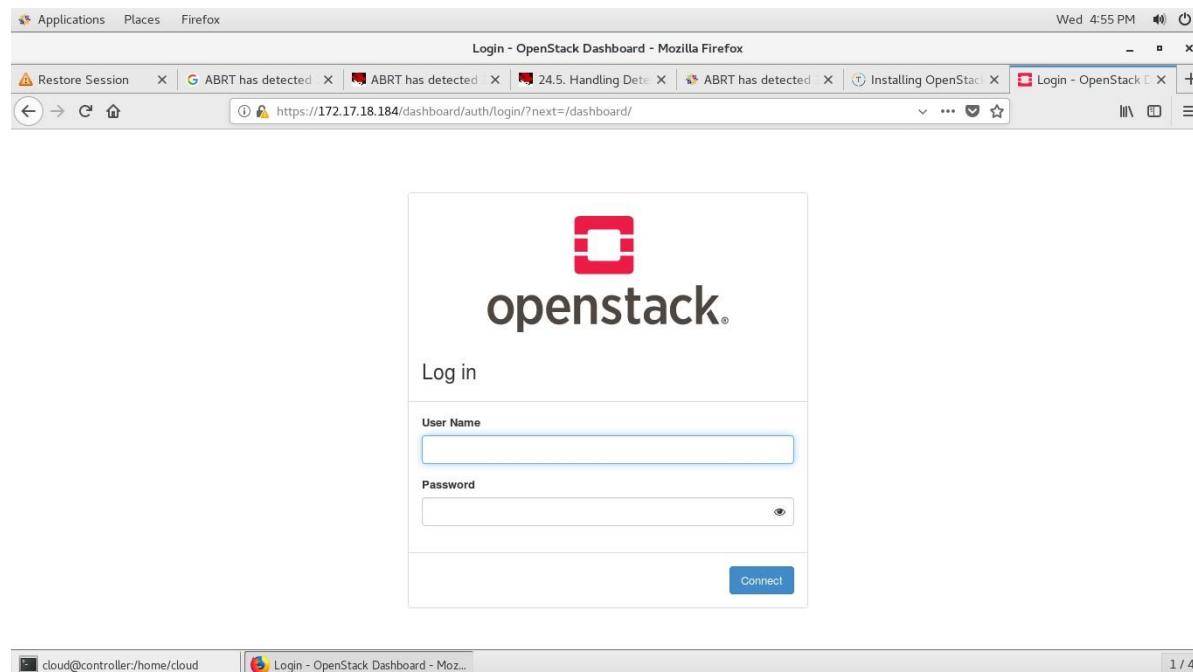
A seemingly endless stream of activity ensues. When complete you will see a summary of stack.sh's work, including the relevant URLs, accounts and passwords to poke at your shiny new OpenStack.

Using OpenStack

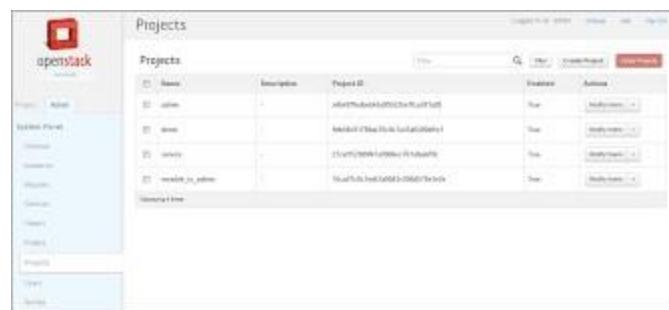
At this point you should be able to access the dashboard from other computers on the local network. In this example that would be <http://192.168.1.201/> for the dashboard (aka Horizon). Launch VMs and if you give them floating IPs and security group access those VMs will be accessible from other machines on your network.

b) Procedure to Launch an instance in openstack

1. Log in to the dashboard.

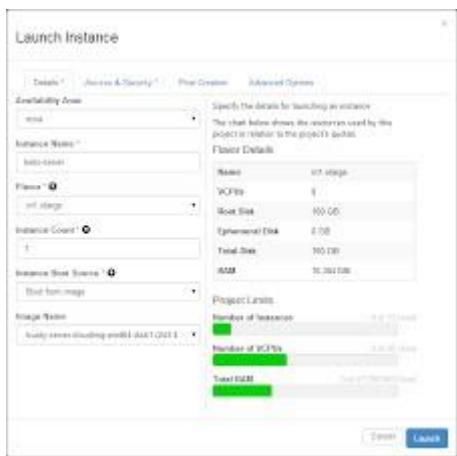


2. Select the appropriate project from the drop down menu at the top left.

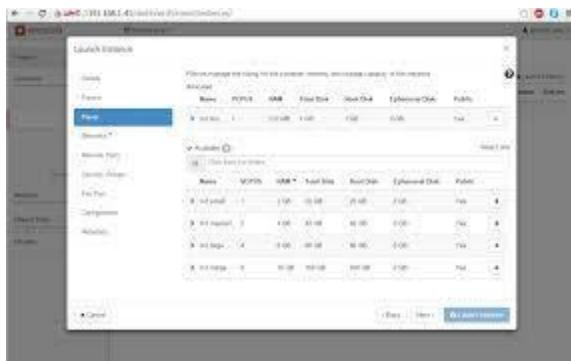


3. On the Project tab, open the Compute tab and click Instances category.

The dashboard shows the instances with its name, its private and floating IP addresses, size, status, task, power state, and so on.



4. Click Launch Instance.



5. In the Launch Instance dialog box, specify the following values:

Details tab

Instance Name

Assign a name to the virtual machine.

Description

You can assign a brief description of the virtual machine.

Availability Zone

By default, this value is set to the availability zone given by the cloud provider (for example, us-west or apac-south). For some cases, it could be nova.

Count

To launch multiple instances, enter a value greater than 1. The default is 1.

Source tab Instance Boot Source

Your options are:

Boot from image

If you choose this option, a new field for Image Name displays. You can select the image from the list.

Boot from snapshot

If you choose this option, a new field for Instance Snapshot displays. You can select the snapshot from the list.

Boot from volume

If you choose this option, a new field for Volume displays. You can select the volume from the list.

Boot from image (creates a new volume)

With this option, you can boot from an image and create a volume by entering the Device Size and Device Name for your volume. Click the Delete Volume on Instance Delete option to delete the volume on deleting the instance.

Boot from volume snapshot (creates a new volume)

Using this option, you can boot from a volume snapshot and create a new volume by choosing Volume Snapshot from a list and adding a Device Name for your volume. Click the Delete Volume on Instance Delete option to delete the volume on deleting the instance.

Image Name

This field changes based on your previous selection. If you have chosen to launch an instance using an image, the Image Name field displays. Select the image name from the dropdown list. Instance Snapshot

This field changes based on your previous selection. If you have chosen to launch an instance using a snapshot, the Instance Snapshot field displays. Select the snapshot name from the dropdown list.

Volume

This field changes based on your previous selection. If you have chosen to launch an instance using a volume, the Volume field displays. Select the volume name from the dropdown list. If you want to delete the volume on instance delete, check the Delete Volume on Instance Delete option.

Flavor tab

Flavor

Specify the size of the instance to launch.

Networks tab

Selected Networks

To add a network to the instance, click the + in the Available field.

Network Ports tab

Ports

Activate the ports that you want to assign to the instance.

Security Groups tab

Security Groups

Activate the security groups that you want to assign to the instance.

Security groups are a kind of cloud firewall that define which incoming network traffic is forwarded to instances.

If you have not created any security groups, you can assign only the default security group to the instance.

Key Pair tab

Key Pair

Specify a key pair.

If the image uses a static root password or a static key set (neither is recommended), you do not need to provide a key pair to launch the instance.

Configuration tab

Customization Script Source

Specify a customization script that runs after your instance launches.

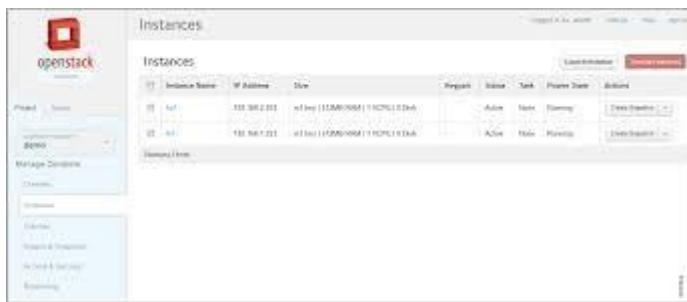
Metadata tab

Available Metadata

Add Metadata items to your instance.

6. Click Launch Instance.

The instance starts on a compute node in the cloud.



Result:

Thus, a private cloud using openstack has been created successfully.

Aim:

To implement mapreduce program in hadoop

Algorithm:

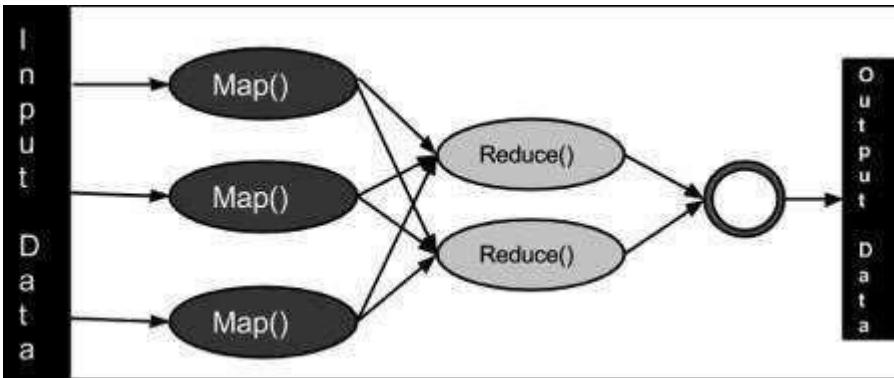
- Generally MapReduce paradigm is based on sending the computer to where the data resides!
- MapReduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.
- **Map stage** – The map or mapper's job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.
- **Reduce stage** – This stage is the combination of the **Shuffle** stage and the **Reduce** stage. The Reducer's job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.

During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.

The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.

Most of the computing takes place on nodes with data on local disks that reduces the network traffic.

After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.



Code:

```

import java.io.IOException; import java.util.StringTokenizer; import
org.apache.hadoop.conf.Configuration; import org.apache.hadoop.fs.Path; import
org.apache.hadoop.io.IntWritable; import org.apache.hadoop.io.Text; import
org.apache.hadoop.mapreduce.Job; import org.apache.hadoop.mapreduce.Mapper; import
org.apache.hadoop.mapreduce.Reducer; import
org.apache.hadoop.mapreduce.lib.input.FileInputFormat; import
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat; public class WordCount {
public static class TokenizerMapper extends Mapper<Object, Text, Text, IntWritable>{
private final static IntWritable one = new IntWritable(1); private Text word = new
Text();
public void map(Object key, Text value, Context context) throws IOException,
InterruptedException {
StringTokenizer itr = new StringTokenizer(value.toString()); while
(itr.hasMoreTokens()) {
word.set(itr.nextToken()); context.write(word, one);
} } }
public static class IntSumReducer extends
Reducer<Text,IntWritable,Text,IntWritable> { private IntWritable result = new
IntWritable();
```

```

public void reduce(Text key, Iterable<IntWritable> values, Context context ) throws
IOException, InterruptedException {

    int sum = 0;    for (IntWritable val : values) {      sum += val.get();

}

    result.set(sum);    context.write(key, result);

}  }

public static void main(String[] args) throws Exception {

    Configuration conf = new Configuration();    Job job = Job.getInstance(conf, "word
count");                                job.setJarByClass(WordCount.class);
job.setMapperClass(TokenizerMapper.class);
job.setCombinerClass(IntSumReducer.class);    job.setReducerClass(IntSumReducer.class);
job.setOutputKeyClass(Text.class);   job.setOutputValueClass(IntWritable.class);

    FileInputFormat.addInputPath(job, new Path(args[0]));

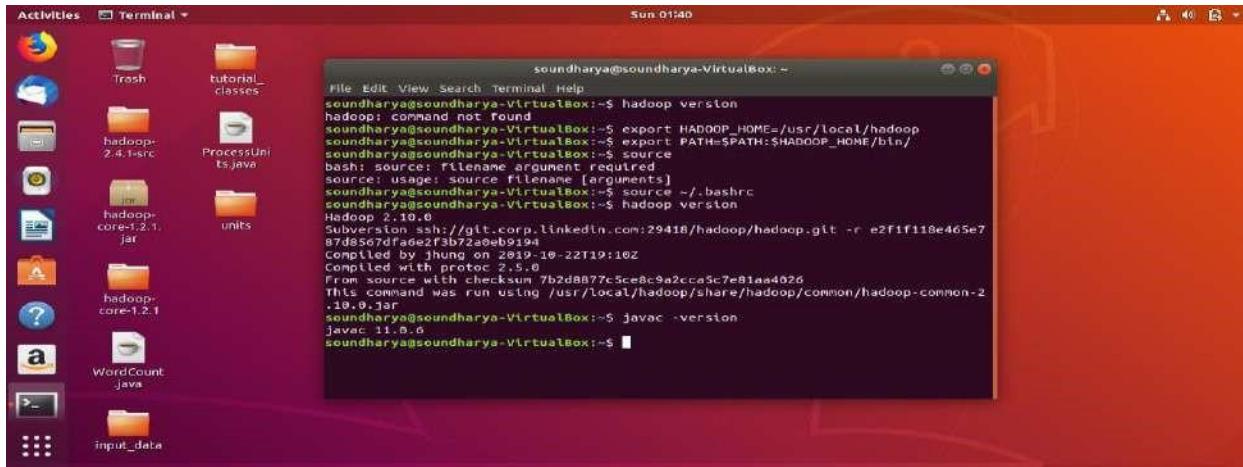
    FileOutputFormat.setOutputPath(job, new Path(args[1]));

    System.exit(job.waitForCompletion(true) ? 0 : 1);  }
}

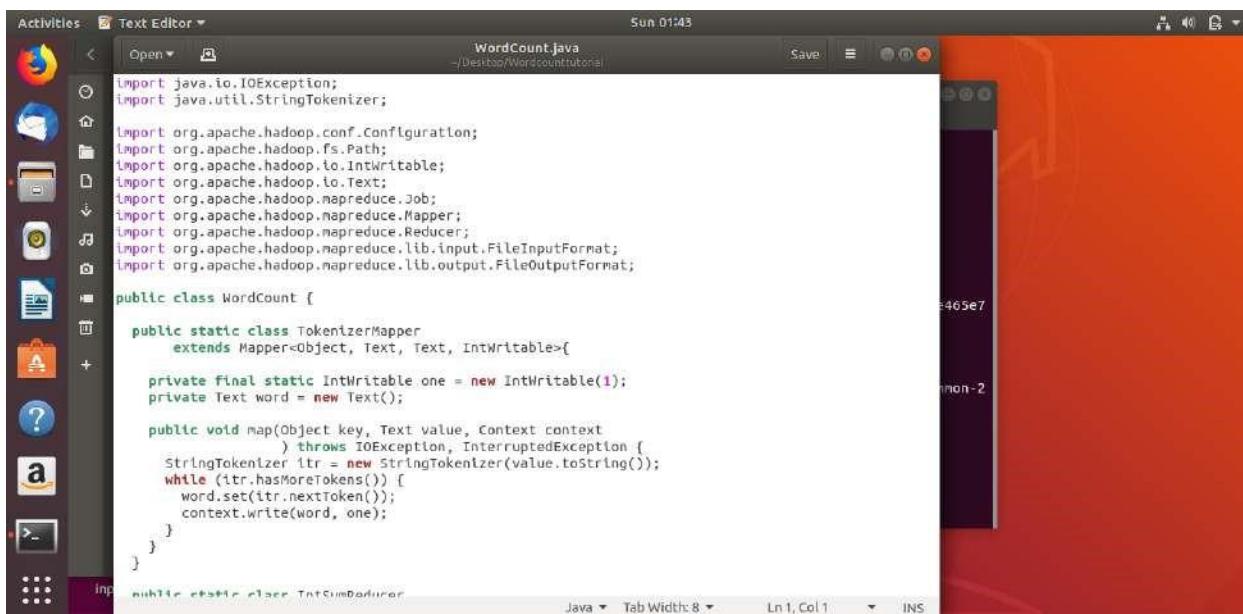
```

Procedure:

Step 1 – Verify Hadoop and Java Version



Step 2 – Create a Folder WordCountTutorial in desktop and save the WordCount.java in that folder



A screenshot of a Linux desktop environment. On the left is a vertical dock with icons for various applications like a browser, file manager, and terminal. The main window is a text editor titled "WordCount.java" with the path "/Desktop/WordcountTutorial". The code in the editor is:

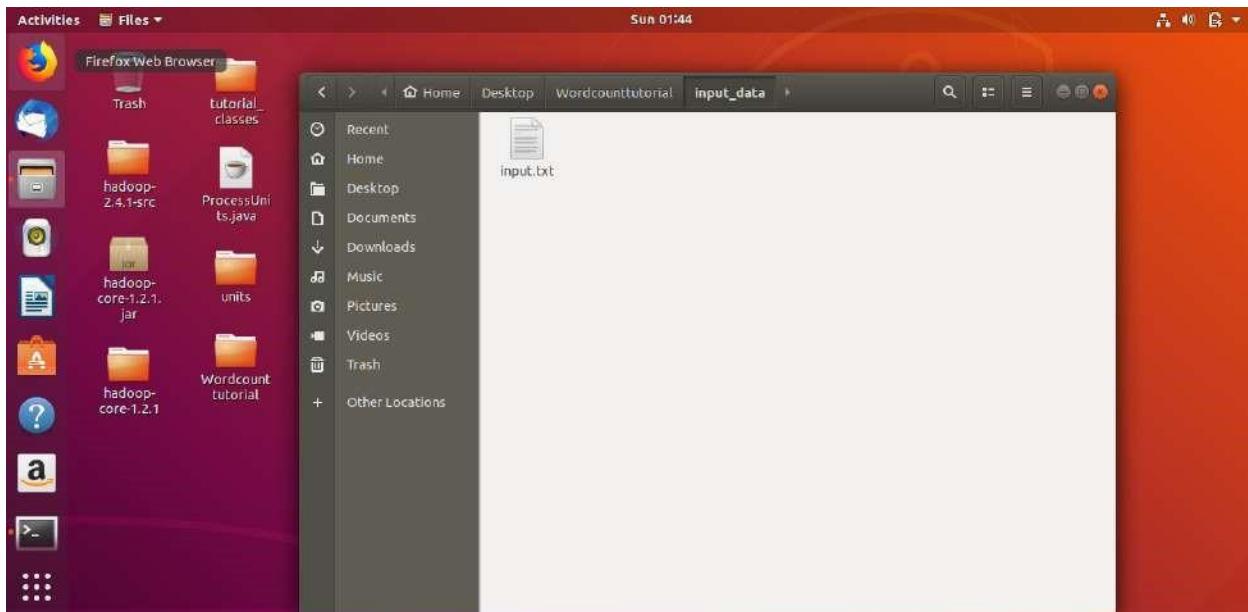
```
public static class IntSumReducer
    extends Reducer<Text,IntWritable,Text,IntWritable> {
private IntWritable result = new IntWritable();

public void reduce(Text key, Iterable<IntWritable> values,
                    Context context
                    ) throws IOException, InterruptedException {
    int sum = 0;
    for (IntWritable val : values) {
        sum += val.get();
    }
    result.set(sum);
    context.write(key, result);
}

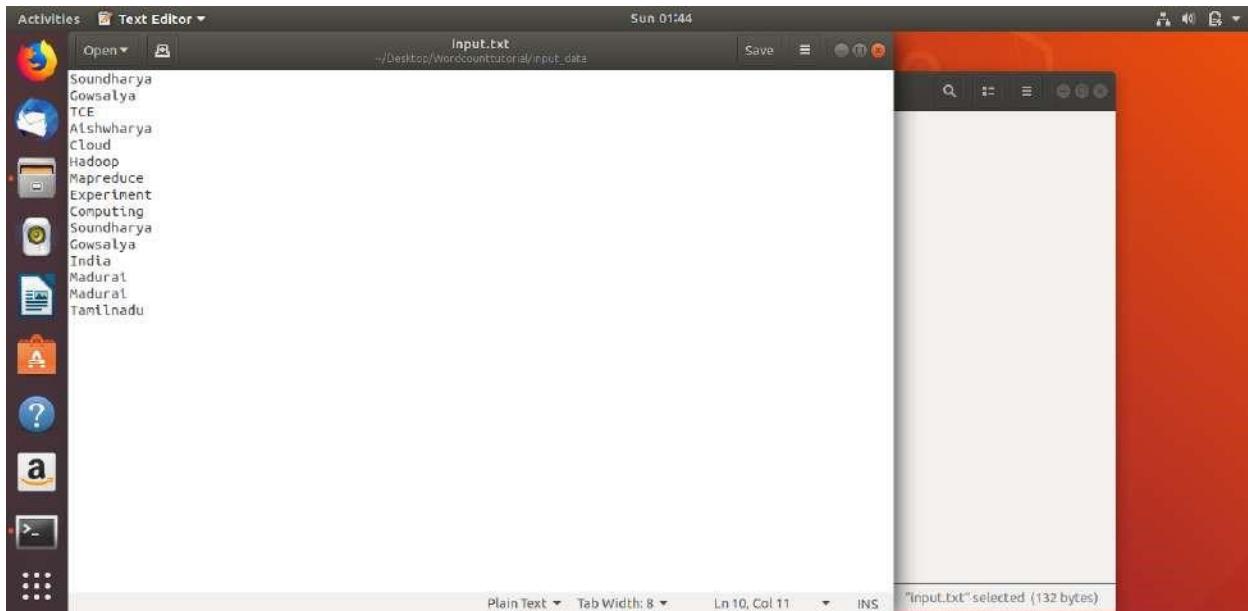
public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    fileInputFormat.addInputPath(job, new Path(args[0]));
    fileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
```

The terminal window to the right shows the output of the Java code, which includes the word "mon-2" and the value "465e7".

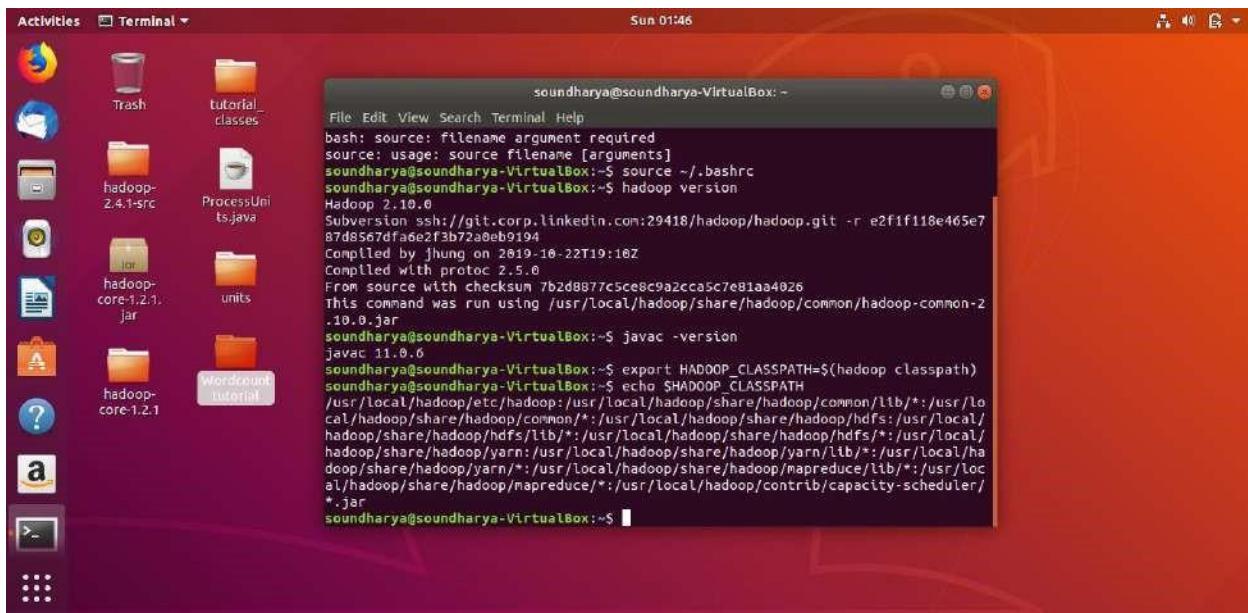
Step 3 – Create a folder InputData in WordCountTutorial and create a file input.txt



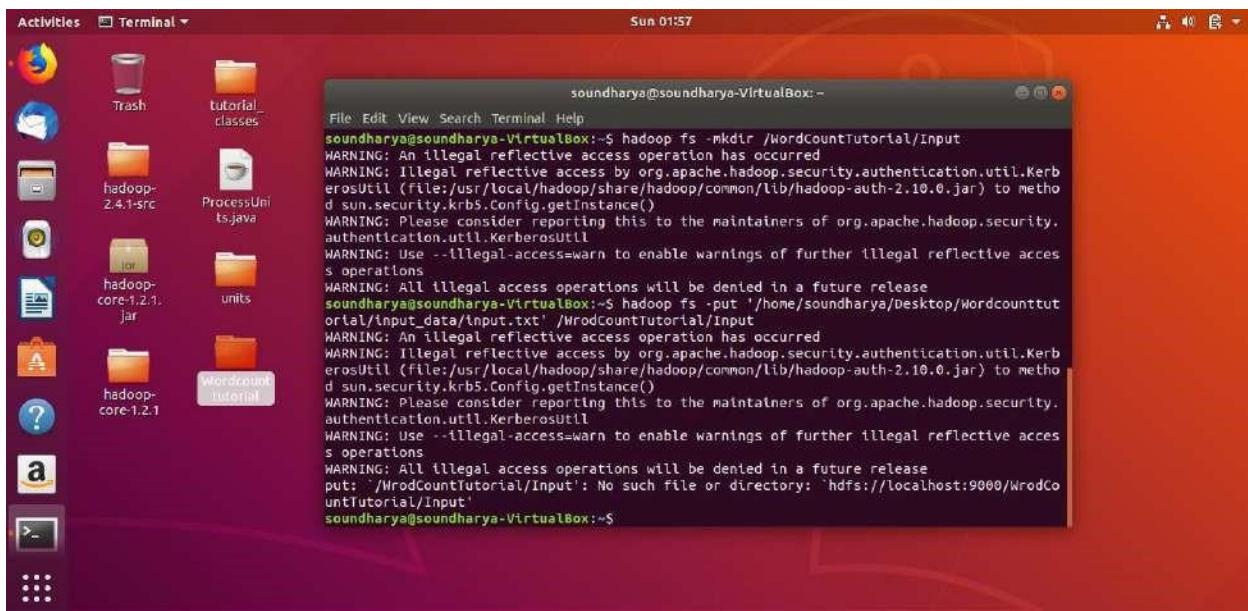
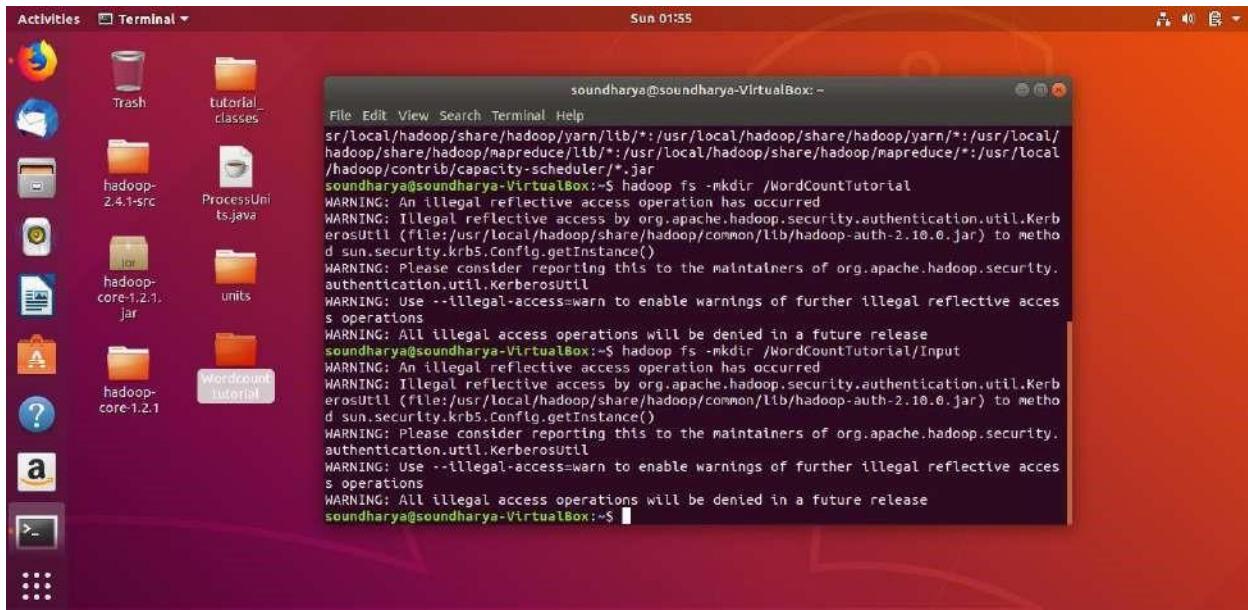
Write your own set of lines in that file



Step 4 – Export hadoop classpath



Step 5 : Create the same in Hadoop



Check it in browser localhost:50070

Sun 02:03

Namenode Information - Mozilla Firefox

localhost:50070/dfshealth.html#tab-overview

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Overview 'localhost:9000' (active)

Started:	Sat Apr 11 12:22:15 -0400 2020
Version:	2.10.0, re2f1f118e465e787d8567dfa6e2f3b72a0eb9194
Compiled:	Tue Oct 22 15:10:00 -0400 2019 by jhung from branch-2.10.0
Cluster ID:	CID-ed38bb2c-aedb-4586-9461-03ff570dce37
Block Pool ID:	BP-1807937076-127.0.1.1-1586622099458

Summary

Security is off.

Sun 02:04

Browsing HDFS - Mozilla Firefox

File Edit View Search Terminal Help

Hadoop Overview

Browse Directories

/

Show 25 entries

Permission drwxr-xr-x

Showing 1 to 1 of 1 entries

```
soundharya@soundharya-VirtualBox:~$ hadoop fs -mkdir /WordCountTutorial
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.0.jar) to method sun.security.krb5.Config.newInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
put: '/WordCountTutorial/Input': No such file or directory: 'hdfs://localhost:9000/WordCountTutorial/Input'
soundharya@soundharya-VirtualBox:~$ hadoop fs -mkdir /WordCountTutorial
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.0.jar) to method sun.security.krb5.Config.newInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
mkdir: '/WordCountTutorial': File exists
soundharya@soundharya-VirtualBox:~$
```

Hadoop, 2019.

Create a folder wordcounttutorial

Activities Terminal ▾ Sun 02:07 Browsing HDFS - Mozilla Firefox

SUBMITTED Applications X Hadoop-2.2.0 "It looks like it's working" X Browsing HDFS X +

localhost:50070/explorer.html#/ soundharya@soundharya-VirtualBox: ~

Hadoop Overview

Browse Directory

File Edit View Search Terminal Help

```
untTutorial/Input'
soundharya@soundharya-VirtualBox:~$ hadoop fs -mkdir /WordCountTutorial
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
mkdir: '/WordCountTutorial': File exists
soundharya@soundharya-VirtualBox:~$ hadoop fs -mkdir /WordCountTutorial/Input
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.0.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
mkdir: '/WordCountTutorial/Input': File exists
soundharya@soundharya-VirtualBox:~$
```

Show 25 entries

Permission drwxr-xr-x

Showing 1 to 1 of 1 entries

Hadoop, 2019.

Activities Firefox Web Browser ▾ Sun 02:09 Browsing HDFS - Mozilla Firefox

SUBMITTED Applications X Hadoop-2.2.0 "It looks like it's working" X Browsing HDFS X +

localhost:50070/explorer.html#/ soundharya@soundharya-VirtualBox: ~

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/

Show 25 entries

Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	soundharya	supergroup	0 B	Apr 12 01:48	0	0 B	WordCountTutorial

Showing 1 to 1 of 1 entries

Previous | Next

Hadoop, 2019.

Create a folder in wordcounttutorial

Browsing HDFS - Mozilla Firefox
localhost:50070/explorer.html#/WordCountTutorial

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/WordCountTutorial

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	soundharya	supergroup	0 B	Apr 12 01:48	0	0 B	Input

Showing 1 to 1 of 1 entries

Hadoop, 2019.

Create a file input.txt in that folder.

Browsing HDFS - Mozilla Firefox
localhost:50070/explorer.html#/WordCountTutorial/input

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

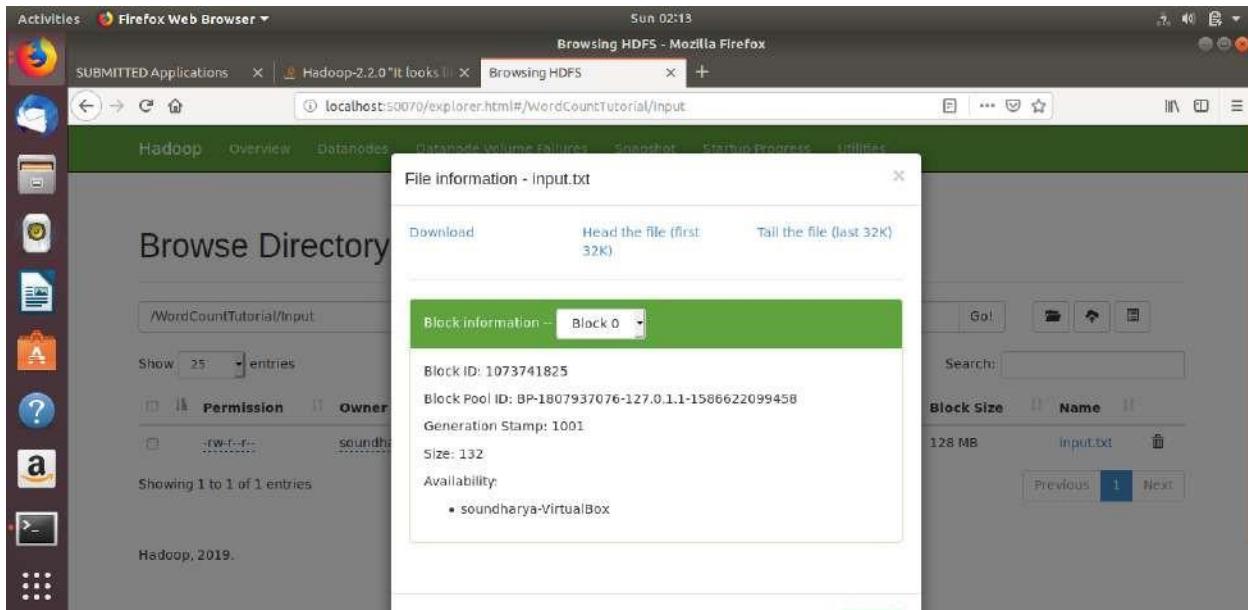
/WordCountTutorial/Input

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	soundharya	supergroup	132 B	Apr 12 02:12	1	128 MB	input.txt

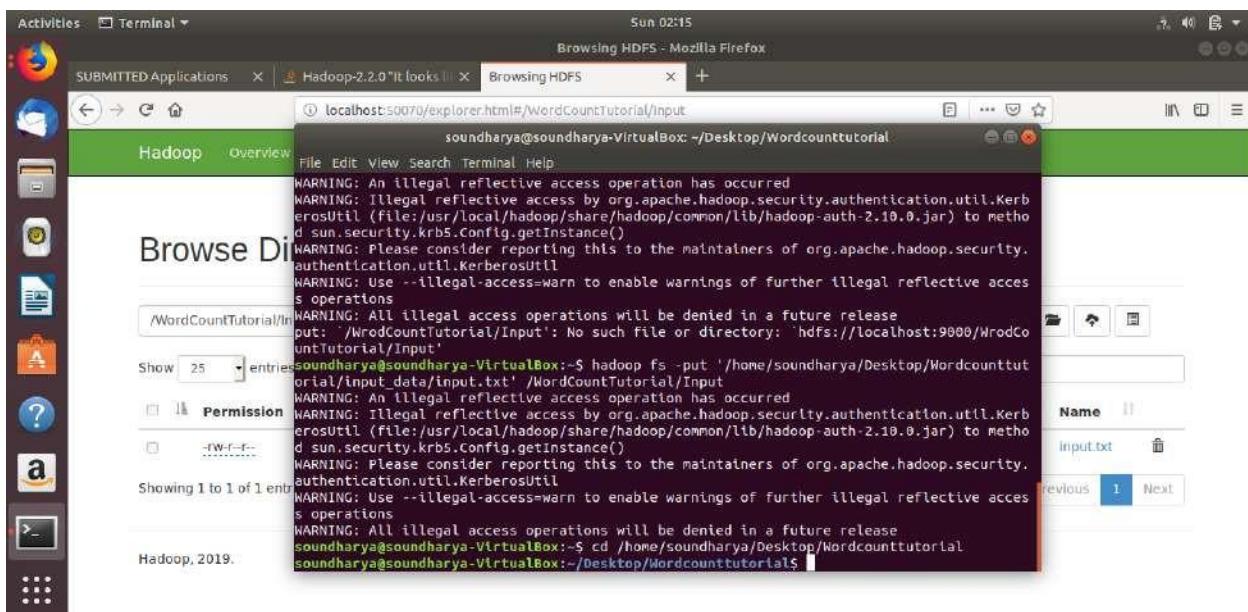
Showing 1 to 1 of 1 entries

Hadoop, 2019.

Check for that file



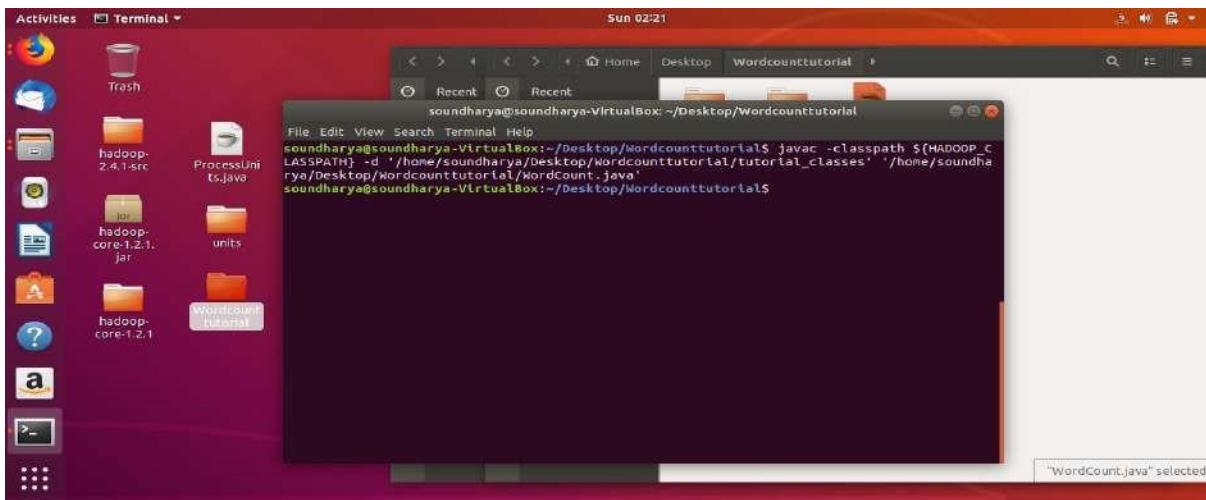
Go to the Wordcounttutorial folder path



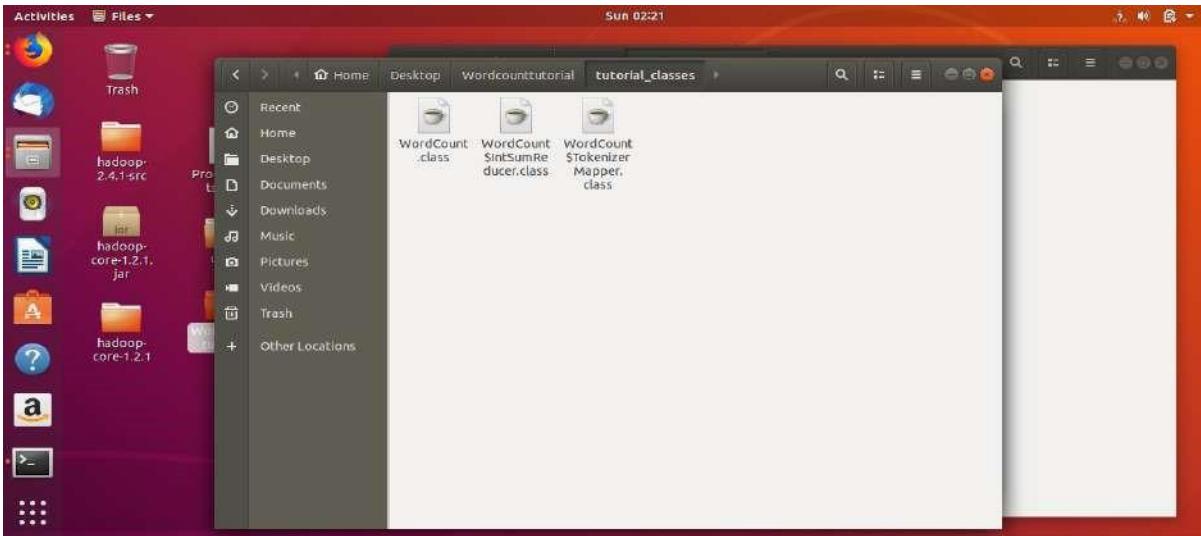
Step 6 : Create an empty folder tutorial classes



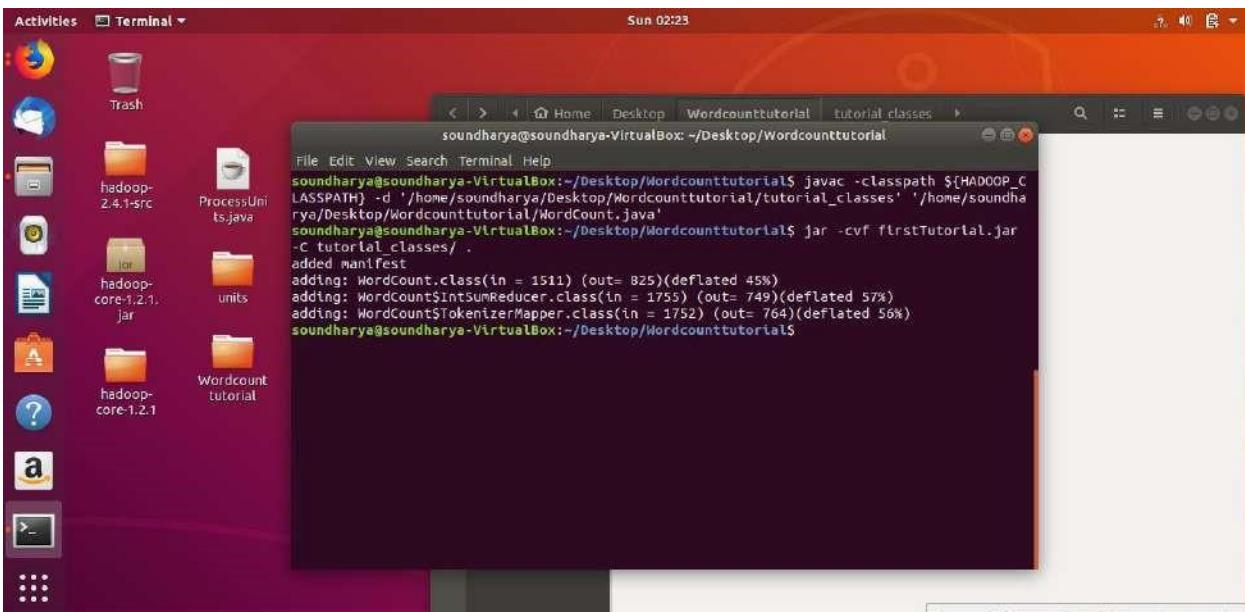
Compile the files using javac

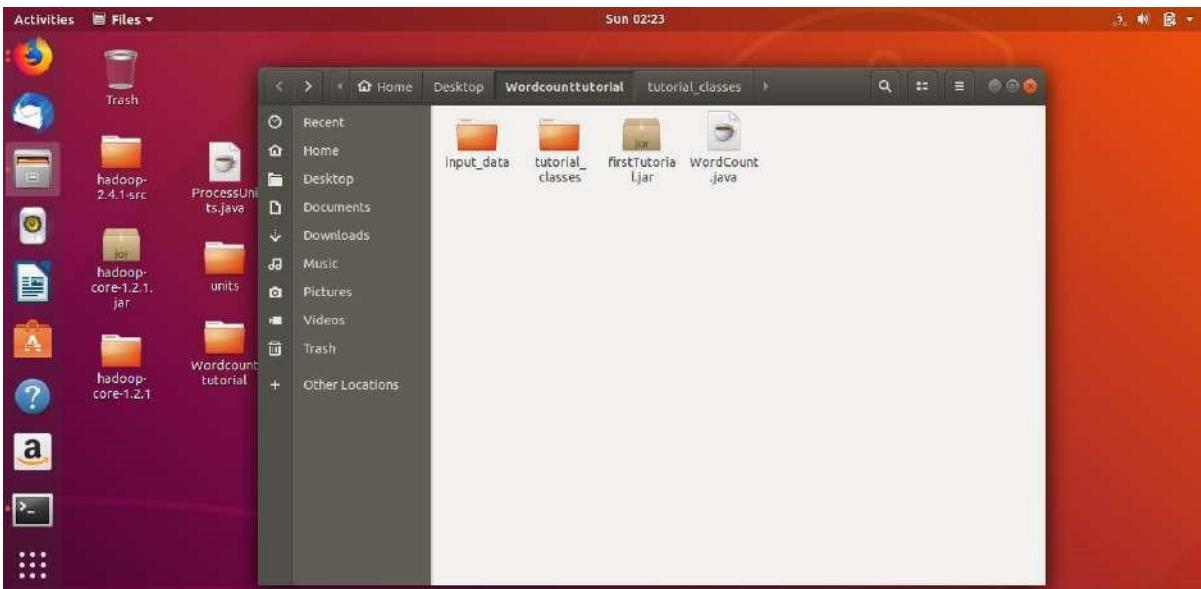


Now 3 new files created automatically in the empty folder tutorial_classes

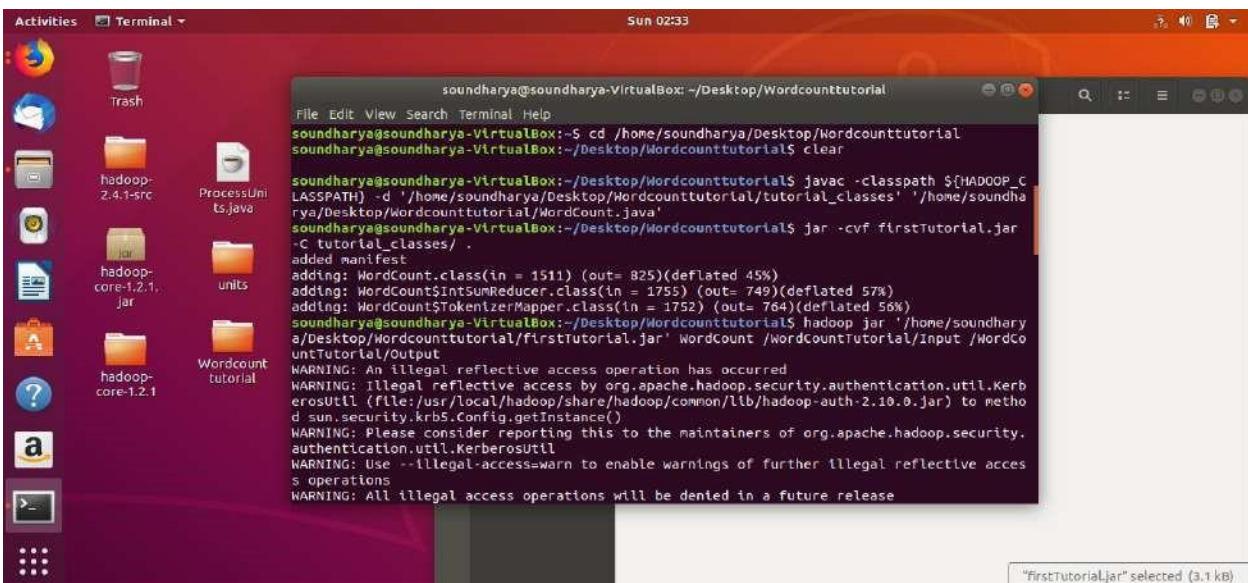


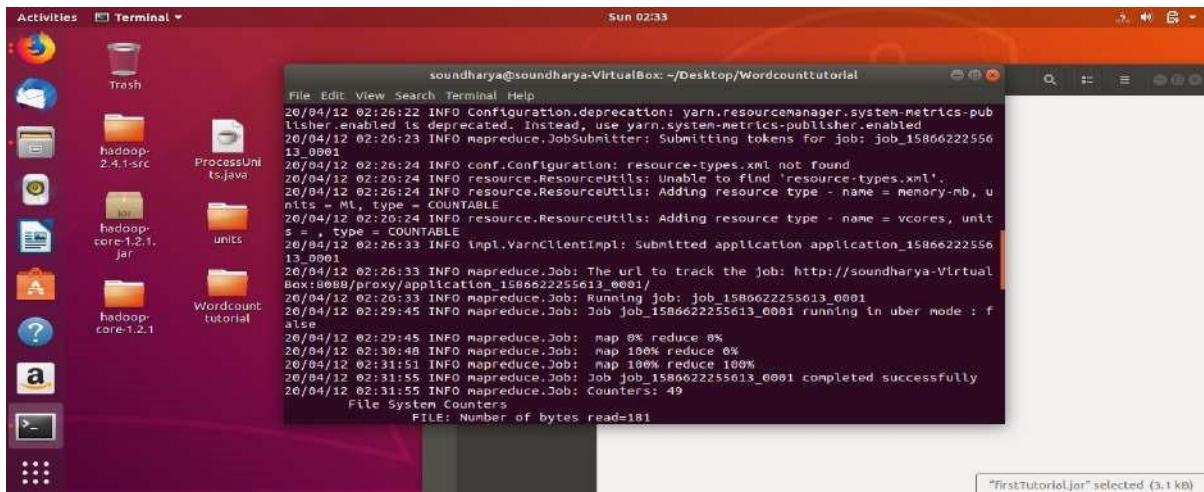
Create a jar file for it





Now use it in hadoop



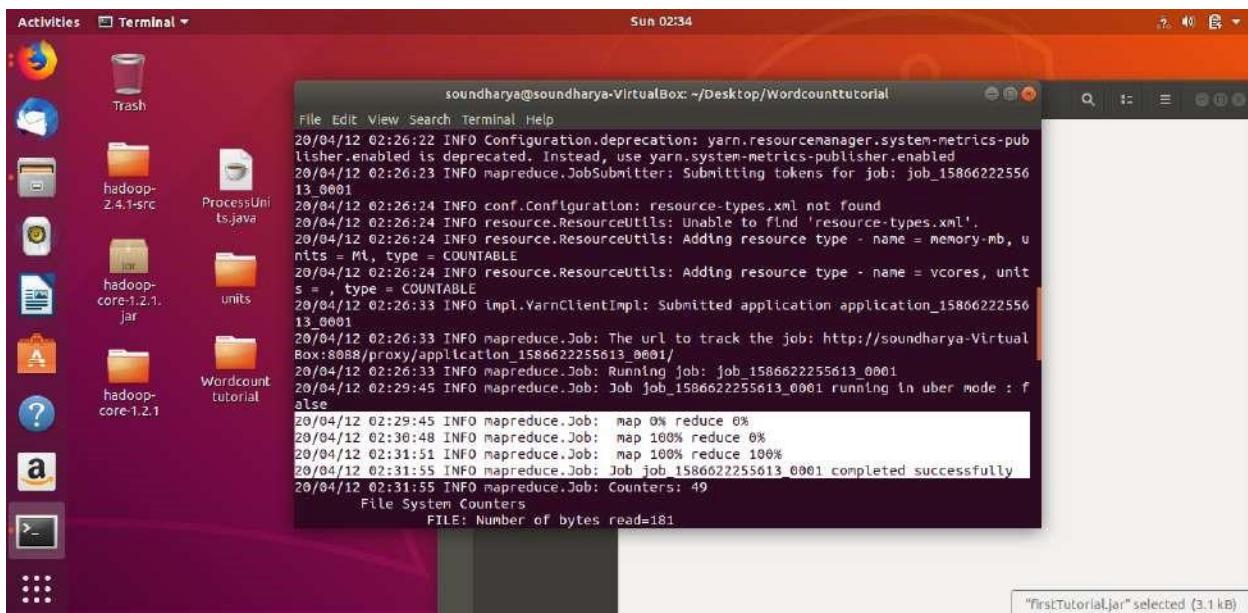


A screenshot of an Ubuntu desktop environment. On the left is a dock with icons for various applications like the Dash, Home, and Dash to Dock. In the center, a terminal window is open with the title "soundharya@soundharya-VirtualBox: ~/Desktop/Wordcountutorial". The terminal displays log output from a MapReduce job:

```
File Edit View Search Terminal Help
20/04/12 02:26:22 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-pub
lisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
20/04/12 02:26:23 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_15866222556
13_0001
20/04/12 02:26:24 INFO conf.Configuration: resource-types.xml not found
20/04/12 02:26:24 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
20/04/12 02:26:24 INFO resource.ResourceUtils: Adding resource type - name = memory-mb, u
nits = Ml, type = COUNTABLE
20/04/12 02:26:24 INFO resource.ResourceUtils: Adding resource type - name = vcores, unit
s = , type = COUNTABLE
20/04/12 02:26:33 INFO impl.YarnClientImpl: Submitted application application_15866222556
13_0001
20/04/12 02:26:33 INFO mapreduce.Job: The url to track the job: http://soundharya-Virtual
Box:8088/proxy/application_1586622255613_0001/
20/04/12 02:26:33 INFO mapreduce.Job: Running job: job_1586622255613_0001
20/04/12 02:29:45 INFO mapreduce.Job: Job job_1586622255613_0001 running in uber mode : f
alse
20/04/12 02:29:45 INFO mapreduce.Job: map 0% reduce 0%
20/04/12 02:30:48 INFO mapreduce.Job: map 100% reduce 0%
20/04/12 02:31:51 INFO mapreduce.Job: map 100% reduce 100%
20/04/12 02:31:55 INFO mapreduce.Job: Job job_1586622255613_0001 completed successfully
20/04/12 02:31:55 INFO mapreduce.Job: Counters: 49
File System Counters
FILE: Number of bytes read=181
```

The status bar at the bottom right shows "FirstTutorial.jar selected (3.1 kB)".

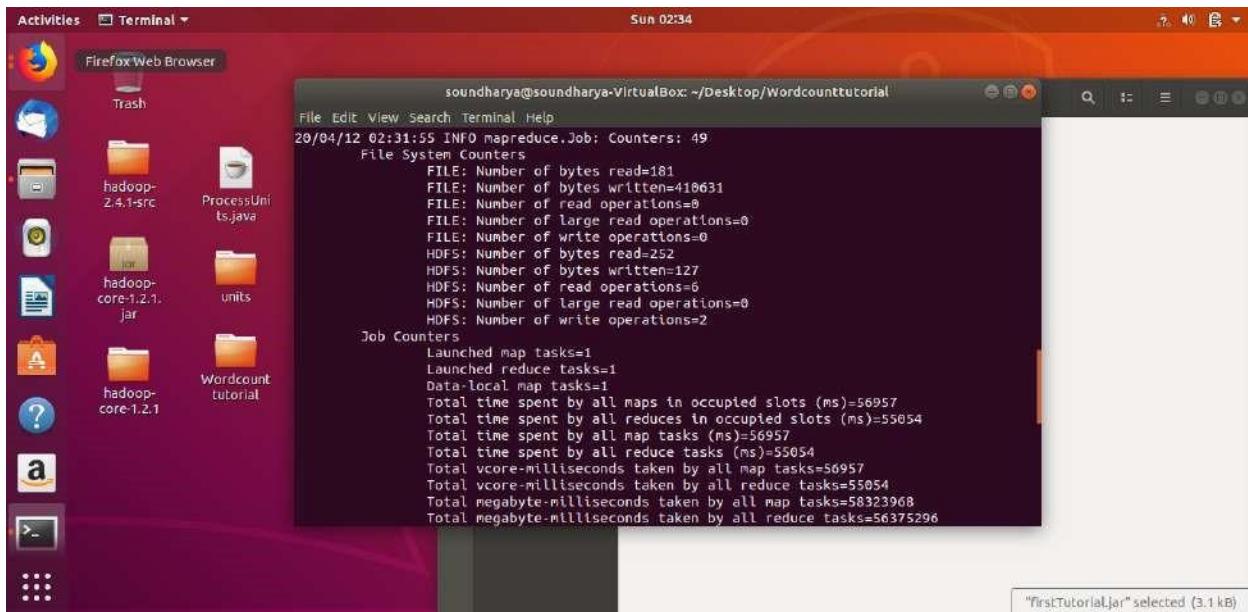
Mapreduce is running and being completed successfully



A second screenshot of the same Ubuntu desktop environment, showing the terminal window again. The log output is identical to the first one, indicating the job has completed successfully.

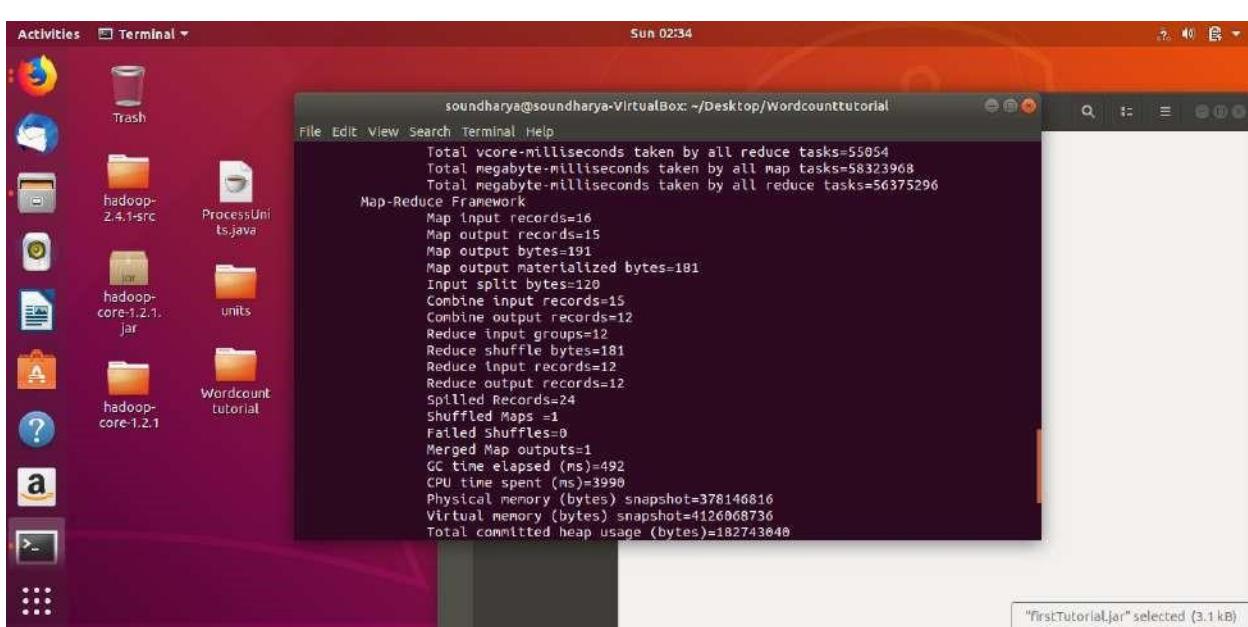
```
File Edit View Search Terminal Help
20/04/12 02:26:22 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-pub
lisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
20/04/12 02:26:23 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_15866222556
13_0001
20/04/12 02:26:24 INFO conf.Configuration: resource-types.xml not found
20/04/12 02:26:24 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
20/04/12 02:26:24 INFO resource.ResourceUtils: Adding resource type - name = memory-mb, u
nits = Ml, type = COUNTABLE
20/04/12 02:26:24 INFO resource.ResourceUtils: Adding resource type - name = vcores, unit
s = , type = COUNTABLE
20/04/12 02:26:33 INFO impl.YarnClientImpl: Submitted application application_15866222556
13_0001
20/04/12 02:26:33 INFO mapreduce.Job: The url to track the job: http://soundharya-Virtual
Box:8088/proxy/application_1586622255613_0001/
20/04/12 02:26:33 INFO mapreduce.Job: Running job: job_1586622255613_0001
20/04/12 02:29:45 INFO mapreduce.Job: Job job_1586622255613_0001 running in uber mode : f
alse
20/04/12 02:29:45 INFO mapreduce.Job: map 0% reduce 0%
20/04/12 02:30:48 INFO mapreduce.Job: map 100% reduce 0%
20/04/12 02:31:51 INFO mapreduce.Job: map 100% reduce 100%
20/04/12 02:31:55 INFO mapreduce.Job: Job job_1586622255613_0001 completed successfully
20/04/12 02:31:55 INFO mapreduce.Job: Counters: 49
File System Counters
FILE: Number of bytes read=181
```

The status bar at the bottom right shows "FirstTutorial.jar selected (3.1 kB)".



Output is being displayed

```
soundharya@soundharya-VirtualBox: ~/Desktop/Wordcounttutorial
File Edit View Search Terminal Help
20/04/12 02:31:55 INFO mapreduce.Job: Counters: 49
  File System Counters
    FILE: Number of bytes read=181
    FILE: Number of bytes written=410631
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
    HDFS: Number of bytes read=252
    HDFS: Number of bytes written=127
    HDFS: Number of read operations=6
    HDFS: Number of large read operations=0
    HDFS: Number of write operations=2
  Job Counters
    Launched map tasks=1
    Launched reduce tasks=1
    Data-local map tasks=1
    Total time spent by all maps in occupied slots (ms)=56957
    Total time spent by all reduces in occupied slots (ms)=55054
    Total time spent by all map tasks (ms)=56957
    Total time spent by all reduce tasks (ms)=55054
    Total vcore-milliseconds taken by all map tasks=56957
    Total vcore-milliseconds taken by all reduce tasks=55054
    Total megabyte-milliseconds taken by all map tasks=58323968
    Total Megabyte-milliseconds taken by all reduce tasks=56375296
```



```
soundharya@soundharya-VirtualBox: ~/Desktop/Wordcounttutorial
File Edit View Search Terminal Help
Total vcore-milliseconds taken by all reduce tasks=55054
Total megabyte-milliseconds taken by all map tasks=58323968
Total megabyte-milliseconds taken by all reduce tasks=56375296
Map-Reduce Framework
  Map input records=16
  Map output records=15
  Map output bytes=191
  Map output materialized bytes=181
  Input split bytes=120
  Combine input records=15
  Combine output records=12
  Reduce input groups=12
  Reduce shuffle bytes=181
  Reduce input records=12
  Reduce output records=12
  Spilled Records=24
  Shuffled Maps =1
  Failed Shuffles=0
  Merged Map outputs=1
  GC time elapsed (ms)=492
  CPU time spent (ms)=3990
  Physical memory (bytes) snapshot=378146816
  Virtual memory (bytes) snapshot=4126068736
  Total committed heap usage (bytes)=182743040
```

A screenshot of an Ubuntu desktop environment. On the left is a dock with icons for various applications like the Dash, Home, and Dash to Dock. In the center is a terminal window titled "soundharya@soundharya-VirtualBox: ~/Desktop/Wordcounttutorial". The terminal displays the output of a Hadoop word count job:

```
File Edit View Search Terminal Help
Reduce shuffle bytes=181
Reduce input records=12
Reduce output records=12
Spilled Records=24
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=492
CPU time spent (ms)=3998
Physical memory (bytes) snapshot=378146816
Virtual memory (bytes) snapshot=4126068736
Total committed heap usage (bytes)=182743040
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=132
File Output Format Counters
Bytes Written=127
soundharya@soundharya-VirtualBox:~/Desktop/Wordcounttutorial$
```

Now the program has been executed and the word count of the input.txt is displayed

A screenshot of an Ubuntu desktop environment. On the left is a dock with icons for various applications like the Dash, Home, and Dash to Dock. In the center is a terminal window titled "soundharya@soundharya-VirtualBox: ~/Desktop/Wordcounttutorial". The terminal displays the results of the Hadoop word count job, showing the word counts for each word in the input file:

```
File Edit View Search Terminal Help
Bytes Read=132
File Output Format Counters
Bytes Written=127
soundharya@soundharya-VirtualBox:~/Desktop/Wordcounttutorial$ hadoop dfs -cat /WordCountTutorial/Output/*
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.10.0.jar) to method sun.security.krb5.Config.getinstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Aishwarya 1
Cloud 1
Computing 1
Experiment 1
Gowsalya 2
Hadoop 1
India 1
Madurai 2
Mapreduce 1
Soundharya 2
TCE 1
Tamilnadu 1
soundharya@soundharya-VirtualBox:~/Desktop/Wordcounttutorial$
```

Result:

Thus the hadoop mapreduce has been done and the program for wordcount has been executed successfully.

Ex.No : 13	
08.05.2021	

Study on LibVirt

Aim:

To study on LibVirt virtualization on Linux.

Description:

Libvirt is an open-source API, daemon and management tool for managing platform virtualization. It can be used to manage KVM, Xen, VMware ESXi, QEMU and other virtualization technologies. These APIs are widely used in the orchestration layer of hypervisors in the development of a cloud-based solution.

Contribution Required:

The libvirt project is always looking for new contributors to participate in ongoing activities. While code development is a major part of the project, assistance is needed in many other areas including documentation writing, bug triage, testing, application integration, website / wiki content management, translation, branding, social media and more. The only requirement is an interest in virtualization and desire to help.

The following is a non-exhaustive list of areas in which people can contribute to libvirt. If you have ideas for other contributions feel free to follow them.

- Software development. The core library / daemon (and thus the bulk of coding) is written in C, but there are language bindings written in Python, Perl, Java, Ruby, Php, OCaml and Go. There are also higher level wrappers mapping libvirt into other object frameworks, such GLib, CIM and SNMP. For those interested in working on the core parts of libvirt, the contributor guidelines are mandatory reading
- Translation. All the libvirt modules aim to support translations where appropriate. All translation is handled outside of the normal libvirt review process, using the Fedora instance of the Weblate tool. Thus people wishing to contribute to translation should join the Fedora translation team

- Documentation. There are docbook guides on various aspects of libvirt, particularly application development guides for the C library and Python, and a virsh command reference. There is also a need for people to review existing content for copy editing and identifying gaps in the docs
- Website / wiki curation. The bulk of the website is maintained in the primary GIT repository, while the wiki site uses mediawiki. In both cases there is a need for people to both write new content and curate existing content to identify outdated information, improve its organization and target gaps.
- Testing. There are a number of tests suites that can run automated tests against libvirt. The coverage of the tests is never complete, so there is a need for people to create new test suites and / or provide environments to actually run the tests in a variety of deployment scenarios.
- Code analysis. The libvirt project has access to the coverity tool to run static analysis against the codebase, however, there are other types of code analysis that can be useful. In particular fuzzing of the inputs can be very effective at identifying problematic edge cases.
- Security handling. Downstream (operating system) vendors who distribute libvirt may wish to propose a person to be part of the security handling team, to get early access to information about forthcoming vulnerability fixes.
- Evangelism. Work done by the project is of no benefit unless the (potential) user community knows that it exists. This can take many forms, writing blog posts (about usage of features, personal user experiences, areas for future work, and more), syndicating docs and blogs via social media, giving user group and/or conference talks about libvirt.
- User assistance. Since documentation is never perfect, there are inevitably cases where users will struggle to attain a deployment goal they have, or run into trouble with managing an existing deployment. While some users may be able to contact a software vendor to obtain support, it is common to rely on community help forums such as libvirt users mailing list, or sites.

Supported Hypervisor:

Libvirt is a C library with bindings in other languages, notably in Python, Perl, OCaml, Ruby, Java, JavaScript (via Node.js) and PHP. libvirt for these programming languages is composed of wrappers around another class/package called libvirtmod. libvirtmod's implementation is closely associated with its counterpart in C/C++ in syntax and functionality.

- LXC – lightweight Linux container system
- OpenVZ – lightweight Linux container system
- Kernel-based Virtual Machine/QEMU (KVM) – open-source hypervisor for Linux and SmartOS[11]
- Xen – bare-metal hypervisor
- User-mode Linux (UML) – paravirtualized kernel
- VirtualBox – hypervisor by Oracle (formerly by Sun) for Windows, Linux, macOS, and Solaris
- VMware ESXi and GSX – hypervisors for Intel hardware
- VMware Workstation and Player – hypervisors for Windows and Linux
- Hyper-V – hypervisor for Windows by Microsoft
- PowerVM – hypervisor by IBM for AIX, Linux and IBM i
- Bhyve – hypervisor for FreeBSD 10+[12] (support added with libvirt 1.2.2)

User Interfaces:

Various virtualization programs and platforms use libvirt. Virtual Machine Manager, GNOME Boxes and others provide graphical interfaces. The most popular command line interface is virsh, and higher level tools such as oVirt.

XML Configuration:

Objects in the libvirt API are configured using XML documents to allow for ease of extension in future releases. Each XML document has an associated Relax-NG schema that can be used to validate documents prior to usage.

- Domains
- Networks
- Network filtering
- Network ports
- Storage
- Storage encryption
- Capabilities
- Domain capabilities
- Storage Pool capabilities
- Node devices

Features:

- It is a toolkit to manage virtualization platforms
- It is accessible from C, Python, Perl, Go and more
- It is licensed under open source licenses
- It supports KVM, QEMU, Xen, Virtuozzo, VMWare ESX, LXC, BHyve and more • It targets Linux, FreeBSD, Windows and macOS • It is used by many applications.

Result:

Hence the study on LibVirt virtualization on Linux are provided.