

Aim:

To Upload, Share and download user files in cloud environments like Google Drive.

Concept:

“Cloud” is short for “cloud computing,” and it refers to tasks and services provided or hosted via the internet on a pay-as-we-go basis.

The cloud is a collection of servers and data centers scattered across the globe that store data. Essentially, it's a digital storage unit where we can keep all our files. With the cloud, we can access our data from any device so long as it has an internet connection.

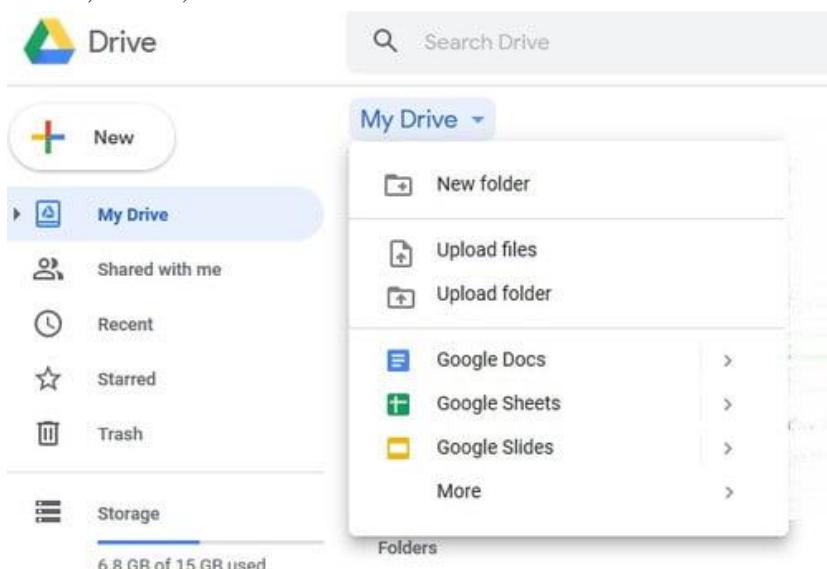
Process /Procedure:*Google Drive*

1. Go to drive.google.com
2. Upload or create files to google drive
3. Share and organize files or folders, so other people can view, edit, or comment on them.

Experiment:**Step 1:** Go to drive.google.com

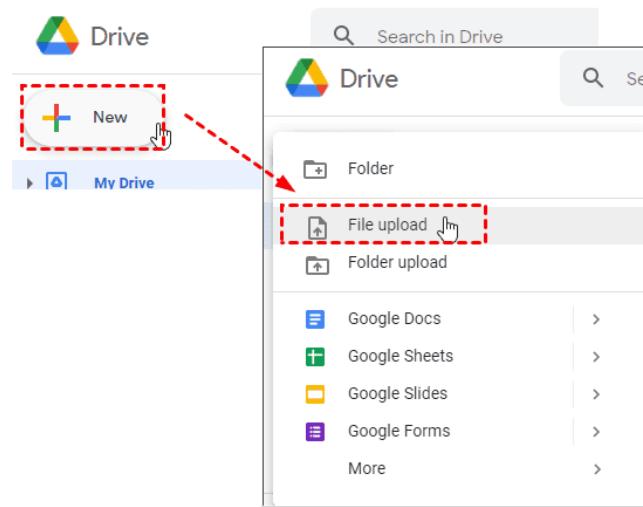
On our computer, go to drive.google.com. We'll see "My Drive," which has:

- Files and folders we upload or sync
- Google Docs, Sheets, Slides, and Forms we create

**Learn how to back up and sync files from our Mac or PC****Step 2:** Upload or create files

We can upload files from our computer or create files in Google Drive.

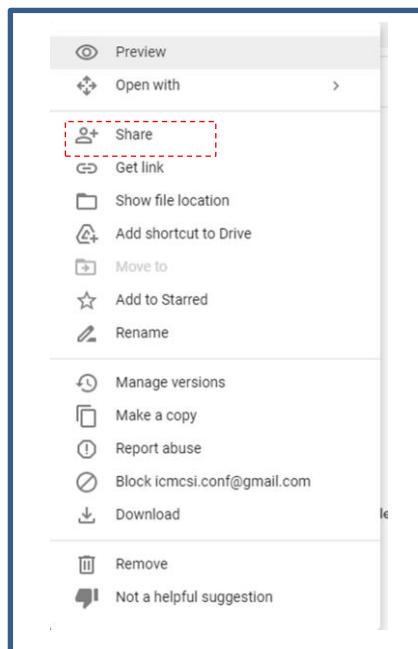
- Upload files and folders to Google Drive
- Work with Office files
- Create, edit, and format Google Docs, Sheets, and Slides



Step 3: Share and organize files

We can share files or folders, so other people can view, edit, or comment on them.

- Share files from Google Drive
- Share folders from Google Drive
- Make someone else the owner of a file



Result:

The above steps are used for uploading, sharing and downloading files using google drive cloud environments.

Aim:

To Upload, Share and download user files in cloud environments like Dropbox.

Concept:

“Cloud” is short for “cloud computing,” and it refers to tasks and services provided or hosted via the internet on a pay-as-we-go basis.

In Dropbox, any business initiative, succeeding in sales enablement means crafting a solid strategy and executing on it. Much has been written on sales enablement strategy, and the field is constantly evolving as technology, buyers’ habits, and other factors change over time.

Process/Procedure:*Dropbox*

1. Create DropBox Account by filling required details
2. Upload Files - Navigate to “Upload Files/Folder” , Select the File , Select the file we want to upload and tap “open.”
3. Share a File or Folder - Locate the File in Our Dropbox Folder, Generate a Share Link , Copy the Link
4. Download Files - Locate the File , Select the File and Download

Experiment

Step 1: To begin using Dropbox, we’ll first need to create an account. Thankfully, it only takes a couple of minutes to do, and we can follow the steps below to get started.

1. Locate the “Sign in” Prompt on the Website

Once we have reached the Dropbox website, select “sign in” in the top right-hand corner.

The image shows the Dropbox login page. At the top, there are two buttons: "Log in" on the left and "Sign up for free" on the right. Below these are two social login buttons: "Log in with Google" (with a Google logo) and "Log in with Apple" (with an Apple logo). A horizontal line with the word "or" in the center separates these from the manual login fields. There is a "Email" input field, a "Password" input field, and a "Remember me" checkbox followed by a "Forgot password?" link. At the bottom is a large blue "Log in" button.

2. Select “Create an Account”

To create a new account, select the “create an account” option.

Welcome to Dropbox!

Bring your photos, docs and videos anywhere. [Take a tour.](#)

I agree to [Dropbox Terms](#)

[Create account](#)



3. Enter Our Personal Details

We'll be prompted to **enter our name, email address and password**. (If we want to keep our passwords secure, check out our pick of the best password managers available.) For a slightly quicker process, we can also sign up to Dropbox using our Google account.

4. Select Our Plan

On the next page, we will be asked to **select the plan** we would like to use. If we prefer to stick to the free plan, select the option “or continue with 2GB Dropbox Basic plan” located at the bottom of the screen.

Dropbox will send us an email confirmation of our account. We will need to verify our email address before we can use the service.

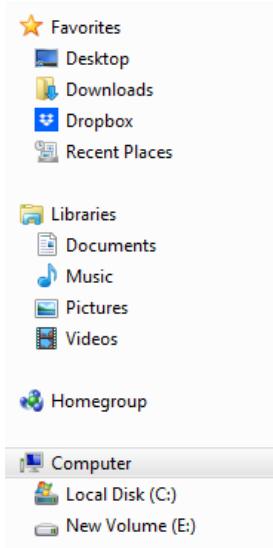
Once the sign-up process is complete, we will be prompted to download Dropbox to our computer. It's a good idea to do this if we want to automatically sync files from our computer to the service.

Use the Dropbox Folder on Desktop:

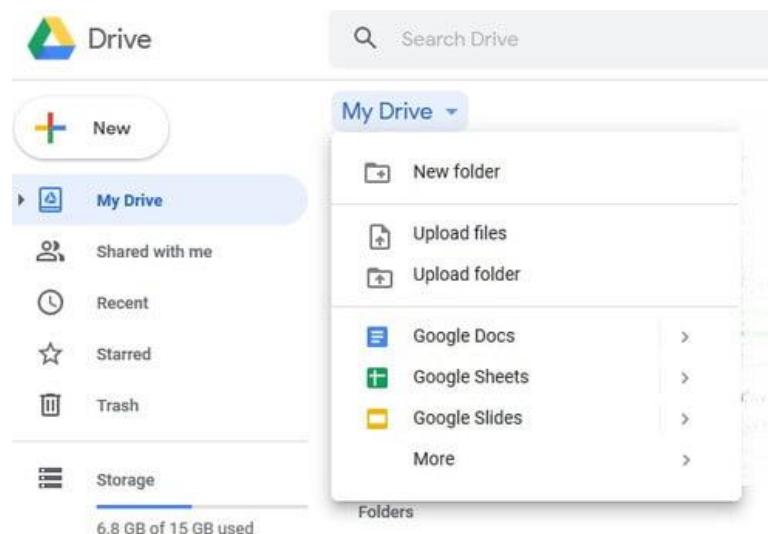
Step 2: Downloading Dropbox to our desktop automatically creates a Dropbox folder. We can use the steps below to sync files to this folder.

1. Locate the Dropbox Folder

Open “**finder**” and locate “**Dropbox**” in our “**favorites**” tray.



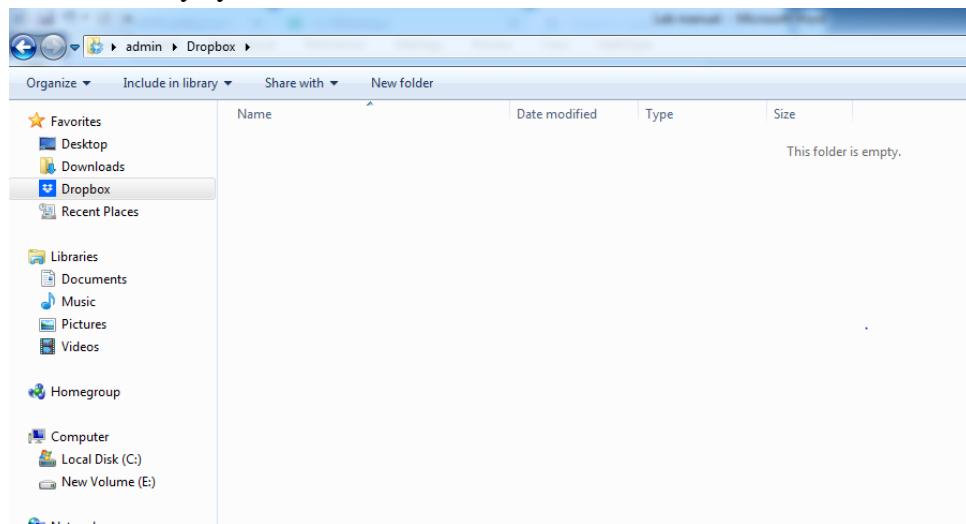
2. Drag and Drop Files Into Our Folder



To add files, simply **drag and drop** them into our Dropbox folder. All files added will be synced to our Dropbox account and we will be able to access them across multiple devices.

3. Save Files to Our Dropbox Folder

When saving a file, we can choose to save it directly in our Dropbox folder. All files saved to Dropbox will automatically sync to our account.



Use Dropbox on Web

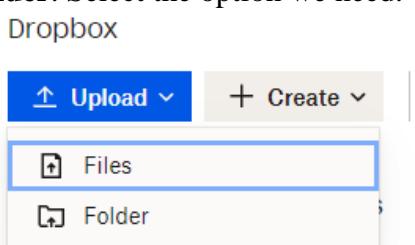
Step 3: If we prefer not to download Dropbox to our computer, we can upload files directly through the web application. It's easy to do; just follow the instructions below to begin backing up our files.

Uploading Files

The first thing we'll want to do with Dropbox is upload some files.

1. Navigate to “Upload Files/Folder”

Log in to our Dropbox account via our web browser. On the right-hand side of the home screen, we'll see the option to **upload a file or folder**. Select the option we need.



2. Select the File

Select the file we want to upload and tap “open.” The file will be uploaded to our Dropbox account. We can follow the same process to upload a folder.

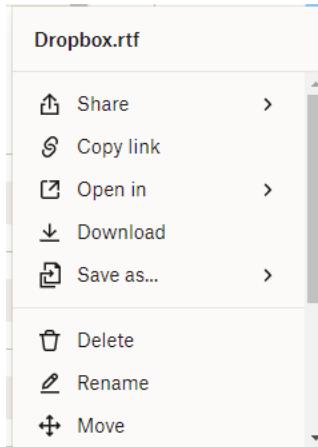
The screenshot shows the Dropbox web interface. At the top, there are buttons for 'Upload', 'Create', and 'More'. Below that is a table with three columns: 'Name', 'Modified', and 'Who can access'. A single file, 'Dropbox.rtf', is listed. It was modified on 11/3/2022 at 11:53 am and has 'Only you' as the access level. There is also a small star icon next to the file name.

Navigate the Files Using Dropbox

Step 4: Finding files on the web application is straightforward and takes no time at all.

1. Select All Files

On the right-hand side of the home screen, we will find six options. To find a file or folder, tap “all files.” We can then browse this section to find the files we want.



2. Or Use the Search Bar

Alternatively, if we know the name of the file or folder we want to locate, we can type it into the search bar.

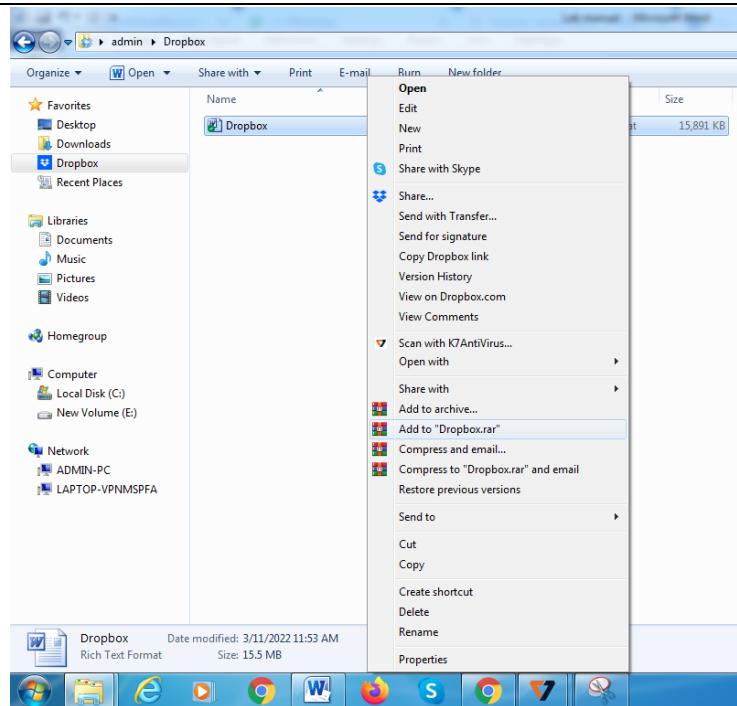


Use Dropbox to Share a File or Folder

There are multiple ways to share a file or folder through Dropbox.

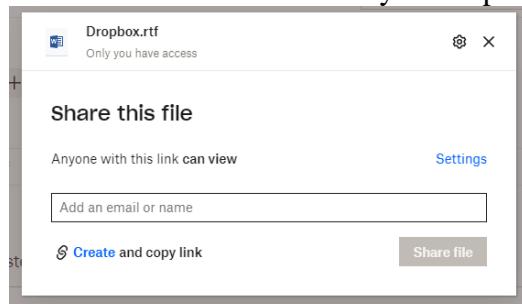
Locate the File in our Dropbox Folder

Step 5: Go to the **Dropbox folder on our desktop**. Search for the file we would like to share and right-click our mouse. Find “share” in the dropdown menu and select it.



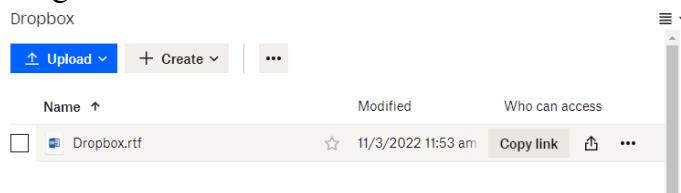
1. Email the File

To email the file access directly, **add the contact information** for the recipient in the “to:” section. Next, **click “share”** and access to the file will be sent directly to the person we selected.



2. Generate a Share Link

Instead of emailing our file, we can generate a shareable link. After selecting “**share**,” we will see “**create link**” at the bottom right-hand side of the window.



3. Copy the Link

Next, copy the link and share it with anyone we would like to access our files.

Use Dropbox to Download Files

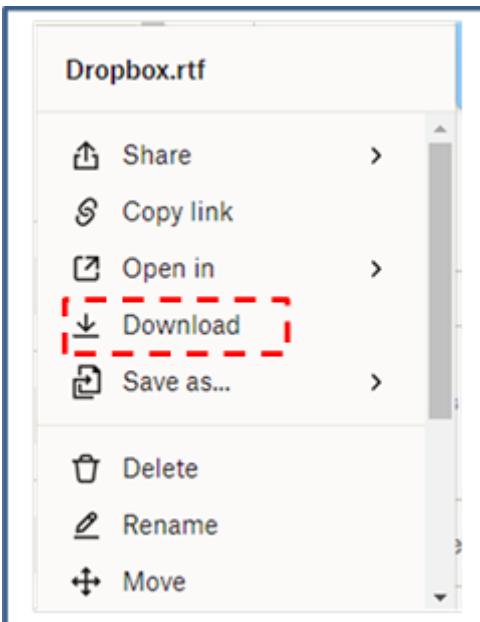
Step 6: Let’s say that we’re on a new device and we need access to a file we saved to Dropbox, but we don’t have the service installed and don’t want to install it. Instead, our option is to log in via our browser and navigate to the file we want to download.

1. Locate the File

To download a file onto our computer from the web application, first **locate the file** using the search function as outlined above.

2. Select the File and Download

Select the file and **tap the ellipsis**. From the dropdown menu, select “**download**.”

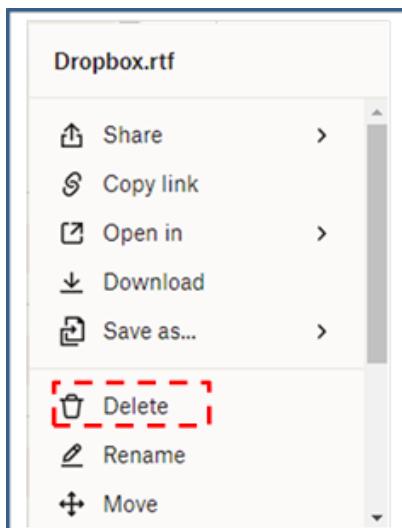


Use Dropbox to Restore Files

Step 7: If we accidentally delete a file or folder, fear not: we have 30 days to recover it (120 days on Professional plans). Here's what we can do to recover our files.

1. Locate Deleted Files

On the left-hand side of the home screen in the web application, we will find “**deleted files**.” Select it to move forward.



2. Select the File We Want to Restore

Select the file we would like to restore and press “**restore**” on the right-hand side of the screen. If we would rather delete the file completely, select “**delete permanently**,” located underneath the “**restore**” button.

Use Dropbox to Request Files

Step 8: File requests allow inviting people who don’t have a Dropbox account to upload files to ours. We may find that useful when we need to get files from employees or relatives who don’t use the platform. It’s also a good way for teachers to collect files from students.

1. Log in and Locate “File Requests”

To request a file, **log in** to our online Dropbox account. On the left-hand side, we’ll see “**file requests**.” Click on it to move to the next page.

The screenshot shows the 'Deleted files' section of the Dropbox web interface. On the left, a sidebar lists various categories like Home, All files, Recents, Starred, Photos, Shared, File requests, Automations, and Deleted files. The 'Deleted files' link is highlighted with a red dashed box. The main area displays a list of deleted files with their names, deletion times, and preview icons. To the right, there are filters for 'From date', 'To date', 'Deleted by', 'Email or name', 'In folder', and a 'Reset filters' button.

2. Select “New Request”

Next, select “new request” on the right-hand side of the screen.

3. Name the Folder and Add a Description

On the next page, we’ll be asked to **name the files** we’re requesting. For example, we could call them “birthday photos” or “tax documents.” Once we’ve decided on a name, choose the folder we want Dropbox to put them in. Tap “create” to move forward.

The screenshot shows the 'Create new request' dialog box. The sidebar on the left shows the 'File requests' tab is selected. The main area contains a form with fields for 'Title' (placeholder: 'Explain what the request is for'), 'Description (optional)' (placeholder: 'Add any extra details about the request'), 'Folder for uploaded files' (set to 'File requests - Dropbox' with a 'Change Folder' button), and a 'Set a deadline' checkbox. At the bottom are 'Cancel' and 'Create' buttons. The 'File requests' link in the sidebar is highlighted with a red dashed box.

4. Add Contacts to the Shared Folder

On the next page, we can add the contacts we would like to have access to the folder. We can either send the request via email, or copy and paste the shareable link and manually send it to them.

Result:

The above steps are used for uploading, sharing and downloading files using dropbox cloud environments.

Aim:

To Create a Warehouse application in salesforce.com cloud environment.

Concept:

Salesforce is a cloud computing Service as a Software (SaaS) company that specializes in Customer Relationship Management (CRM). Salesforce's services allow businesses to use cloud technology to better connect with customers, partners and potential customers.

Process/Procedure:

1. Go to <https://developer.salesforce.com/signup> and login with credentials
2. Click Setup, right top side of the screen.
3. From menu, click Create > Apps.
4. Click new button to create a new App.
5. Select the type of an App and click Next button
6. Specify a label for an App.
7. Click the Add and Remove arrow buttons to add or remove tabs from the App.
8. Add required Custom objects related to the App.

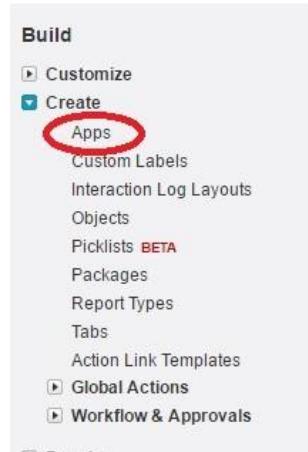
Experiment:**Create an application with the name Warehouse as below:**

Go to <https://developer.salesforce.com/signup> and login with our credentials

Step 1: Click Setup, right top side in our screen.



Step 2: Once we're in the Setup area, we'll see the menu on the left side of the page. From that menu, click Create > Apps.



Step 3: Click new button to create a new App.

Action	App Label	Console	Custom	Description
Edit	App Launcher	<input type="checkbox"/>	<input type="checkbox"/>	App Launcher tabs
Edit	Call Center	<input type="checkbox"/>	<input type="checkbox"/>	State-of-the-Art On-Demand Customer Service
Edit	Community	<input type="checkbox"/>	<input type="checkbox"/>	Salesforce CRM Communities
Edit	Content	<input type="checkbox"/>	<input type="checkbox"/>	Salesforce CRM Content
Edit	Marketing	<input type="checkbox"/>	<input type="checkbox"/>	Best-in-class on-demand marketing automation
Edit	Platform	<input type="checkbox"/>	<input type="checkbox"/>	The fundamental Force.com platform
Edit	Sales	<input type="checkbox"/>	<input type="checkbox"/>	The world's most popular sales force automation (SFA) solution
Edit	Salesforce Chatter	<input type="checkbox"/>	<input type="checkbox"/>	The Salesforce Chatter social network, including profiles and feeds
Edit	Sample Console	<input checked="" type="checkbox"/>	<input type="checkbox"/>	The out-of-the box console for users who work with multiple records on one screen
Edit	Site.com	<input type="checkbox"/>	<input type="checkbox"/>	Build pixel-perfect, data-rich websites using the drag-and-drop Site.com application, and manage content and published sites.
Edit Del	Warehouse	<input type="checkbox"/>	<input checked="" type="checkbox"/>	this is an App for storage of goods for sell.

Step 4: Select the type of an App; here we are selecting a custom app and click Next button

Step 1. Select Type Step 1 of 5

Select the type of app to create.

Custom app
 Console

Next **Cancel**

Step 5: Specify a label for an App. The label can have a max of 40 characters, including spaces. This label is an App's name in the Force.com App tab/menu. Optionally, enter a description of an App and Click Next.

Step 2. Enter the Details Step 2 of 5

Fill in the fields below to define the custom app.

Custom App Information | = Required Information

App Label Example: HRforce, Financeforce, Bugforce
 App Name i
 Description
Previous Next Cancel

Step 6: Optionally, specify a custom logo for an App. Click Insert an image to choose an image file from the document library and Click Next.

Step 3. Choose the Image Source for the Custom App Logo Step 3 of 5

Optionally, specify a logo for this custom app. To do so, choose an image file from the document library.

(i) The file size of a custom app logo must be smaller than 20 KB. (For comparison, the salesforce.com logo is about 3 KB). To upload an image file, add a new document to the Documents tab. Image dimensions should be a maximum of 300 pixels wide by 55 pixels high for best results. Larger images will be resized and may appear distorted.

Insert an Image



Previous Next Cancel

Step 7: Choose the tabs we want to include with our App and click Next button.
 Click the Add and Remove arrow buttons to add or remove tabs from the App.
 Click the up and down arrow buttons to define in which order we want the tabs to display in the App.

Also we can set the default landing tab for our newly created App using the Default Landing Tab drop-down menu below the list of selected tabs. This determines the first tab a user sees when logging into this Salesforce App.

Choose the tabs to include in this custom app.

Available Tabs	Selected Tabs
Price Books	Home
Accounts	
Campaigns	
Cases	
Contacts	
Contracts	
Dashboards	
Documents	
Forecasts	
Leads	
Opportunities	
Orders	
Products	
Reports	

Add
Remove

Up
Down

Default Landing Tab | Home |

Step 8: Check the Visible checkbox for choosing the user profile for which we want the App will be available.

Check the default box for setting the App as that profile's default App. This means that new users who have the same profile will see this App when they log in for the first time.

Finally, Click Save to finish the wizard and then our application got created successfully.

Result:

By following the above steps a Warehouse app is created in salesforce.com with custom objects.

Aim:

To Create the Merchandise object in salesforce.com cloud environment.

Concept:

Salesforce is a cloud computing Service as a Software (SaaS) company that specializes in Customer Relationship Management (CRM). Salesforce's services allow businesses to use cloud technology to better connect with customers, partners and potential customers. An object is created manually with custom application.

Process/Procedure:

1. Our browser should already be open, log into our salesforce account
2. From Setup, enter Objects in the Quick Find box, and then select Objects.
3. Click New Custom Object to display the New Custom Object wizard.
4. For the Label and Plural Label enter Merchandise, leave all other values as they are.
5. Click Save to finish creating our new object.

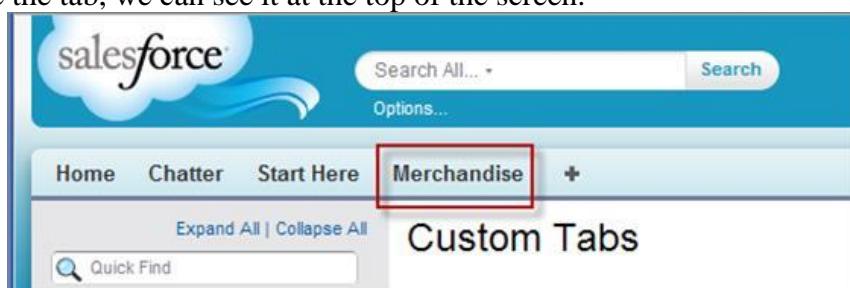
Experiment:

Merchandise – Using this object we will buy the items into our Warehouse.

Step 1: To display information about the Merchandise object, we associate the object with a tab.

1. From Setup, enter Tabs in the Quick Find box, then select Tabs.
2. In the Custom Objects Tabs related list, click New to launch the New Custom Tab wizard.
3. From the Object drop-down list, select Merchandise.
4. For the Tab Style, click the lookup icon and choose an icon.
5. Accept the remaining defaults, and click Next, Next, and then Save.

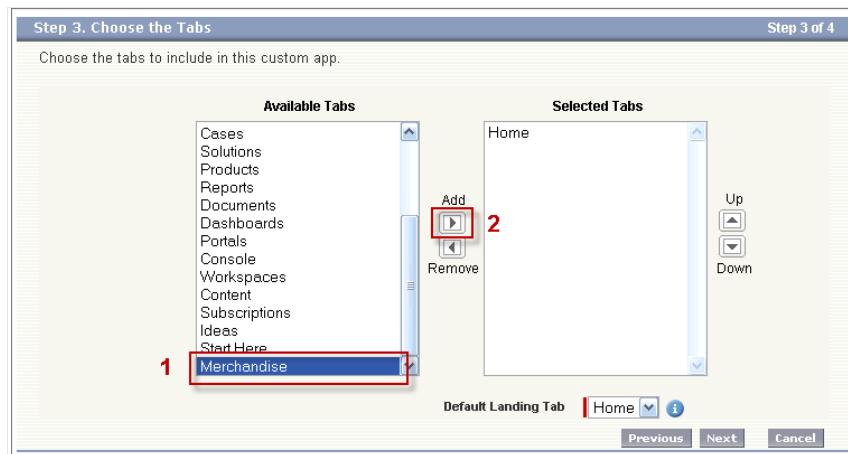
As soon as we create the tab, we can see it at the top of the screen.



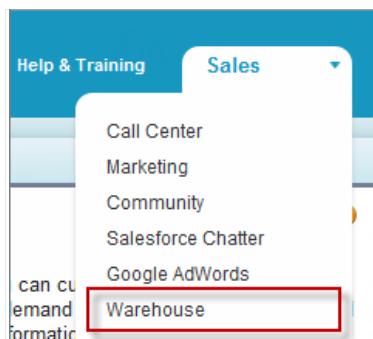
Step 2: Next create an app that contains the tab.

1. From Setup, enter Apps in the Quick Find box, then select Apps.

2. Click New to launch the New Custom App wizard.
3. For the App Label and App Name enter Warehouse.
4. Click Next and Next again.
5. In the Available Tabs list, locate the Merchandise tab and click Add to add it to the list of selected tabs.



6. Leave the Default Landing Tab set to the Home tab, and click Next.
7. Select the Visible checkbox to make the application available to all user profiles.
8. Click Save to create the Warehouse application.
9. The app now appears in the Force.com app menu in the upper right corner of the page. Click it.



Result:

The above steps are used for object creation using salesforce.com.

Aim:

To Create an Application in Salesforce.com using Apex programming Language

Concept:

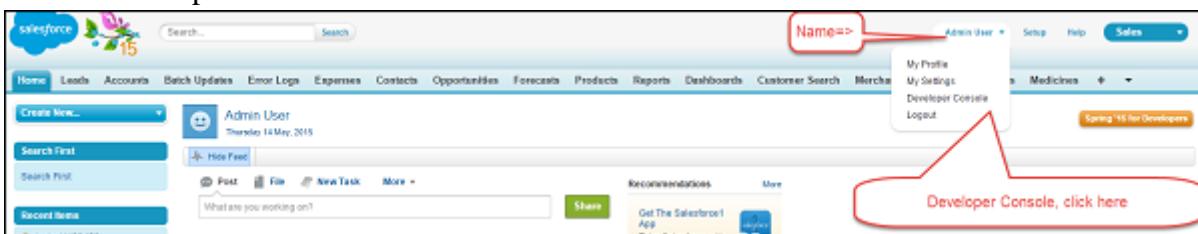
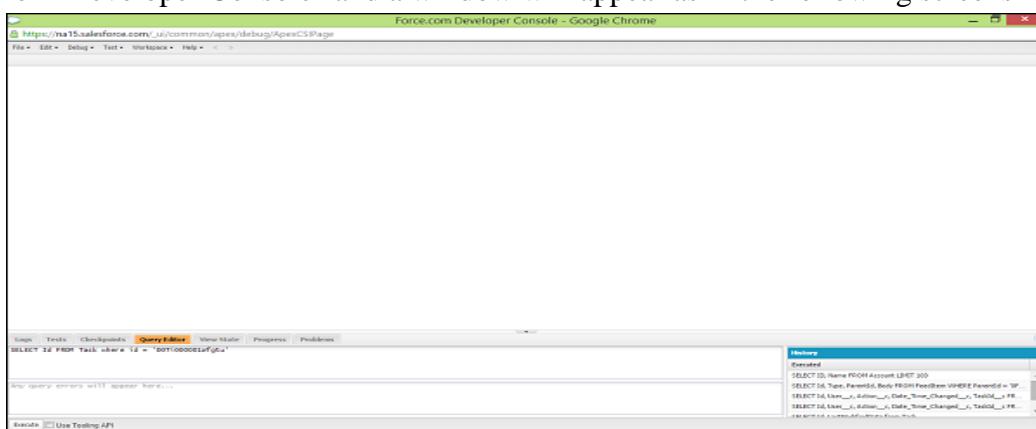
The Developer Console is an Integrated Development Environment with a collection of tools we can use to create, debug, and test applications in Salesforce.com

Process /Procedure:

1. Login to salesforce account
2. Go to Name → Developer Console
3. Click on "Developer Console"
4. Write Apex Code in the console
5. Click on **Execute**, the debug logs will open. Once the log appears then clicks on the log record.
6. Create a New Apex class using File ⇒ New and then click on the Apex class.
7. Click on 'New' and then provide the Name for class and then click Save.
8. Write the Class properties and sample methods of the class

Experiment:**Follow these steps to open the Developer Console –****Step 1:** Login to salesforce account

Go to Name → Developer Console

**Step 2:** Click on "Developer Console" and a window will appear as in the following screenshot.**Following are a few operations that can be performed using the Developer Console.**

- **Writing and compiling code** – We can write the code using the source code editor. When we save a trigger or class, the code is automatically compiled. Any compilation errors will be reported.
- **Debugging** – We can write the code using the source code editor. When we save a trigger or class, the code is automatically compiled. Any compilation errors will be reported.
- **Testing** – We can view debug logs and set checkpoints that aid in debugging.
- **Checking performance** – We can execute tests of specific test classes or all classes in our organization, and we can view test results. Also, we can inspect code coverage.

- **SOQL queries** – We can inspect debug logs to locate performance bottlenecks.
- **Color coding and autocomplete** – The source code editor uses a color scheme for easier readability of code elements and provides auto completion for class and method names.

Executing APEX Code in Developer Console

All the code snippets mentioned in this tutorial need to be executed in the developer console. Follow these steps to execute steps in Developer Console.

Step 1: Login to the Salesforce.com using **login.salesforce.com**. Copy the code snippets mentioned in the tutorial. For now, we will use the following sample code.

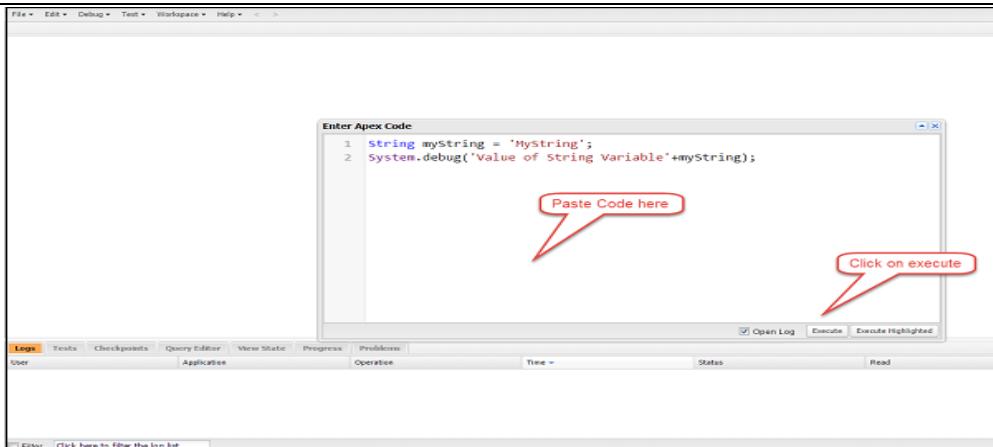
```
String myString = 'MyString';
System.debug('Value of String Variable'+myString);
```



Step 2: To open the Developer Console, click on Name → Developer Console and then click on Execute Anonymous as shown below.

The image consists of two screenshots. The top screenshot shows the Salesforce homepage with a navigation bar at the top. On the right side of the page, there is a user menu with options like 'Admin User', 'My Profile', 'My Settings', 'Developer Console', and 'Logout'. A red box highlights the 'Developer Console' option. The bottom screenshot shows a close-up of the developer console interface. It has a toolbar at the top with various options. One of the buttons in the toolbar is labeled 'Click Here to open execute anonymous', which is also highlighted with a red box. The main area of the interface shows a log or list of items, with tabs for 'Logs', 'Texts', 'Checkpoints', etc.

Step 3: In this step, a window will appear and we can paste the code there.



Step 4: When we click on Execute, the debug logs will open. Once the log appears in window as shown below, then click on the log record.

Execution Log						
User	Application	Operation	Time	Status	Read	Size
Admin User	00000000001807	/services/data/v3.0/tooling/executeAn...	6/4/2015, 11:15:47 AM	Success	1.27 kB	
Admin User	00000000001807	/services/data/v3.0/tooling/executeAn...	6/4/2015, 11:14:59 AM	Success	1.27 kB	

A red callout points to the first log entry with the text: "Always click on the first log as it is latest".

Then type 'USER' in the window as shown below and the output statement will appear in the debug window. This 'USER' statement is used for filtering the output.

Log executeAnonymous @5/8/2015, 5:03:11 PM						
Execution Log			Output of Debug Statement, value of String variable			
Timestamp	Event	Details				
17:03:11:042	USER_DEBUG	[2]DEBUGValue of String VariableMyString				

A red callout points to the input field with the text: "Enter here a String as 'USER' to see the output".

Result:

By following the above steps an application using Apex programming language is created in salesforce.com - Developer Console IDE.

Aim:

To create class in Apex from the Developer Console, force.com

Concept:

The Developer Console is an Integrated Development Environment with a collection of tools we can use to create, debug, and test applications in our force organization.

Process/ Procedure:

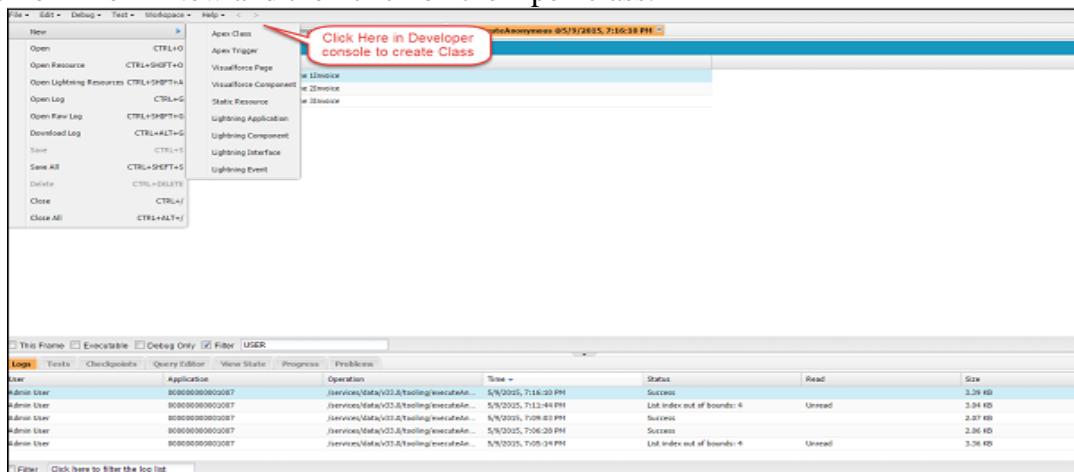
1. Go to Name and click on the Developer Console.
2. Click on File ⇒ New and then click on the Apex class.
3. Open Force.com Eclipse IDE
4. Create a New Project by clicking on File ⇒ New ⇒ Apex Class.
5. Provide the Name for the Class and click on OK.

Experiment:**From Developer Console**

Follow these steps to create an Apex class from the Developer Console –

Step 1: Go to Name and click on the Developer Console.

Step 2: Click on File ⇒ New and then click on the Apex class.

**From Force.com IDE**

Follow these steps to create a class from Force.com IDE –

Step 1: Open Force.com Eclipse IDE

Step 2: Create a New Project by clicking on File ⇒ New ⇒ Apex Class.

Step 3: Provide the Name for the Class and click on OK.

Once this is done, the new class will be created.

From Apex Class Detail Page

Follow these steps to create a class from Apex Class Detail Page –

Step 1: Click on Name ⇒ Setup.

Step 2: Search for 'Apex Class' and click on the link. It will open the Apex Class details page.

Step 3: Click on 'New' and then provide the Name for class and then click Save.

Example

Following is a sample structure for Apex class definition –

```
public class MySampleApexClass
{
    //Class definition and body
    public static Integer myValue = 0; //Class Member variable
    public static String myString = ""; //Class Member variable
    public static Integer getCalculatedValue () {
        // Method definition and body
        // do some calculation
        myValue = myValue+10;
        return myValue;
    }
}
```

Result:

By following the above steps an application using Apex class definition programming is created in force.com - Developer Console IDE.

Aim:

To Explore the Hand-on-lab features provided by Qwiklabs online cloud lab learning environment.

Concept:

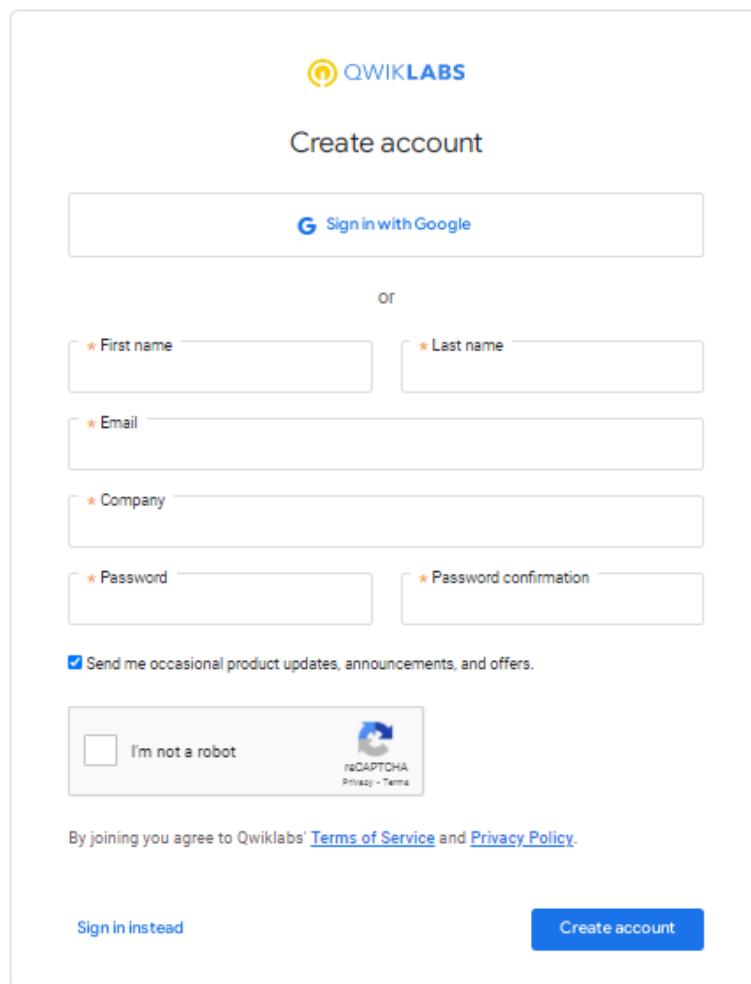
Qwiklabs provides lab learning environments that help developers and IT professionals get hands-on experience working with leading cloud platforms and software. A hands-on lab, also referred to as a “lab”, is a learning experience in Qwiklabs where we complete a scenario based use case by following a set of instructions in a specified amount of time in an interactive hands-on environment. Labs are completed in the real GCP console, as opposed to a simulation or demo environment.

Process/Procedure :

1. Go to <https://www.qwiklabs.com/>
2. Create a new Qwiklabs account and Login
3. After LOGIN Explore the Following Features of Qwiklabs Online Cloud Lab Learning Environment
4. Google Cloud Essentials –begin
 - A Tour of Google Cloud Hands-on Labs
 - Creating a Virtual Machine
 - Compute Engine: Qwik Start - Windows
 - Getting Started with Cloud Shell and gcloud
 - Kubernetes Engine: Qwik Start
 - Set Up Network and HTTP Load Balancers

Experiment

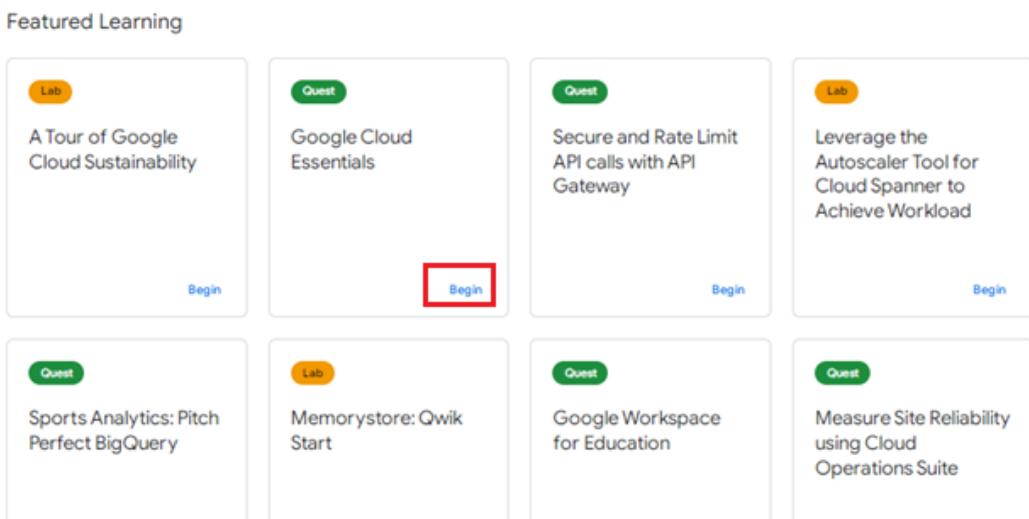
Step 1: Create a new Qwiklabs account and Login



The image shows the 'Create account' page for Qwiklabs. At the top is the Qwiklabs logo. Below it is a large button labeled 'G Sign in with Google'. Underneath this is a section labeled 'or' with four input fields: 'First name', 'Last name', 'Email', and 'Company'. Below these is a section for 'Password' with two fields: 'Password' and 'Password confirmation'. There is also a checkbox for 'Send me occasional product updates, announcements, and offers.' A reCAPTCHA box follows, containing a checkbox for 'I'm not a robot' and a reCAPTCHA logo. At the bottom, there is a note about agreeing to the 'Terms of Service' and 'Privacy Policy', followed by two buttons: 'Sign in instead' and 'Create account'.

After LOGIN Explore the Following Features of Qwiklabs Online Cloud Lab Learning Environment

Step 2: Google Cloud Essentials –begin



The image displays a grid of eight learning modules from Google Cloud Essentials:

- A Tour of Google Cloud Sustainability (Lab)
- Google Cloud Essentials (Quest)
- Secure and Rate Limit API calls with API Gateway (Quest)
- Leverage the Autoscaler Tool for Cloud Spanner to Achieve Workload (Lab)
- Sports Analytics: Pitch Perfect BigQuery (Quest)
- Memorystore: Qwik Start (Lab)
- Google Workspace for Education (Quest)
- Measure Site Reliability using Cloud Operations Suite (Quest)

The 'Memorystore: Qwik Start' module is highlighted with a red box around its 'Begin' button.

1. A Tour of Google Cloud Hands-on Labs
2. Creating a Virtual Machine
3. Compute Engine: Qwik Start - Windows
4. Getting Started with Cloud Shell and gcloud
5. Kubernetes Engine: Qwik Start
6. Set Up Network and HTTP Load Balancers

Lab

A Tour of Google Cloud Hands-on Labs

In this first hands-on lab you will access the Google Cloud Platform Console and use these basic Google Cloud features: Projects, Resources, IAM Users, Roles, Permissions, and APIs.

★★★★★ 30 minutes Introductory Free  

Lab

Creating a Virtual Machine

In this hands-on lab, you'll learn how to create a Google Compute Engine virtual machine and understand zones, regions, and machine types. To preview, watch the short video [Create a Virtual Machine, GCP Essentials](#).

★★★★★ 40 minutes Introductory 1 Credit  

OR

Lab

Compute Engine: Qwik Start - Windows

Google Compute Engine lets you create and run virtual machines on Google infrastructure. In this lab you create a Windows Server instance in the Google Compute Engine and access it with RDP. Watch a short preview, [Launch a Windows Server Instance, GCP Essentials](#).

★★★★★ 40 minutes Introductory 1 Credit  

Lab

Getting Started with Cloud Shell and gcloud

In this hands-on lab you will learn how to connect to computing resources hosted on Google Cloud Platform via the web. You will also learn how to use Cloud Shell and the Cloud SDK gcloud command. For a preview, watch the short video [Get Started with Cloud Shell, GCP Essentials](#).

★★★★★ 45 minutes Introductory 1 Credit  

Lab

Kubernetes Engine: Qwik Start

Google Kubernetes Engine provides a managed environment for deploying, managing, and scaling your containerized applications using Google infrastructure. This hands-on lab shows you how deploy a containerized application with Kubernetes Engine. Watch the short video [Manage Containerized Apps with Kubernetes Engine](#).

★★★★★ 30 minutes Introductory 1 Credit  

Lab

Set Up Network and HTTP Load Balancers

In this hands-on lab, you'll learn how setup both network load balancers and HTTP load balancers for your application running in Compute Engine virtual machines.

★★★★★ 1 hour Introductory 1 Credit  

Result:

Using the Above steps services offered by Qwiklabs online cloud lab learning environment are explored.

Aim:

To Deploy an App in Google Cloud Platform using Qwiklabs interface.

Concept:

Writing Cloud Shell commands and deploying our first virtual machine, to running applications on Kubernetes Engine or with load balancing, Google Cloud Essentials is a prime introduction to the platform's basic features.

Process /Procedure:

Go to <https://www.qwiklabs.com/focuses/2794?parent=catalog> - Start lab

1. Enable App Engine in a Cloud project
 - Select a project, or create a new one.
 - qwiklabs-gcp-02-f315b3ef540d
 - To open Cloud Shell, click Cloud Shell. Show me
 - Enable App Engine:
 - gcloud app create
 - If an authorization window appears, click Authorize.
 - When prompted, choose a region that's closest to we.
 - The process is complete when we get this message:
 - Success! The app is now created.
2. Clone a sample app
 - Instead of creating a new app, clone the sample "Hello, world" app from GitHub:
 - git clone \
 - https://github.com/GoogleCloudPlatform/golang-samples
 - To explore the app's source files, open the app's directory in the Cloud Shell Editor:
 - cloudshell workspace \
 - golang-samples/appengine/go11x/helloworld
 - The app consists of:
 - helloworld.go, a Go file with the function that responds to requests with "Hello, World!"
 - helloworld_test.go, a file for tests
 - go.mod, a module file
 - app.yaml, an App Engine configuration file with the minimum required settings
3. Test the app before deploying it
 - To open a shell in Cloud Shell Editor, click the Terminal menu and select **New Terminal**.
 - To run the app, use the following command:
 - go run .
 - When the app is running, the output in Cloud Shell shows the following lines:
 - [timestamp] Defaulting to port 8080
 - [timestamp] Listening on port 8080
 - In the Console, click Web preview and choose **Preview on port 8080**.
 - The app's tab displays "Hello, World!".
 - To stop the app, in Cloud Shell enter Ctrl+C.
4. Deploy the app
 - Now that we know our app is running correctly, deploy it to App Engine:
 - In Cloud Shell, configure gcloud to use our project:
 - gcloud config set project \
 - qwiklabs-gcp-02-f315b3ef540d
 - The App Engine plugin uses gcloud to deploy our app, and gcloud needs to know which project to deploy our app to.

- gcloud app deploy
 - If we are prompted, "Do we want to continue?", type **Y**.
 - When this process is finished, the output shows the following:
 - Deployed service [default] to https://qwiklabs-gcp-02-f315b3ef540d.appspot.com
 - If we get a NOT_FOUND: Unable to retrieve P4SA error after running the deploy command, try waiting 60 seconds then run the gcloud app deploy command again.
- 5.** View and monitor the app
- Visit the deployed app at its default URL: qwiklabs-gcp-02-f315b3ef540d.appspot.com.
 - To view the App Engine dashboard, in the Cloud Console navigation menu select App Engine
- 6.** Clean up
- In the App Engine navigation menu, click Settings.
 - Click Disable Application.
 - Or we can delete the project that we created.

Experiment:

Step 1: Go to <https://www.qwiklabs.com/>

The screenshot shows the Qwiklabs Home page. At the top, there's a navigation bar with links for Apps, Free Basic Excel Co..., KLU LMS: Administr..., Welcome, KL Mail, ADDITIONAL CORS..., KL ERP, data str, 2020-21_COURSES..., house sales coursera, and a Reading list. Below the navigation bar, there are several cards representing learning paths:

- Featured Learning:**
 - A Tour of Google Cloud Sustainability (Lab)
 - Google Cloud Essentials (Quest) - This card is circled in red.
 - Secure and Rate Limit API calls with API Gateway (Quest)
 - Leverage the Autoscaler Tool for Cloud Spanner to Achieve Workload (Lab)
- Other Learning Paths:**
 - Sports Analytics: Pitch Perfect BigQuery (Quest)
 - Memorystore: Qwik Start (Lab)
 - Google Workspace for Education (Quest)
 - Measure Site Reliability using Cloud Operations Suite (Quest)

Step 2: Click on **Begin -- Google Cloud Essentials** –

The screenshot shows the details page for the 'Google Cloud Essentials' quest. At the top, there's a navigation bar with links for Apps, Free Basic Excel Co..., KLU LMS: Administr..., Welcome, KL Mail, ADDITIONAL CORS..., KL ERP, data str, 2020-21_COURSES..., house sales coursera, and a Reading list. Below the navigation bar, there's a sidebar titled 'Quest Info' with sections for Prerequisites and Available languages.

The main content area displays two lab components:

- A Tour of Google Cloud Hands-on Labs**: A 30-minute introductory free lab. It includes a brief description: "In this first hands-on lab you will access the Google Cloud Platform Console and use these basic Google Cloud features: Projects, Resources, IAM Users, Roles, Permissions, and APIs.", a rating of 4 stars, and a link to watch a preview video.
- Creating a Virtual Machine**: A 40-minute introductory 1 Credit lab. It includes a brief description: "In this hands-on lab, you'll learn how to create a Google Compute Engine virtual machine and understand zones, regions, and machine types. To preview, watch the short video Create a Virtual Machine, GCP Essentials.", a rating of 4 stars, and a link to watch a preview video.

Step 3: Click on A Tour of Google Cloud hand –on labs

A Tour of Google Cloud Hands-on Labs

Start Lab 00:45:00

A Tour of Google Cloud Hands-on Labs

45 minutes Free ★★★★☆

GSP282

Google Cloud Self-Paced Labs

Overview	
Lab fundamentals	
Accessing the Cloud Console	
Projects in the Cloud Console	
Navigation menu and services	
Roles and permissions	
APIs and services	
Ending your lab	
Congratulations!	

Step 4: Read all Instructions of Google Cloud Self –Paced Labs

Step 5: Click on START LAB –(Green Color Button)

End Lab 00:44:56

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked.
Learn more.

Open Google Console

Username: student-02-44d3ee0b8122@com

Password: L5WYqHynqqt1

GCP Project ID: quicklabs-gcp-02-2e5b335ef

A Tour of Google Cloud Hands-on Labs

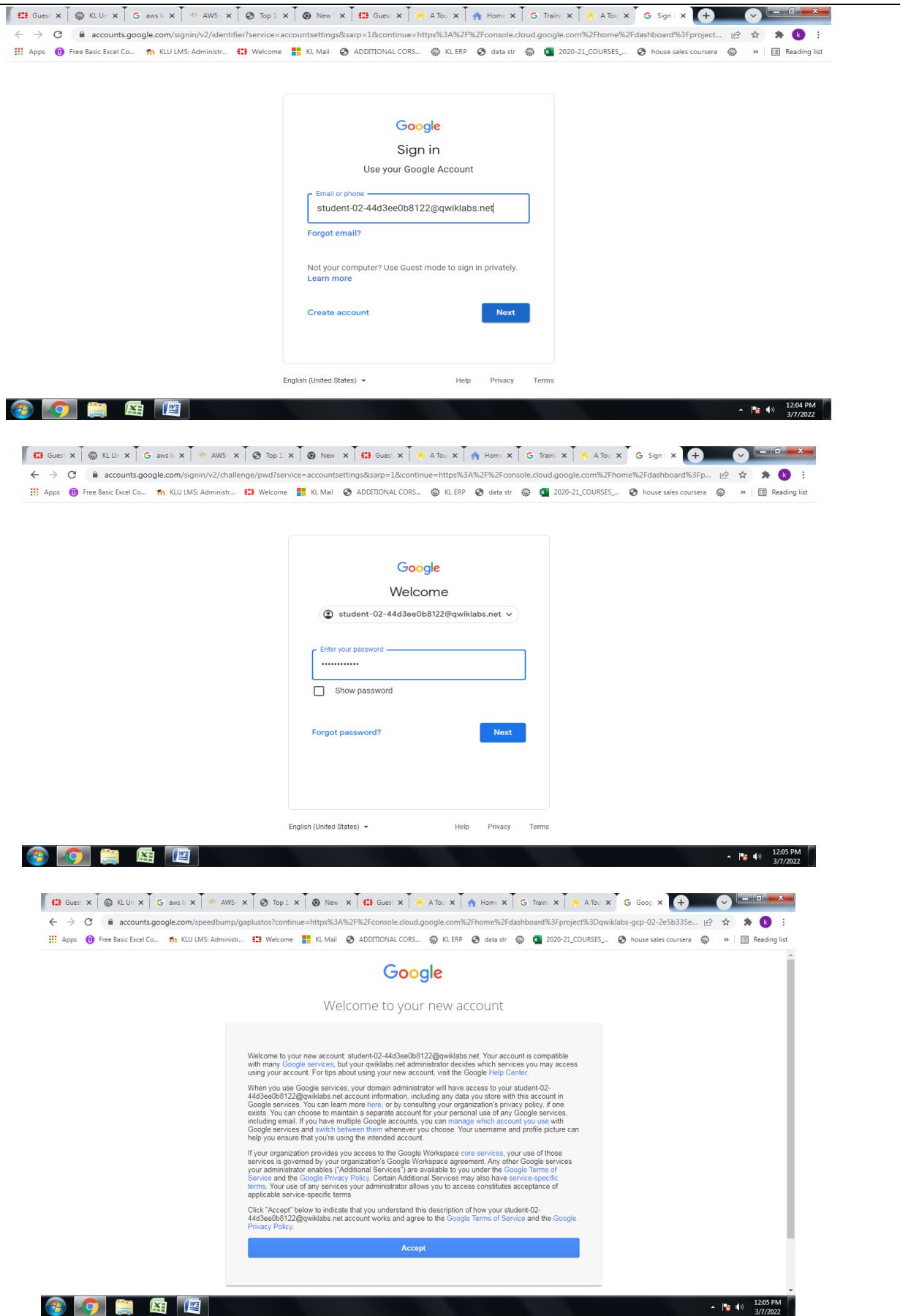
45 minutes Free ★★★★☆

GSP282

Google Cloud Self-Paced Labs

Overview	
Lab fundamentals	
Accessing the Cloud Console	
Projects in the Cloud Console	
Navigation menu and services	
Roles and permissions	
APIs and services	
Ending your lab	
Congratulations!	

Step 6: Click on Open Google Console and Login with Username ,password generated on Left side of the above window



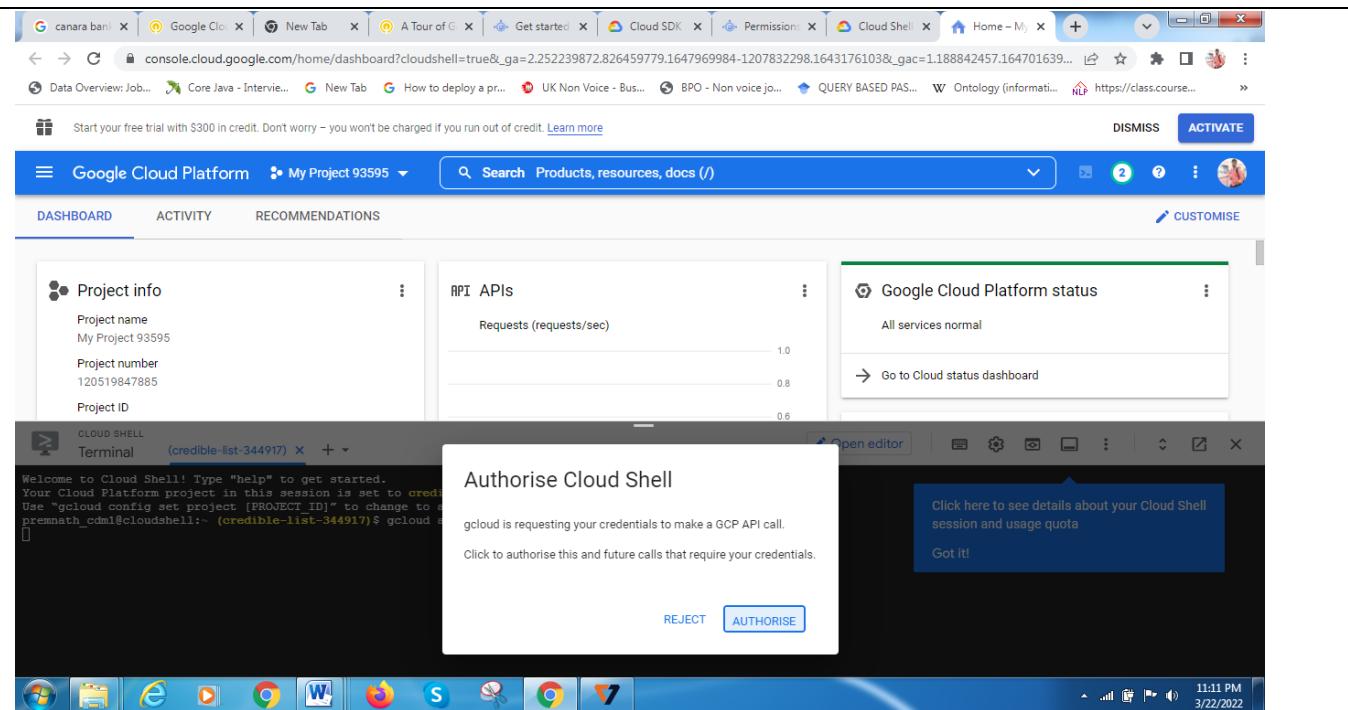
Step 7: Enable App Engine in a Cloud project

- Select a project, or create a new one.
 - qwiklabs-gcp-02-f315b3ef540d
- To open Cloud Shell, click Cloud Shell. Show me
- Enable App Engine:

- `gcloud app create`
- If an authorization window appears, click Authorize.
- When prompted, choose a region that's closest to we.
- The process is complete when we get this message:
 - Success! The app is now created.

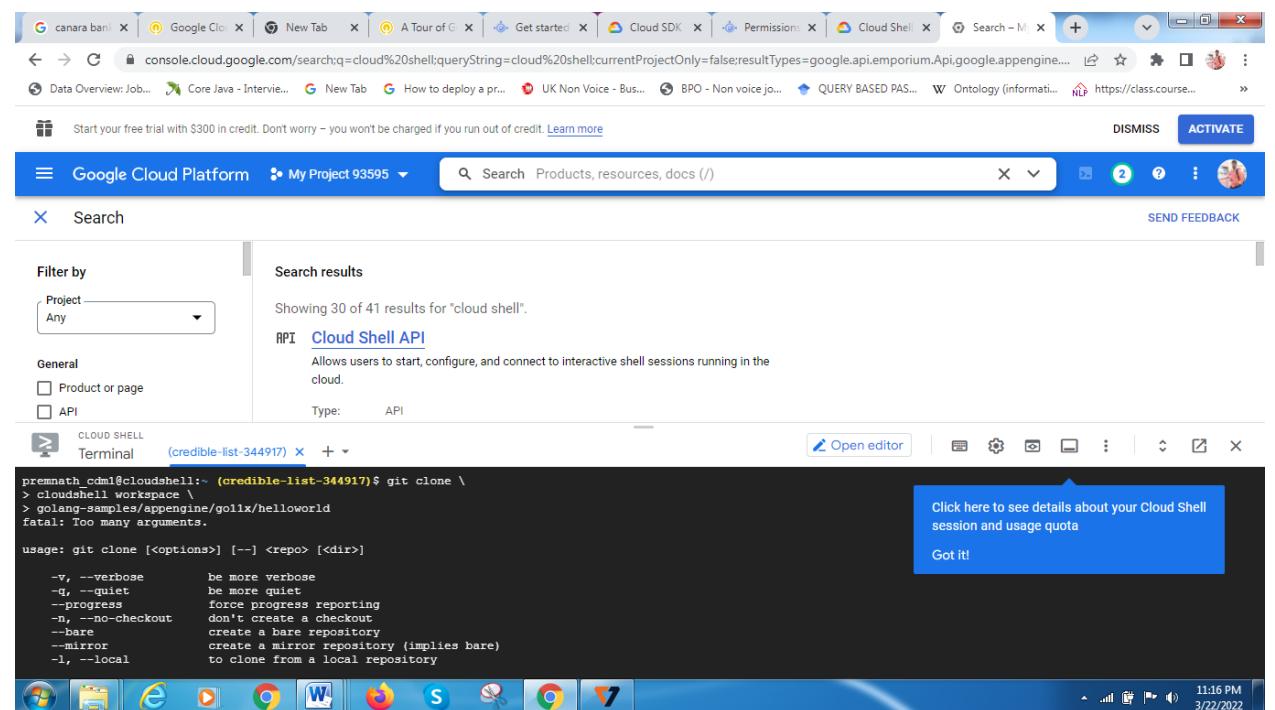
The screenshot shows the Google Cloud Platform dashboard for a project named "My Project 93595". On the left, there's a sidebar with links like Instances, Task queues, Cron jobs, Security scans, Firewall rules, Quotas, Memcache, Search, Settings, and Release notes. The main area has a search bar at the top with the query "app engine". A modal window titled "Configure application" is open, with the sub-section "Creating application." visible. The modal lists several options under "PRODUCTS & PAGES" and "DOCUMENTATION AND TUTORIALS". To the right of the modal, there's a sidebar titled "Recommended for you" with links to "App Engine overview", "Choosing an App Engine environment", "Structuring web services in App Engine", "Installing an SDK for App Engine", "App Engine pricing", and "Quotas in App Engine". The status bar at the bottom shows the date as 3/22/2022 and the time as 11:06 PM.

The screenshot shows the Google Cloud Platform dashboard for the same project. A modal window titled "Cloud Shell" is open, displaying a terminal session. The terminal output shows a command being run: `git config --remote .sparse-checkout add .`. Below the terminal, there's a "DOCUMENTATION AND TUTORIALS" section with links to "Learn about the Cloud Shell CLI" and another link. There's also a "MARKETPLACE" section listing "Cloud Shell API", "Ansible AWX (Open source alternative to Ansible Tower)", and "Apache Web Server + MySQL + phpMyAdmin on CentOS Server 8.3". The status bar at the bottom shows the date as 3/22/2022 and the time as 11:15 PM.



Step 8: Clone a sample app

- Instead of creating a new app, clone the sample "Hello, world" app from GitHub:
- `git clone \`
- <https://github.com/GoogleCloudPlatform/golang-samples>
- To explore the app's source files, open the app's directory in the Cloud Shell Editor:
- `cloudshell workspace \`
- `golang-samples/appengine/go11x/helloworld`
- The app consists of:
- `helloworld.go`, a Go file with the function that responds to requests with "Hello, World!"
- `helloworld_test.go`, a file for tests
- `go.mod`, a module file
- `app.yaml`, an App Engine configuration file with the minimum required settings



Step 9: Test the app before deploying it

- To open a shell in Cloud Shell Editor, click the Terminal menu and select **New Terminal**.
- To run the app, use the following command:
• go run .
- When the app is running, the output in Cloud Shell shows the following lines:
 - [timestamp] Defaulting to port 8080
 - [timestamp] Listening on port 8080
- In the Console, click Web preview and choose **Preview on port 8080**.
- The app's tab displays "Hello, World!".
- To stop the app, in Cloud Shell enter Ctrl+C.

Step 10: Deploy the app

- Now that we know our app is running correctly, deploy it to App Engine:
- In Cloud Shell, configure gcloud to use our project:
 - gcloud config set project \
 - quicklabs-gcp-02-f315b3ef540d
- The App Engine plugin uses gcloud to deploy our app, and gcloud needs to know which project to deploy our app to.
- gcloud app deploy
- If we are prompted, "Do we want to continue?", type **Y**.
- When this process is finished, the output shows the following:
 - Deployed service [default] to https://quicklabs-gcp-02-f315b3ef540d.appspot.com
- If we get a NOT_FOUND: Unable to retrieve P4SA error after running the deploy command, try waiting 60 seconds then run the gcloud app deploy command again.

Step 11: View and monitor the app

- Visit the deployed app at its default URL: quicklabs-gcp-02-f315b3ef540d.appspot.com.
- To view the App Engine dashboard, in the Cloud Console navigation menu select App Engine

Step 12: Clean up

- In the App Engine navigation menu, click Settings.
- Click Disable Application.
- Or we can delete the project that we created.

Result:

Using the Above steps a simple app (Hello World) can be deployed in google cloud platform

Aim:

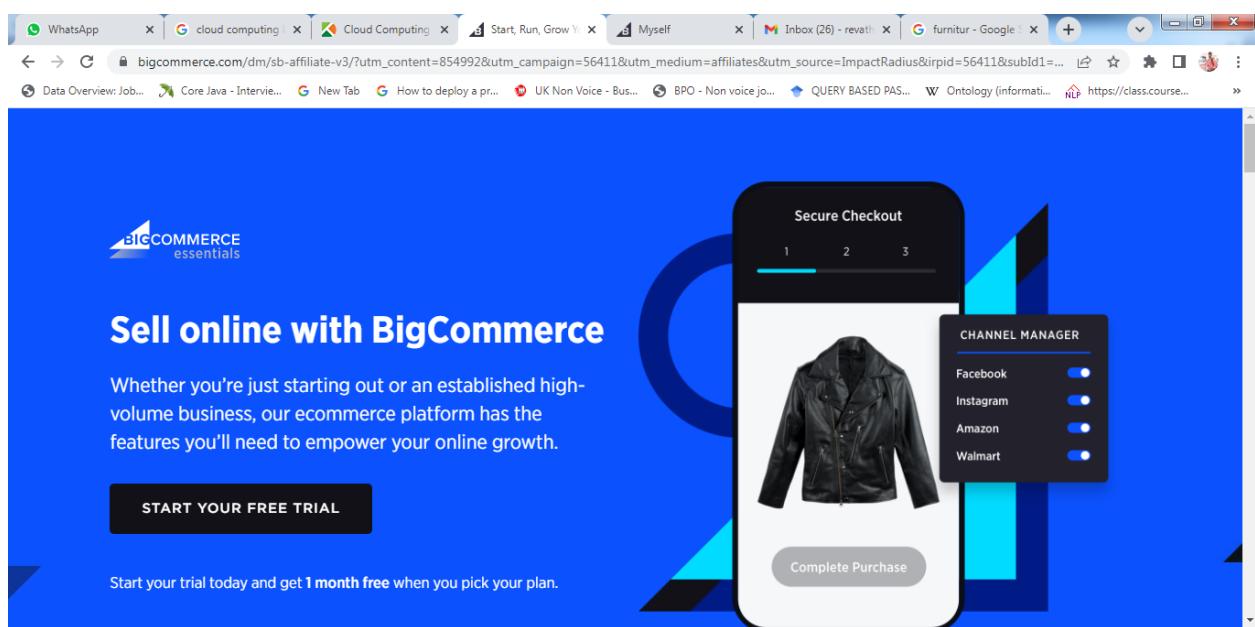
To explore the features provided by Cloud Resources in bigcommerce online cloud lab learning environment.

Concept:

BigCommerce is a public-traded e-commerce platform offering Software-as-a-Service (SaaS) services for building and hosting online stores.

Process/Procedure :

5. Go to <https://www.bigcommerce.com/start-your-trial/>
6. Create a new bigcommerce account and Login
7. After LOGIN Explore the Following Features of bigcommerce Online store
8. Create and Manage Cloud Resources—begin

Experiment**Step 1:** Create a new bigcommerce account and Login

After LOGIN Explore the Following Features of bigcommerce Online Cloud Lab Learning Environment

Step 2: Create and Manage Cloud Resources—begin

The screenshot shows the BigCommerce trial setup page. On the left, there's a large graphic of a blue 'C' shape. A central callout box contains the text "Secure Checkout" with a progress bar at step 1, and a "CHANNEL MANAGER" section with toggles for Facebook and Instagram. To the right, the text "Your BigCommerce store is ready" is displayed, followed by a placeholder text: "Please tell us a little bit about your business so we can help you make the most of your trial." Below this are three dropdown menus: "Do you have an existing website?", "What will you be selling?", and "What best describes how you currently sell your products?". At the bottom right of the page is a "Show all" button.

This screenshot shows the continuation of the BigCommerce trial setup. The central graphic now includes a "Complete Purchase" button. The "CHANNEL MANAGER" section has been updated to include Amazon and Walmart, with all toggles turned on. The three dropdown menus remain the same. At the bottom right, there are two buttons: a large blue "FINISH" button and a smaller "Ask me later" button.

Step 3: Create our own store

bigcommerce.com/start-your-trial/

Create your beautiful store today

Get one month free when you pick your plan.

First Name

Last Name

Phone number

Email address

Password

Store name (You can change this at any time)

What size is your online business?

Show all

bigcommerce.com/start-your-trial/

Create your beautiful store today

Get one month free when you pick your plan.

Phone number

Email address

Password

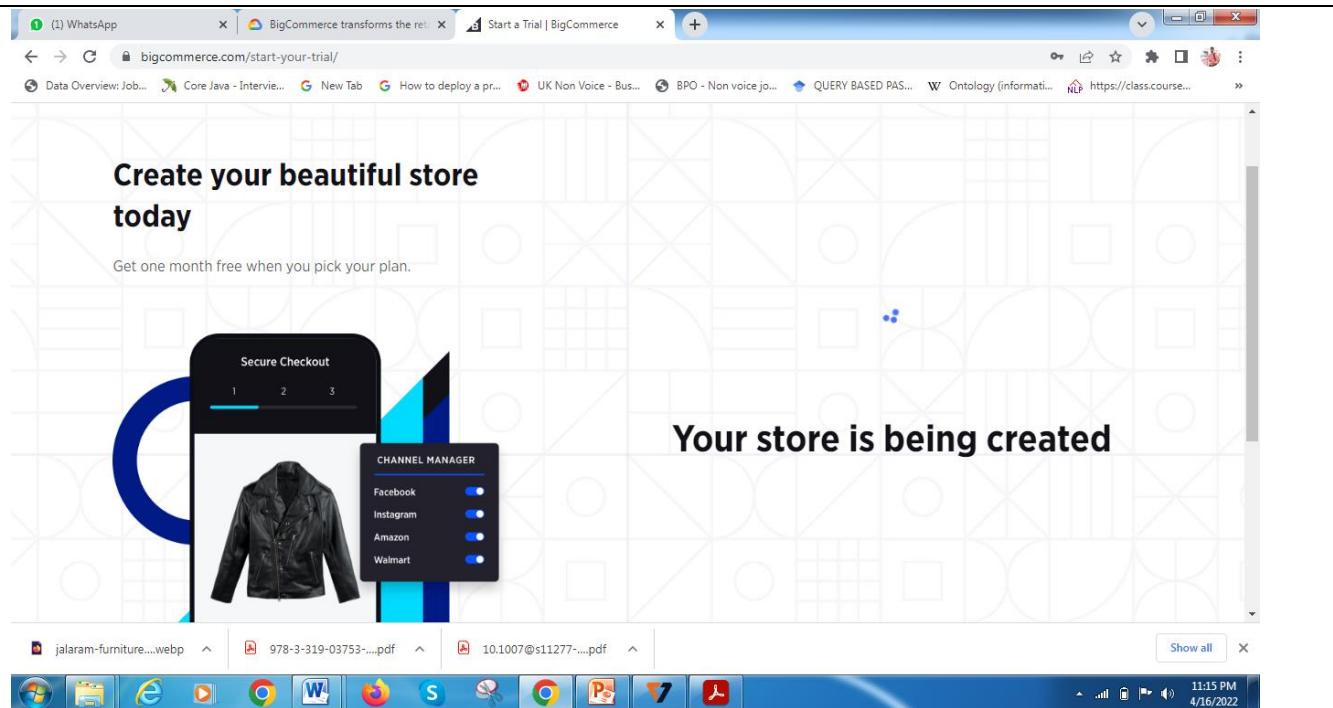
Store name (You can change this at any time)

What size is your online business?

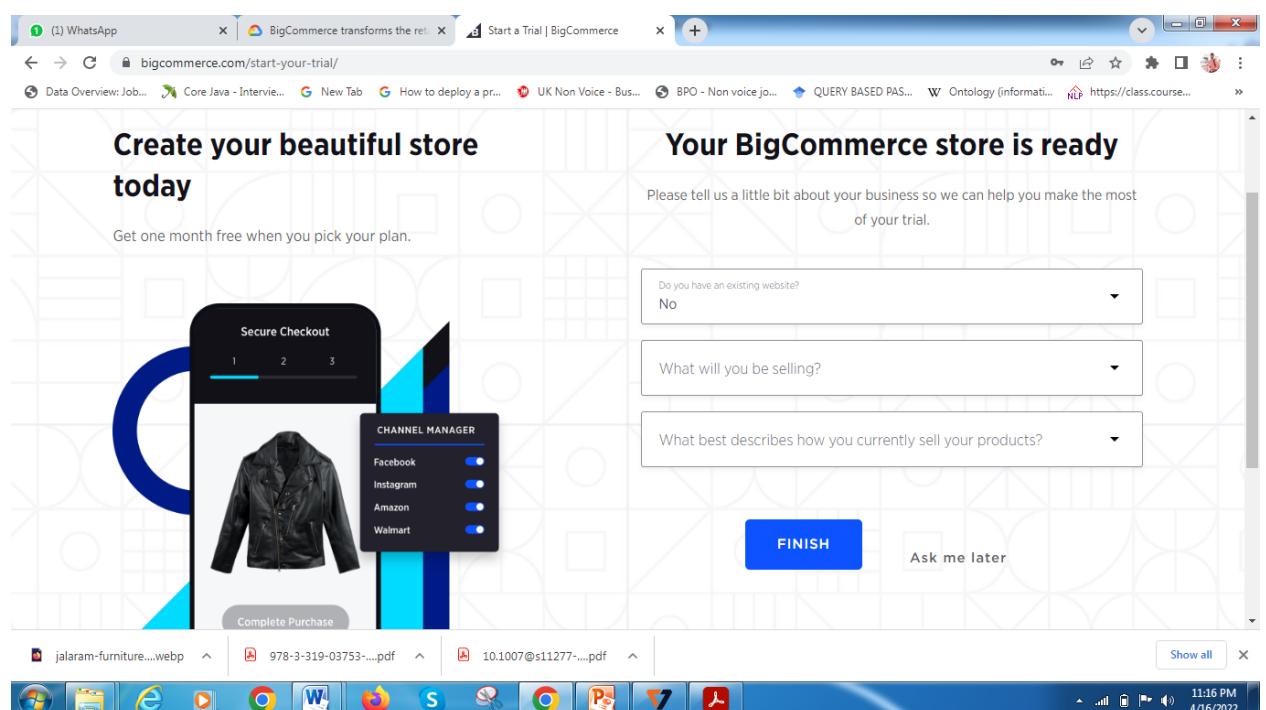
CREATE YOUR STORE

By providing your email, you are agreeing to our [terms of service](#).

Show all



Step 4: Select your selling goods



Step 5: Add products to sell using Add Product icon.

START ACCEPTING ORDERS

1. Add Products
Populate your store with products to sell.
Add Products
2. Get set to ship
Set up your shipping services and methods.
Set up shipping
3. Setup payments
Get started with PayU for fast and secure payments.
Complete setup

List your products on Google
Show your products to the millions of users searching on Google every day
Get started

Review & test your store
Let people preview your store with preview code.
View Store

VIEW PRODUCTS

Add Product

PRODUCT INFORMATION

Basic Information

Description
Images & Videos
Product Identifiers
Pricing
Inventory

PRODUCT OPTIONS

Product Information
Information to help define a product.

Basic Information

Visible on Storefront

Product Name *

SKU

Save

Step 6: Save the product

The screenshot shows the BigCommerce 'Add Product' interface. On the left, a sidebar menu includes 'View Store', 'Home', 'Search', 'Products' (selected), 'View', 'Add', 'Search', 'Import', and 'Export'. The main area has tabs for 'Basic Information', 'Description', 'Images & Videos', 'Product Identifiers', 'Pricing', 'Inventory', 'PRODUCT OPTIONS', 'Variations', and 'Customizations'. A 'Categories' section on the right lists 'Shop All', 'Bath', 'Garden' (selected with a checkmark), 'Kitchen', 'Publications', and 'Utility'. A progress bar at the bottom indicates 'Loading your Product'.

Step 7: Shipping country and minimum amount configuration

The screenshot shows the BigCommerce dashboard. The sidebar menu includes 'View Store', 'Home', 'Search', 'Orders', 'Products', 'Customers', 'Storefront', 'Marketing', 'Analytics' (selected), and 'Help'. The main content area displays a progress bar with steps: '2. Shipping configured' (checkmark), '3. Setup payments' (radio button), '4. Set up your tax rates'.

Step 8: Go to Setup payment option for learning the payment applications

The screenshot shows the BigCommerce dashboard. The sidebar menu includes 'View Store', 'Home', 'Search', 'Orders', 'Products', 'Customers', 'Storefront', 'Marketing', 'Analytics' (selected), and 'Help'. The main content area displays a progress bar with steps: '2. Shipping configured' (checkmark), '3. Setup payments' (radio button highlighted with a red box), '4. Set up your tax rates'.

The screenshot shows the BigCommerce App Marketplace. The URL in the address bar is <https://bigcommerce.co.uk/apps/payu/>. The page displays the PayU app details, including its logo, a rating of 2.0 out of 5 stars from 1 review, and two buttons: 'CONTACT PARTNER' and 'GET THIS APP'. The top navigation bar includes links for 'BIGCOMMERCE', 'APPS & INTEGRATIONS', 'THEMES', 'PARTNER SERVICES', 'OFFERS', 'LOG IN', and 'GET STARTED'.

About the app

PayU is a preferred payment partner for large companies in India – including Flipkart, Netflix, Ola, Tata Cliq, Airbnb, Dream11, Myntra and many more!

Pricing

Custom Price:

Contact us for pricing

The screenshot shows the BigCommerce login page for the PayU app. The URL in the address bar is https://login.bigcommerce.com/deep-links/marketplace/apps/31862?_gl=1*1d77bqz*_ga*MTA3OTA4NzU2NC4xNjUwMTI3Mjg4*_ga_WS2VZYPG6*MTY1MDEyNzI4OC4x.... The page features a heading 'PayU' and a sub-instruction 'To install please select one of the options below'. It provides two buttons: 'Existing BigCommerce store' (with a 'Log in' button) and 'New to BigCommerce?' (with a 'Sign up' button). The browser's top bar shows multiple open tabs, including WhatsApp, Gmail, and various Google search results.

Step 9: Go to setup taxes for set our tax range.

The screenshot shows the BigCommerce 'Myself' dashboard. On the left, a sidebar lists 'Orders', 'Products', 'Customers', 'Storefront', 'Marketing', and 'Analytics'. The main area displays a checklist:

- 2. Shipping configured
- 3. Setup payments
- 4. Set up your tax rates

The 'Set up your tax rates' step is highlighted with a red box around the 'Set Up Taxes' button. To the right, there are sections for 'Review & test your store' and 'Customize your Online Store'.

Step 10: Select products from left side menu and view our products

The screenshot shows the BigCommerce 'Products' page. The left sidebar includes 'View', 'Add', 'Search', 'Import', and 'Export' options. The main area displays a table of products:

Image	Product SKU	Stock Level	Product Name	Price	Action
	001	N/A	Furniture	₹2,000.00	
	SM13	N/A	[Sample] Smith Journal 13	₹25.00	
	DPB	N/A	[Sample] Dustpan & Brush	₹34.95	
	OFSUC	N/A	[Sample] Utility Caddy	₹45.95	
	CLC	N/A	[Sample] Canvas Laundry Cart	₹200.00	

Step 11: Setup tax manually using tax icon

Set up your own tax rules

Manual Tax ENABLED

Manual Tax provides you with the flexibility to configure your own Tax Classes, Rates and Zones. If you have an Automatic Tax service enabled, Manual Tax can act as your Fallback Tax service.

4	1	0
Tax Classes	Tax Zones	Tax Rates

Tax settings Edit

Step 12: Go to store setup and setup payment options, currency option, etc.,

Additional providers

Offline Payment Methods

Online Payment Methods

Digital Wallets

Enable test credit card payments
To process test orders, use card number 4111 1111 1111 1111, name "success" and any future date for expiration date.

The image displays two screenshots of the BigCommerce Control Panel, one above the other, illustrating the configuration of payment methods.

Top Screenshot: Offline Payment Methods

The left sidebar shows the navigation menu with "Payments" selected. The main content area is titled "Offline Payment Methods" and lists the following methods:

- Bank Deposit (with a "Set up" button)
- Cash on Delivery (with a "Set up" button)
- Check (with a "Set up" button)
- Money Order (with a "Set up" button)
- Pay in Store (with a "Set up" button)

Bottom Screenshot: Online Payment Methods

The left sidebar shows the navigation menu with "Payments" selected. The main content area is titled "Online Payment Methods" and lists the following methods:

- Adyen (with a "Set up" button)
- BlueSnap Payments (with a "Set up" button)
- CC Avenue (with a "Set up" button)
- Digital River (with a "Set up" button)
- First Data Payeezy Gateway (with a "Set up" button)
- NMI (with a "Set up" button)

Both screenshots show a standard Windows taskbar at the bottom with icons for File Explorer, Internet Explorer, Google Chrome, Microsoft Word, Mozilla Firefox, and others. The system tray indicates the date and time as 6:54 PM on 4/24/2022.

The screenshot shows the BigCommerce Control Panel for a store named "Myself". The left sidebar is titled "Store Setup" and includes options like "Store Profile", "Currencies", "Payments", "Store Settings", "Shipping", "Tax", and "Accounting". The main content area is titled "Myself" and shows payment integration options: "Sezzle" and "Stripe" with "Set up" buttons; "Digital Wallets" (with "Apple Pay" and "Google Pay" sub-options, each with a "Set up" button); and a checked checkbox for "Enable test credit card payments" with a note below it. The top right corner shows "0 7 Days left in your trial." and a "Select a Plan" button.

Step 13: Go store settings and select the measurement type.

The screenshot shows the "Store Settings" page under "Store Setup". The top navigation tabs are "Website", "Display", "Share", "Date & Timezone", "URL Structure", "Search", "Security & Privacy", and "Miscellaneous". The "Physical Dimension Settings" section is active, showing dropdown menus for "Weight Measurement" (Kilograms), "Length Measurement" (Centimeters), and input fields for "Decimal Token" (.) and "Thousands Token" (,). At the bottom right are "Cancel" and "Save" buttons. The top right corner shows "0 7 Days left in your trial." and a "Select a Plan" button.

Step 14: Finally setup the shipping options.

The screenshot shows the BigCommerce Control Panel interface. The left sidebar is titled "Myself" and includes "Trial Plan Store", "View Store", "Home", "Search", "Store Setup", "Store Profile (NEW)", "Currencies", "Payments", "Store Settings", "Shipping" (which is selected), "Tax", "Accounting", and "Help". The main content area is titled "Myself" and has a sub-section "Checkout Shipping Options". It explains that the Shipping Origin is the address where products are shipped from and used for calculating shipping rates. A button "Add shipping address" is visible. Below this, there's a section titled "Checkout Shipping Options" with a sub-section "Configure your shipping rules". It shows a "Default shipping rules" section for "India" with a toggle switch and a "Configure" button. The top of the screen shows browser tabs and a status bar indicating "6:55 PM 4/24/2022".

Results:

Using the above steps services offered by bigcommerce online cloud shopping application environment are explored.

Aim:

To explore the features provided by Cloud Resources in Adobe Creative Cloud Express account.

Concept:

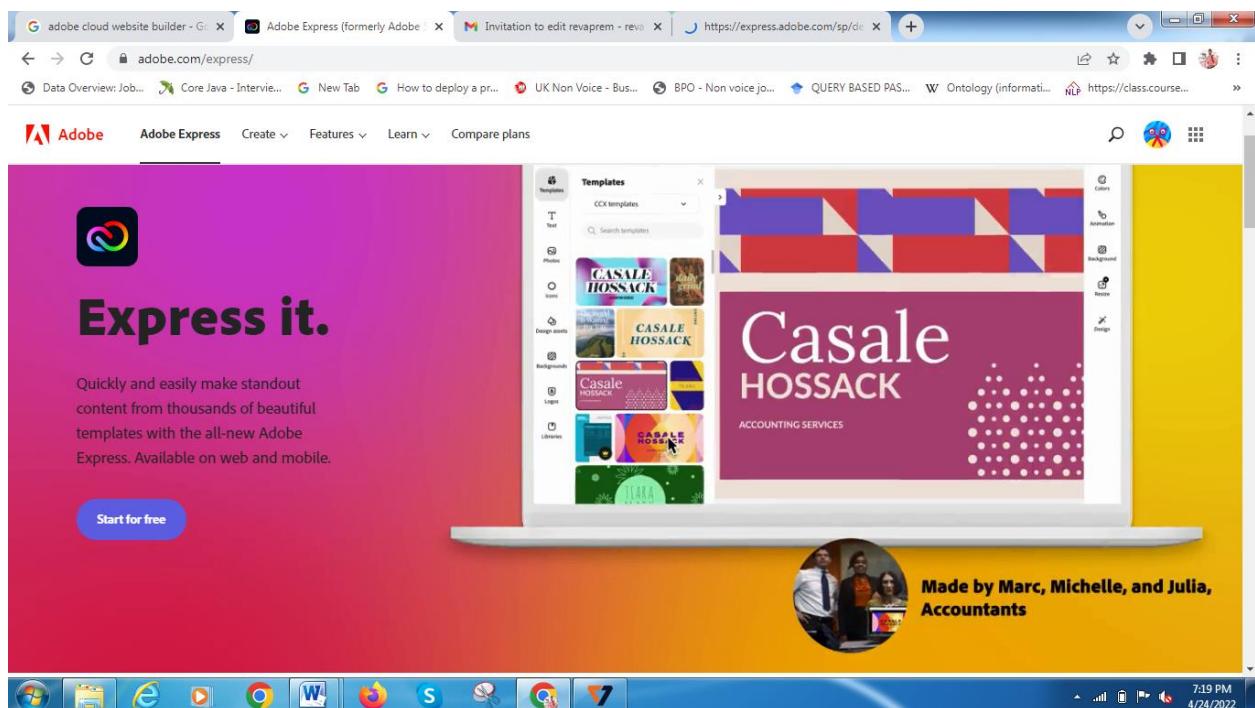
Creative Cloud Express is ideal for projects that don't require more than one page, such as portfolios, resumes, presentations, blog posts, and photo galleries. Creative Cloud Express can showcase a product catalog, advertise a special offer, or act as a weekly or monthly newsletter for businesses.

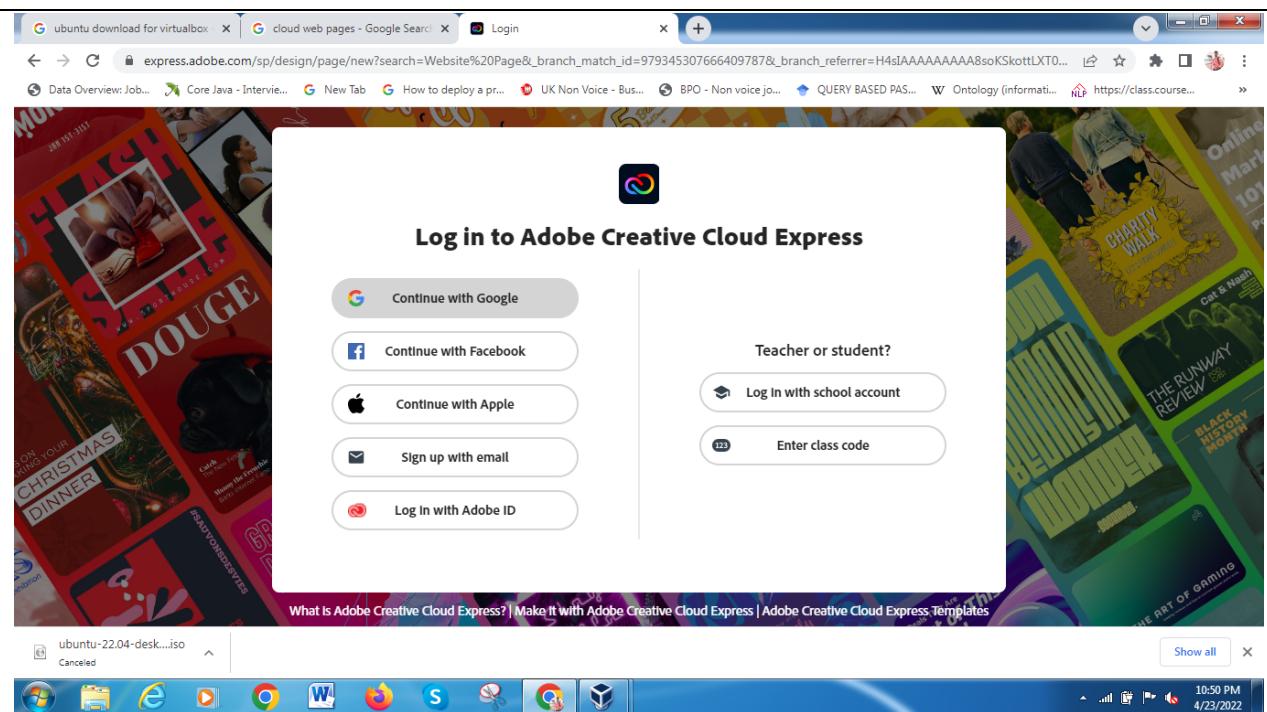
Process/Procedure :

9. Create an account <https://www.adobe.com/express/create/website-page>
10. Pick a theme
11. Choose beautiful images to use
12. Add different elements to your web page.
13. Share your page

Experiment

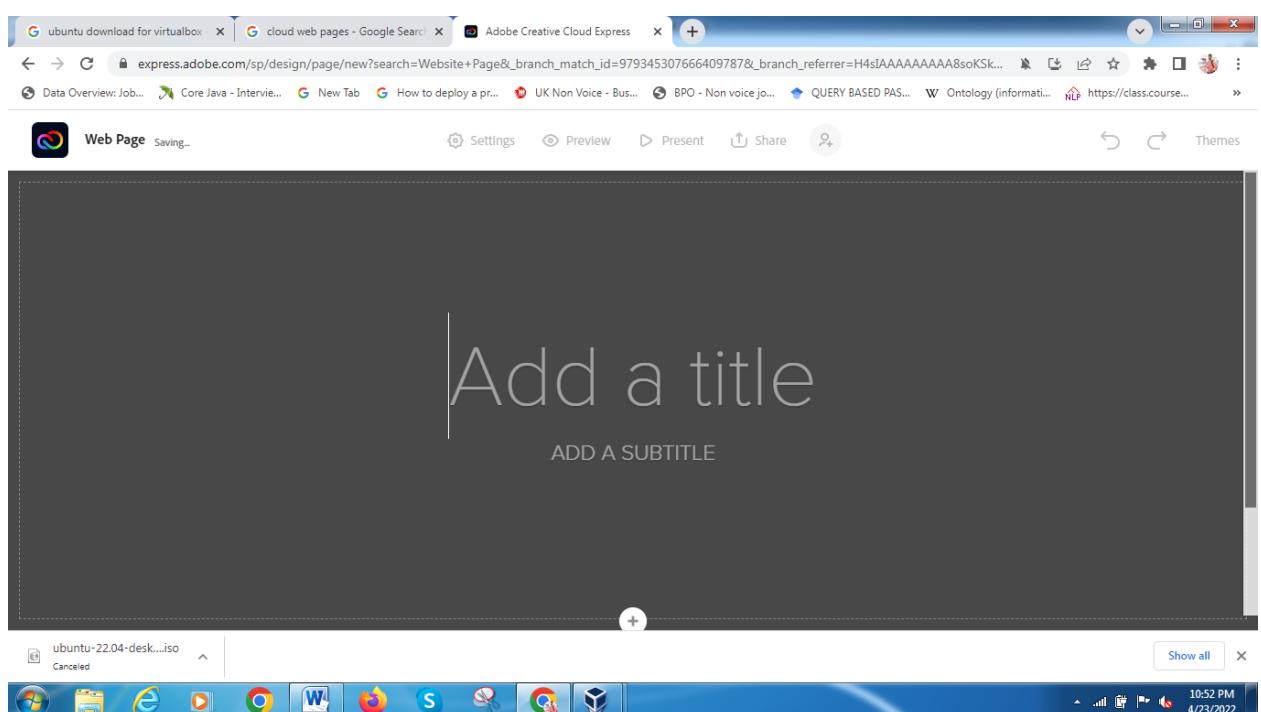
Step 1: Create a free Creative Cloud Express account online or login using our existing accounts



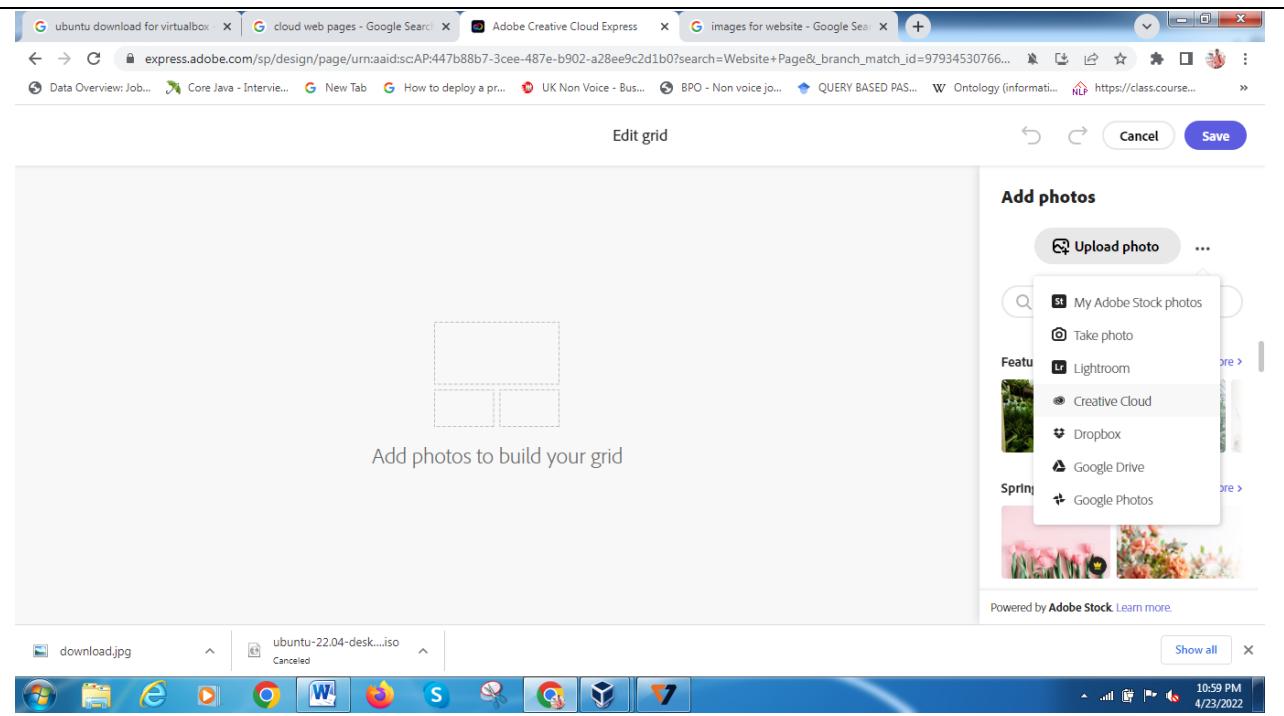


After LOGIN Explore the Following Features of adobe cloud express website design Online Cloud Environment

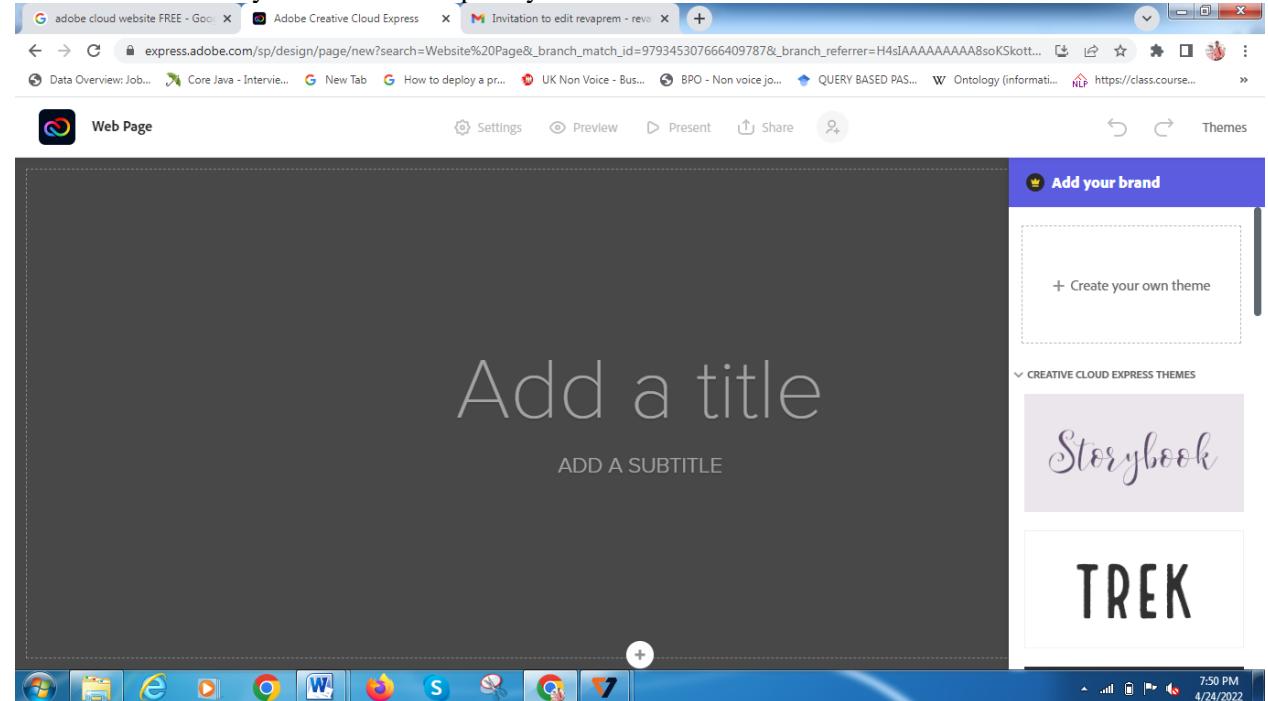
Step 2: Add a title and subtitle for webpage



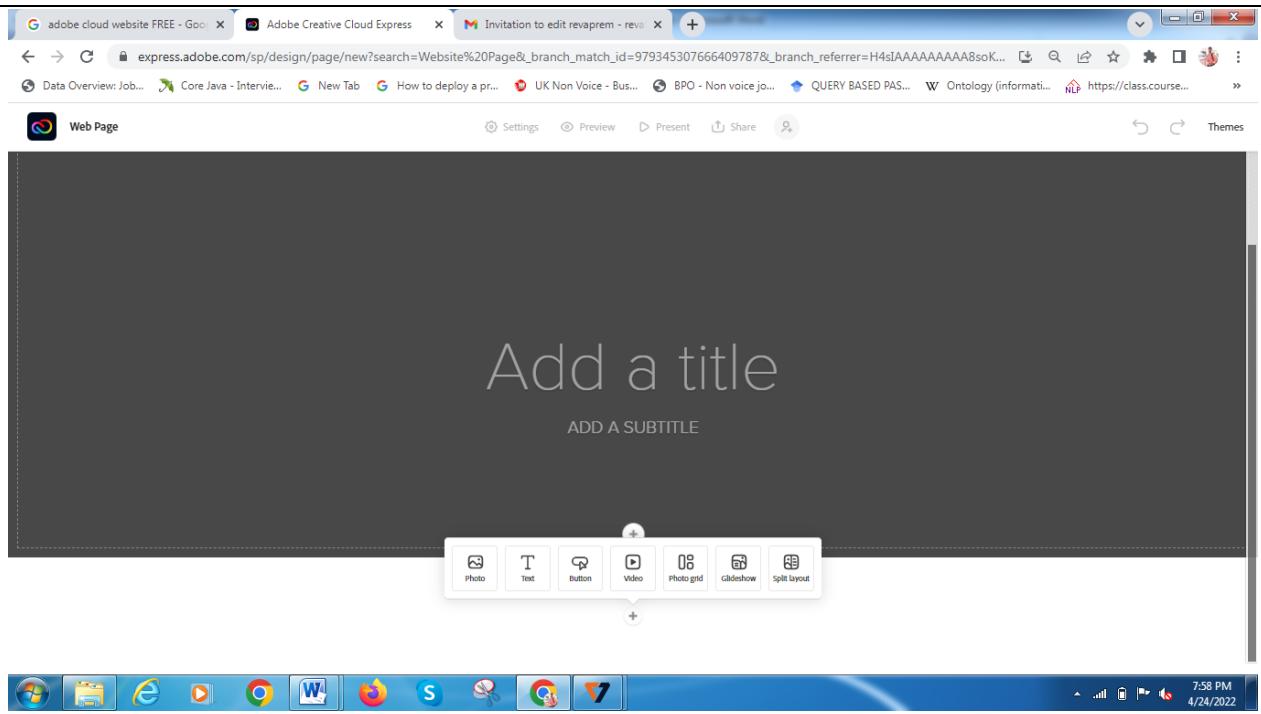
Step 3: Choose a theme from the themes gallery



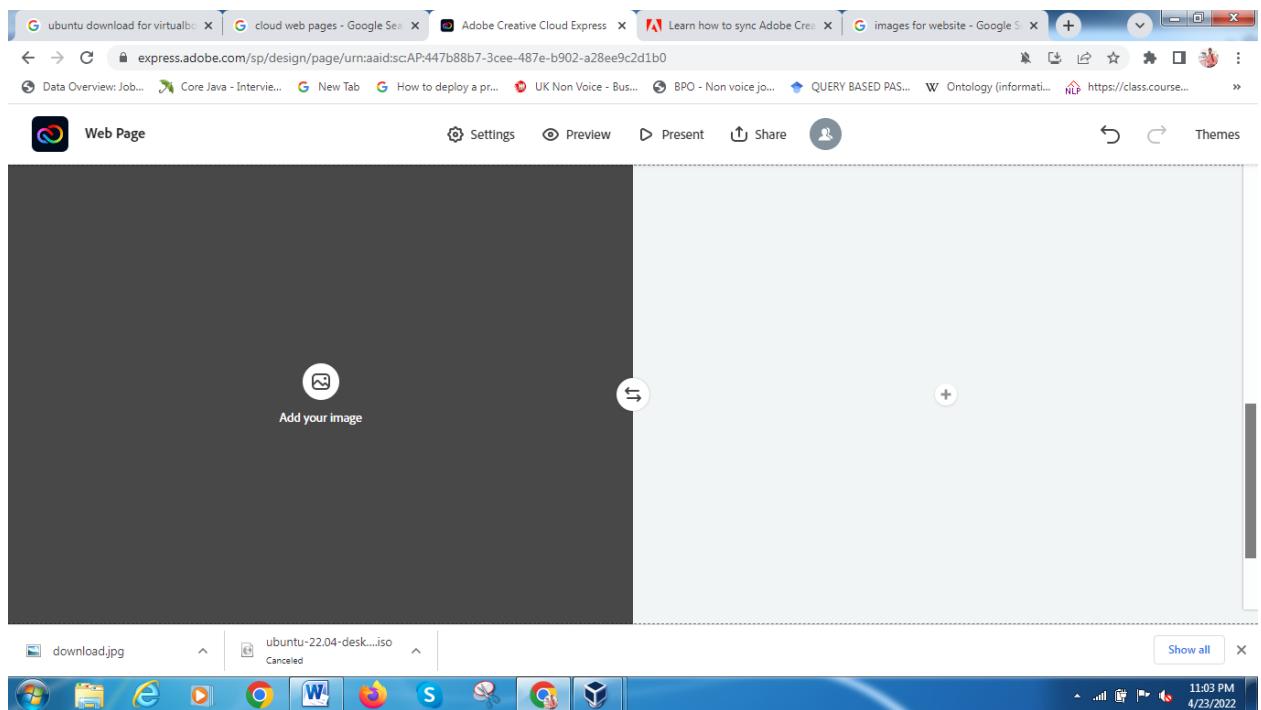
Step 3: To set fonts and styles that will completely transform



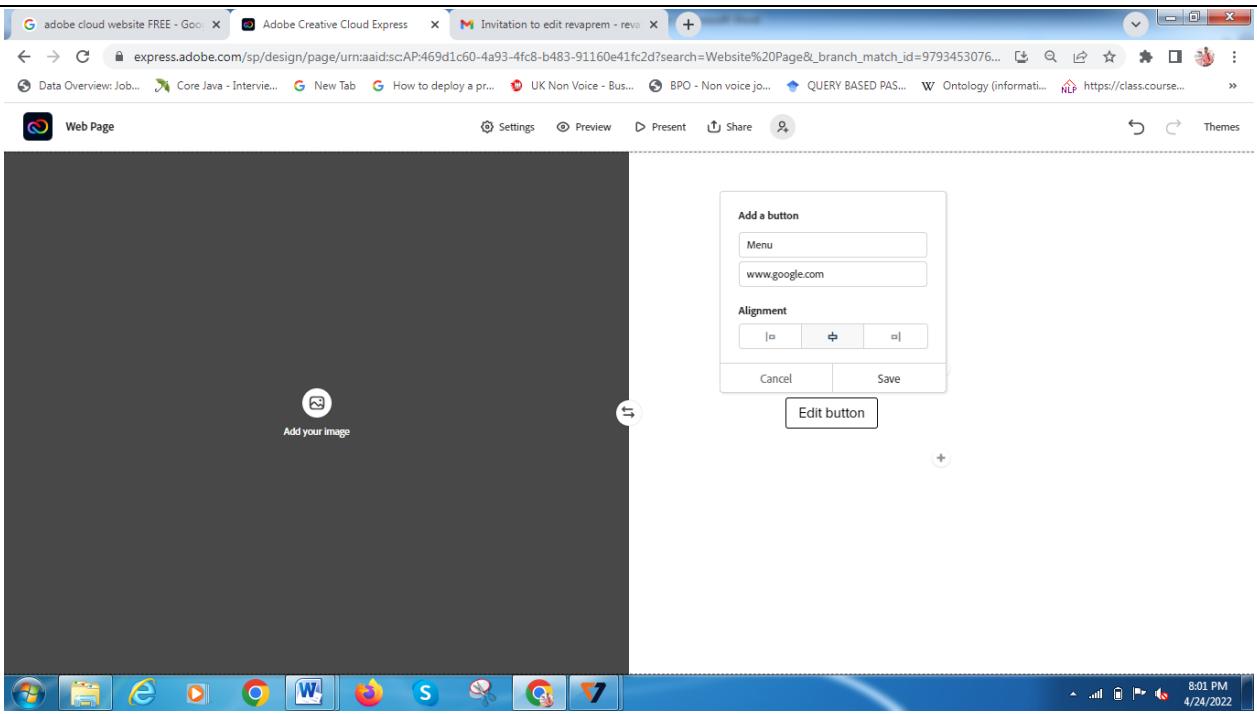
Step 4: Click split layout for two side layout



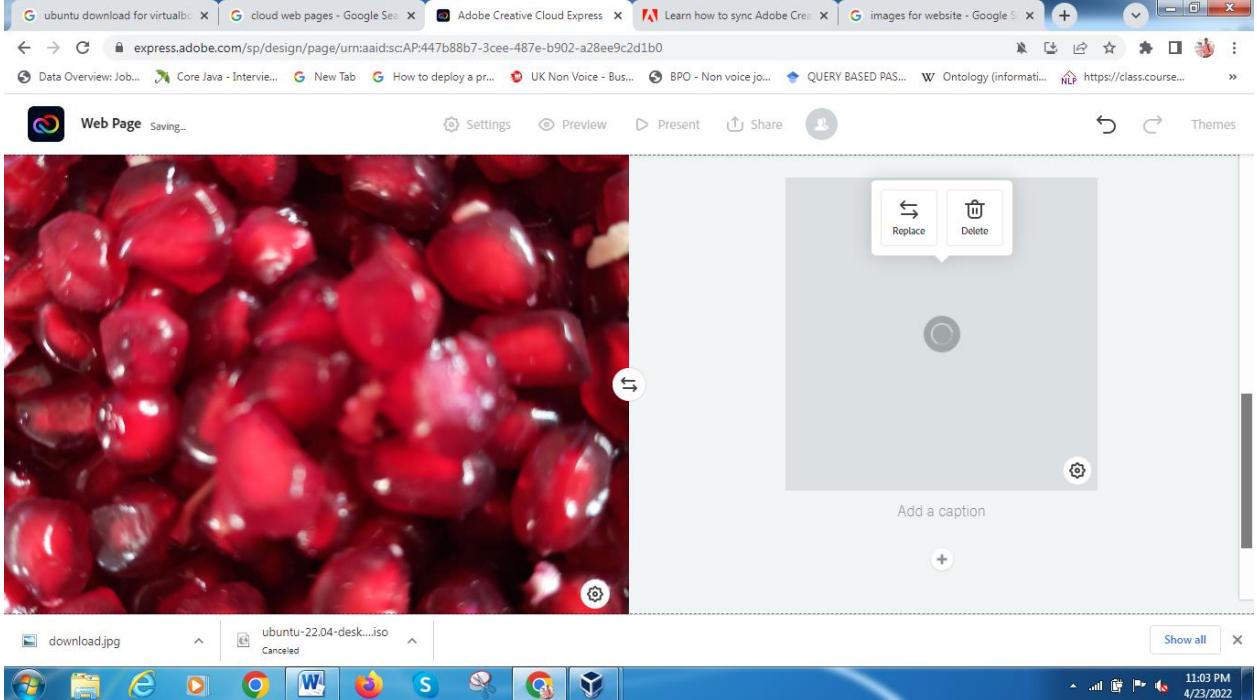
Step 5: Left side add image and right side add a button and name the button



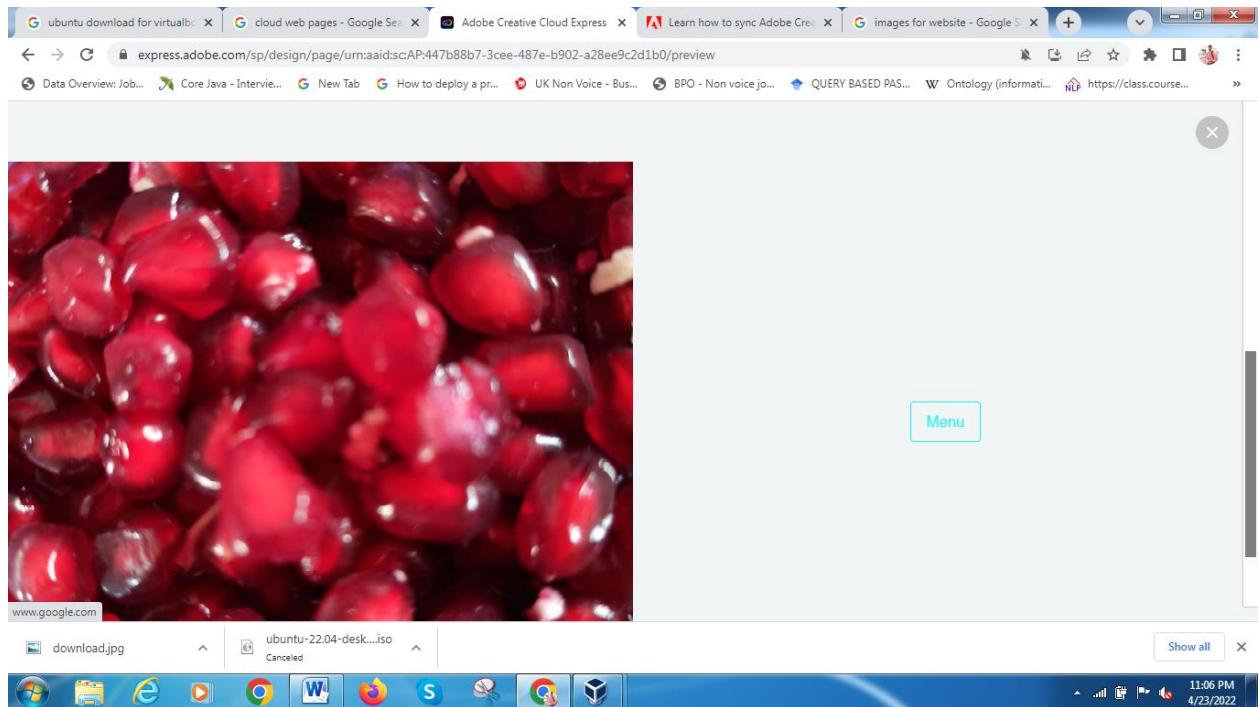
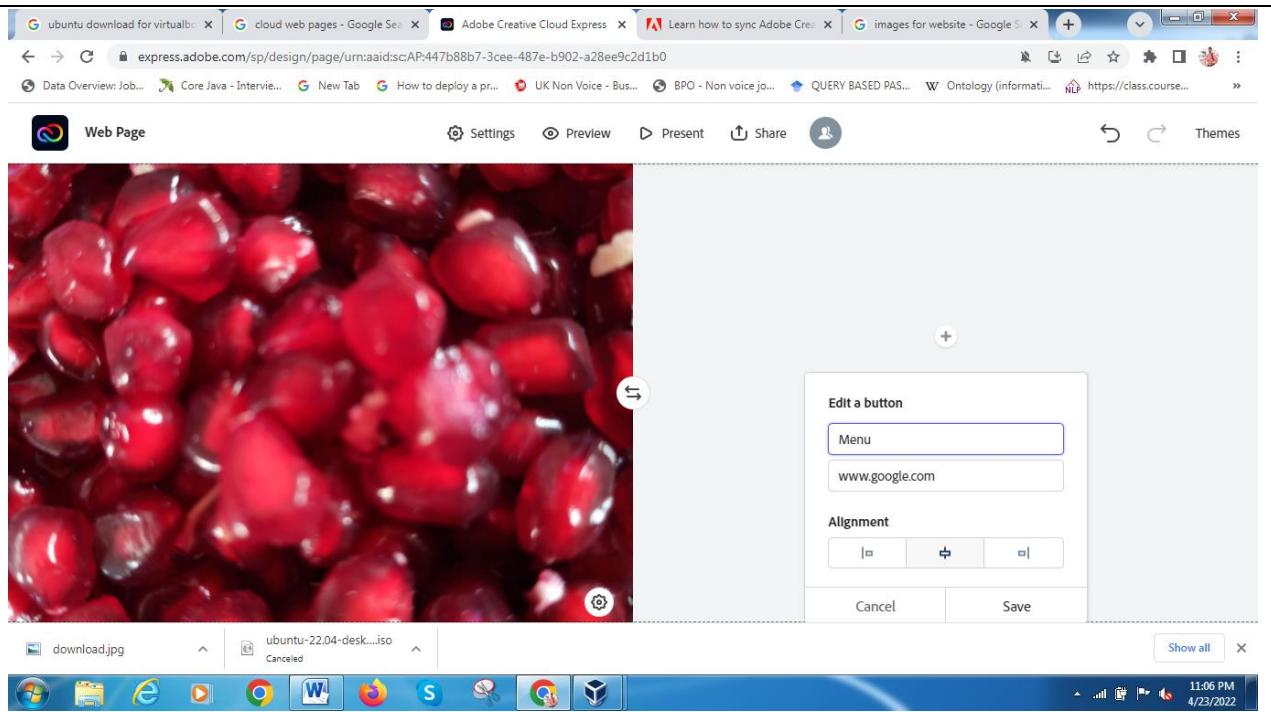
Step 6: Create a button named ourselves and give some webpage.

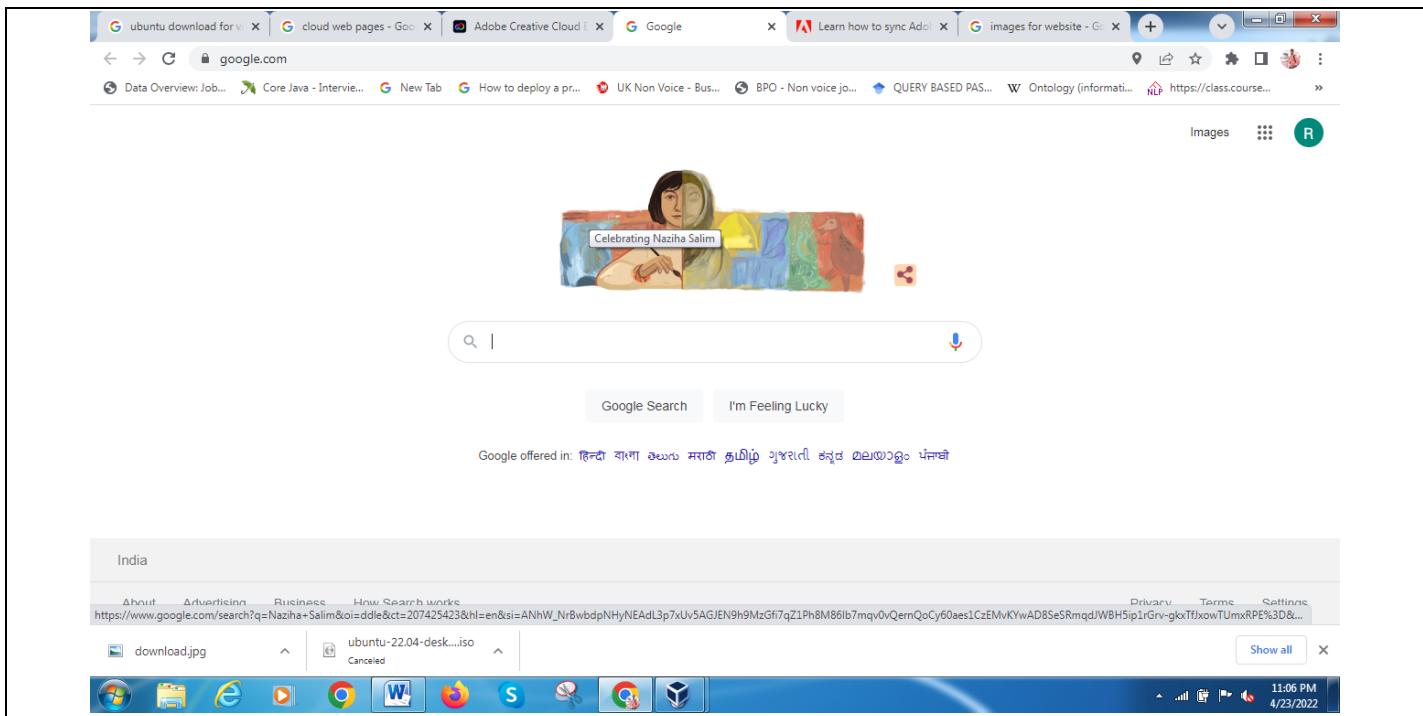


Step 6: Create a button named ourselves and give some webpage.

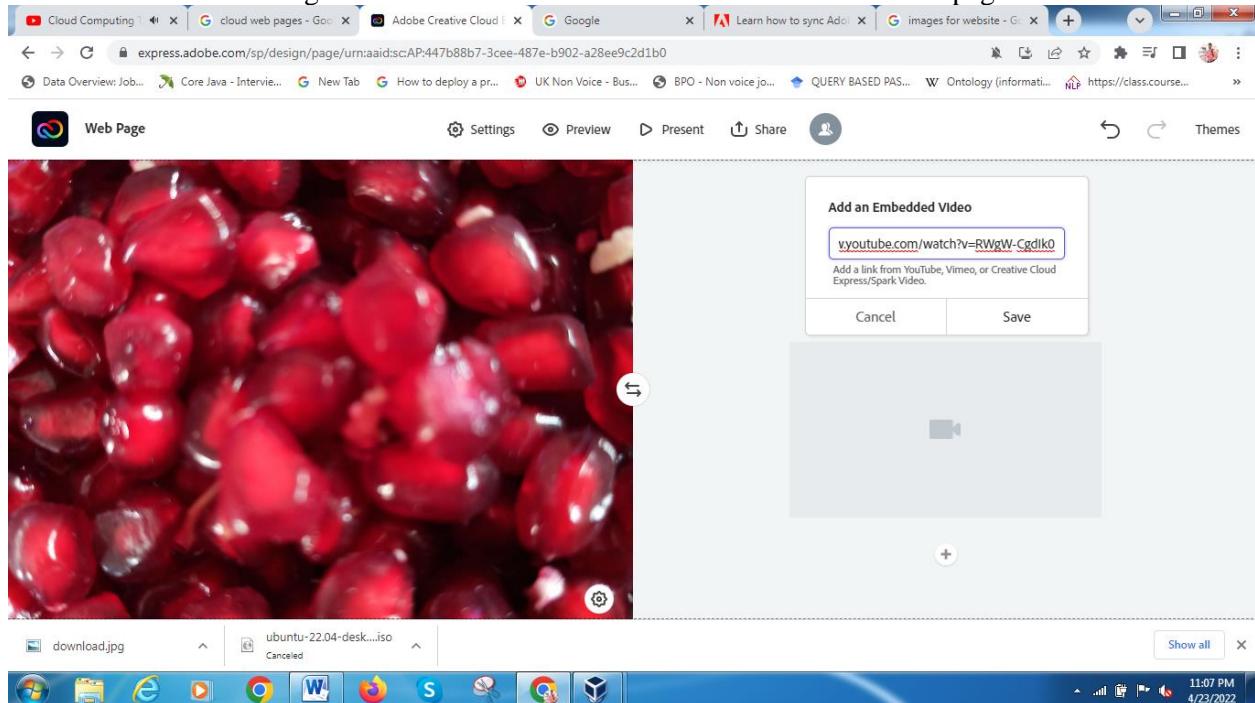


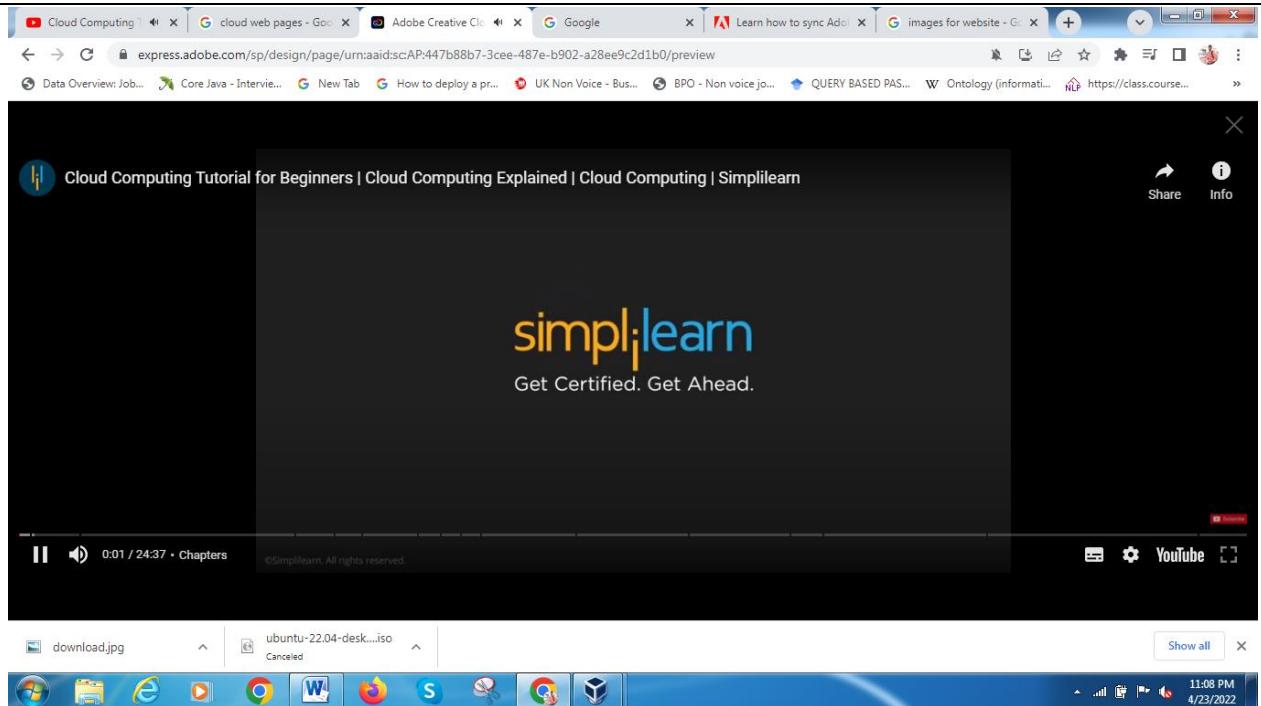
Step 7: Save the button and preview our website design and click the button. It will load the webpage that you added in our page button.





Step 8: create next button and give some video links. It will load the video in our webpage.





Results:

Using the above steps services offered by adobe express website online cloud environment are explored.

Aim:

To explore the features provided by Cloud Resources in Eclipse with Redhat cloud account.

Concept:

Eclipse Che hosted by Red Hat is an open source product based on Eclipse Che that is running on OpenShift Dedicated. The new service is part of the Developer Sandbox for Red Hat OpenShift offering, and is using CodeReady Workspaces, which is built upon Eclipse Che and is optimized for Red Hat OpenShift and Red Hat Linux.

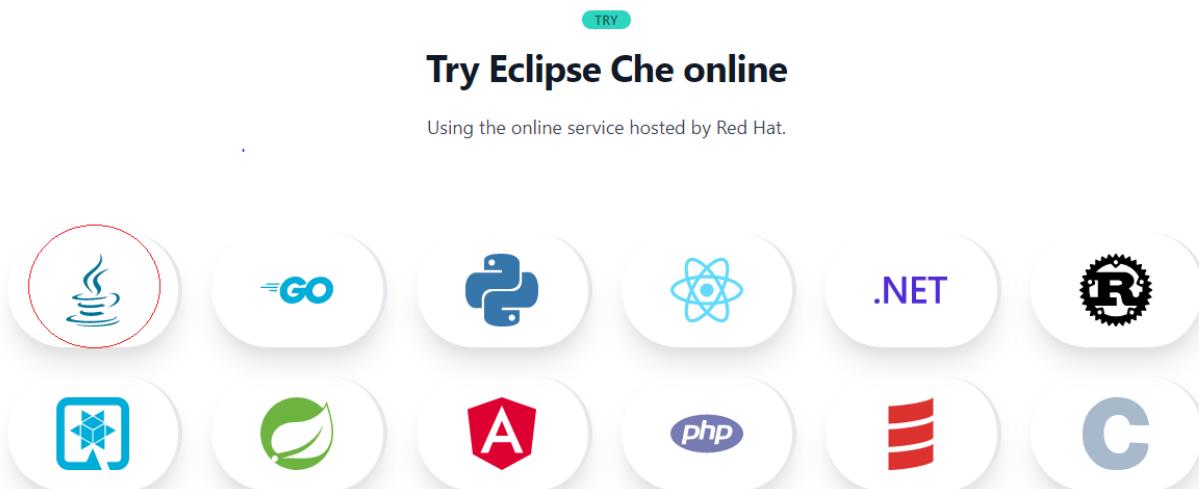
Process/Procedure:

14. Start <https://www.eclipse.org/che/getting-started/cloud/>
15. Try eclipse che online in cloud platform
16. Click any programming language icon.
17. Create an account in redhat sandbox
18. Launch our developer sandbox

Experiment

Step 1: Go to the link <https://www.eclipse.org/che/getting-started/cloud/>

Step 2: Select any programming language.



Step 3: Create an account

lowercase letters, uppercase letters, numbers, and symbols.

I have read and agree to all the terms and conditions below (check all boxes).

* I have read and agree to the Enterprise Agreement [\[link\]](#)

* I have read and agree to the Red Hat Developer Subscription for Individuals [\[link\]](#)

Email opt-in

Receive email notifications of Red Hat Developer services and events, including invitations and reminders.

CREATE MY ACCOUNT

[« Back to login page](#)



Copyright ©2022 Red Hat, Inc. [Privacy statement](#) | [Terms of use](#) | [All policies and guidelines](#)

RED HAT
SUMMIT

Step 4: Launch a sandbox for redhat

Overview Get started in the Sandbox **Sandbox IDE** Use case activities

Start developing directly on OpenShift with CodeReady Workspaces

The Developer Sandbox comes pre-configured with [CodeReady Workspaces](#), Red Hat's productised version of Eclipse Che. You can quickly spin up a development environment with everything you need, all hosted on OpenShift.

Launch a workspace for an application directly from the OpenShift topology viewer to quickly view and make changes to the code!

[Launch your Che Developer Sandbox for Red Hat OpenShift](#)

About the technology

What is the difference between Eclipse Che and CodeReady Workspaces?

CodeReady Workspaces is built upon Eclipse Che and is optimized for Red Hat OpenShift and Red Hat Linux. It is built on RHEL 8. CodeReady Workspaces bundles a curated set of devfiles and plug-ins supported by Red Hat, whereas Eclipse Che has a broader set, including ones that are not supported in the product.

[Learn more about Eclipse Che.](#)

What are the limitations of using CodeReady Workspaces on the Developer Sandbox?

Featured resources

[④ Containerized Python Flask development on Red Hat OpenShift](#)

[④ Getting started with CodeReady Workspaces and Red Hat OpenShift Application Runtimes launcher](#)

[④ Streamline your JBoss EAP dev environment with](#)

Step 5: Fill the details for login

Contact Information

* First name:	<input type="text"/>
* Last name:	<input type="text"/>
* Country or Region:	<input type="text" value="Choose your country"/>
* Address line 1:	<input type="text"/>
Address line 2:	<input type="text"/>
Address line 3:	<input type="text"/>
* Postal code:	<input type="text"/>
* City:	<input type="text"/>
State:	<input type="text"/>

Step 6: Click the developer sandbox

The screenshot shows the Red Hat Developer Sandbox IDE homepage. At the top, there's a navigation bar with links for Data Overview, Core Java - Interview, New Tab, How to deploy a pr..., UK Non Voice - Bus..., BPO - Non voice jo..., QUERY BASED PAS..., Ontology (information), and https://class.course... A banner for the Red Hat Summit is visible, along with a call to action to "Join us at May's virtual event for all our technology content. Register for free →". Below the banner, the Red Hat Developer logo is on the left, and a search bar with "Search All Red Hat" is on the right. The main content area features a heading "Welcome to the Developer Sandbox for Red Hat OpenShift" and a sub-section about accessing Red Hat products without setup or configuration. It includes a screenshot of a terminal window showing system specifications like RAM: 7GB and Storage: 150B. To the right is a graphic illustrating a developer workspace with a laptop, clouds, and code blocks.

Step 7: Verify the mobile number

This screenshot shows the mobile number verification step. It displays a form titled "Confirm your account via text" asking for a "Country Code and Phone Number". A field contains "+91" and another is empty. A "Submitting..." button is visible. Below the form, a note states: "We require a valid phone number to reduce the creation of fraudulent accounts. Red Hat will not use this information for any other reason. Cost may apply for receiving texts if abroad. Please consult with your service provider for details." Navigation links for Overview, Get started in the Sandbox, Sandbox IDE, and Use case activities are at the bottom.

Start developing directly on OpenShift with CodeReady Workspaces

The Developer Sandbox comes pre-configured with CodeReady Workspaces, Red Hat's productised version of Eclipse Che. You can quickly spin up a development environment with everything you need, all hosted on OpenShift.

Launch a workspace for an application directly from the OpenShift topology viewer to quickly view and make changes to the code!

This screenshot shows the mobile number verification step again, with the "Submitting..." button now active. The rest of the page content is identical to the previous screenshot, including the "Start developing directly on OpenShift with CodeReady Workspaces" section and navigation links.

Start developing directly on OpenShift with CodeReady Workspaces

The Developer Sandbox comes pre-configured with CodeReady Workspaces, Red Hat's productised version of Eclipse Che. You can quickly spin up a development environment with everything you need, all hosted on OpenShift.

Launch a workspace for an application directly from the OpenShift topology viewer to quickly view and make changes to the code!

This screenshot shows the mobile number verification step again, with the "Submitting..." button now active. The rest of the page content is identical to the previous screenshots, including the "Start developing directly on OpenShift with CodeReady Workspaces" section and a note about verification codes being sent from a US number.

Step 8: Click the 'start using your sandbox'

Start developing directly on OpenShift with CodeReady Workspaces

The Developer Sandbox comes pre-configured with [CodeReady Workspaces](#), Red Hat's productised version of Eclipse Che. You can quickly spin up a development environment with everything you need, all hosted on OpenShift.

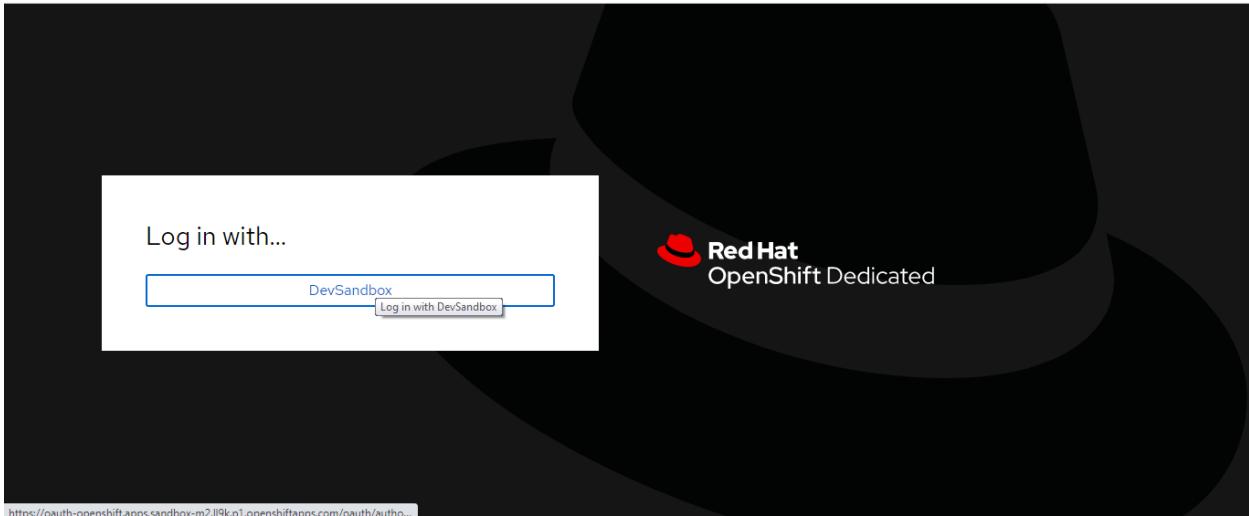
Launch a workspace for an application directly from the OpenShift topology viewer to quickly view and make changes to the code!

You're ready to get started!

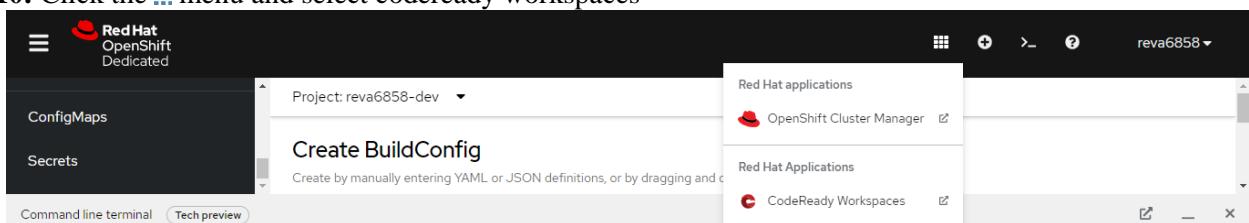
To launch your Che sandbox, click the button below and select DevSandbox when prompted.

[Start using your sandbox](#)

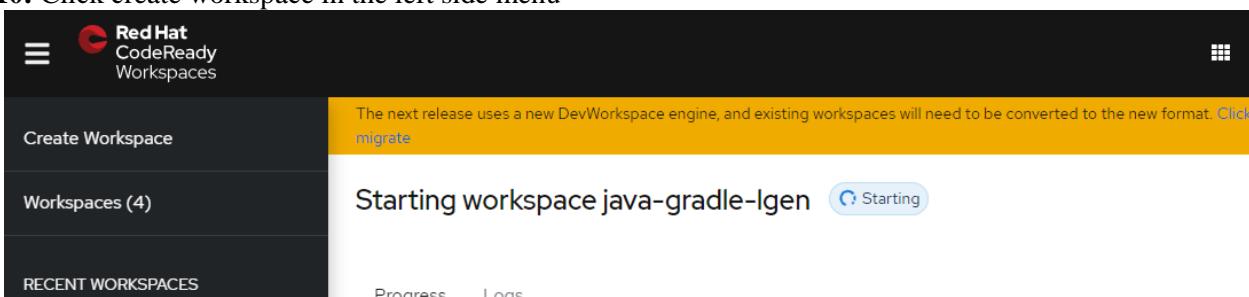
Step 9: Click the DevSandbox



Step 10: Click the ☰ menu and select codeready workspaces



Step 10: Click create workspace in the left side menu



Step 11: Select the C++ language

The screenshot shows the Eclipse IDE interface with the following details:

- Left Sidebar:**
 - Create Workspace**
 - Workspaces (4)**
 - RECENT WORKSPACES**
 - java-gradle-lgen
 - cpp-bvyl
 - java-mongo-e5yc
 - java-gradle-kifm
- Workspace Options:**
 - .NET stack with .NET Core SDK 3.1.103, Runtime, C# Language Support and Debugger
 - Stack with .Net 3.1
 - Stack for developing ASP.NET Core Web Application
- Community Workspaces:**
 - Apache Camel K** (Community)
 - Stack with tooling ready to develop Integration projects with Apache Camel K
 - Apache Camel based on Spring Boot** (Community)
 - Stack with environment ready to develop Integration projects with Apache Camel based on Spring Boot.
 - Bash** (Community)
 - Stack with environment ready to develop bash scripts.
- Tech-Preview Workspaces:**
 - C++** (Tech-Preview)
 - C/C++
 - C and C++ Developer Tools stack with GCC, cmake and make
 - C/C++** (Community)
 - C/C++
 - Stack with C/C++ and Clang 8
 - Go** (Tech-Preview)
 - Stack with Go

Results:

Using the above steps services offered by Eclipse with Redhat cloud environment is explored.

Aim:

To explore the features provided by Eclipse with Redhat cloud account and create Password using C++ Program Language.

Concept:

Eclipse Che hosted by Red Hat is an open source product based on Eclipse Che that is running on OpenShift Dedicated. The new service is part of the Developer Sandbox for Red Hat OpenShift offering, and is using CodeReady Workspaces, which is built upon Eclipse Che and is optimized for Red Hat OpenShift and Red Hat Linux.

Process/Procedure:

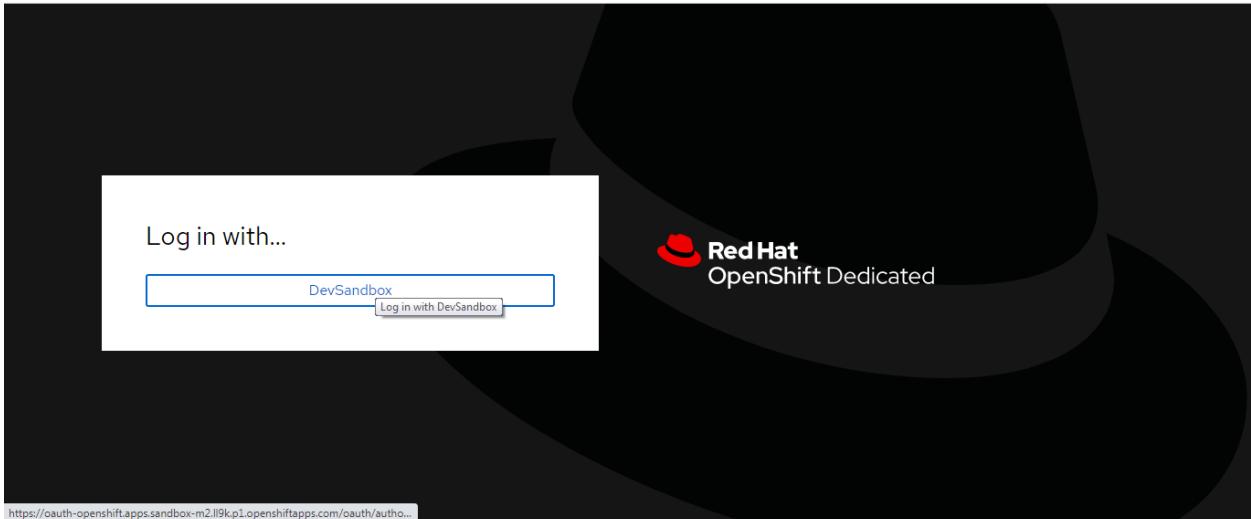
19. Start <https://www.eclipse.org/che/getting-started/cloud/>
20. Try eclipse che online in cloud platform
21. Click any programming language icon.
22. Create an account in redhat sandbox
23. Launch our developer sandbox

Experiment

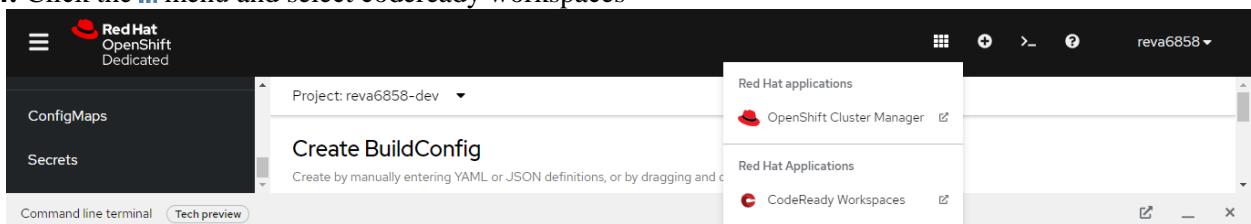
Step 1: Go to the link <https://www.eclipse.org/che/getting-started/cloud/>

Step 2: Select any programming language.

Step 3: Click the DevSandbox



Step 4: Click the menu and select codeready workspaces



Step 5: Click create workspace in the left side menu

The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. [Click here to migrate](#)

[Create Workspace](#)
[Workspaces \(4\)](#)
RECENT WORKSPACES
Step 6: Select the C++ language

Create Workspace	.NET stack with .NET Core SDK 3.1.103, Runtime, C# Language Support and Debugger	Stack with .Net 3.1	Stack for developing ASP.NET Core Web Application
Workspaces (4)			
RECENT WORKSPACES			
<input checked="" type="radio"/> java-gradle-lgen <input type="radio"/> cpp-bvyl <input type="radio"/> java-mongo-e5yc <input type="radio"/> java-gradle-kifm	 Apache Camel K Stack with tooling ready to develop Integration projects with Apache Camel K	 Apache Camel based on Spring Boot Stack with environment ready to develop Integration projects with Apache Camel based on Spring Boot.	 Bash Stack with environment ready to develop bash scripts.
	 C/C++ C and C++ Developer Tools stack with GCC, cmake and make	 C/C++ Stack with C/C++ and Clang 8	 Go Stack with Go

[Create Workspace](#)
[Workspaces \(5\)](#)
RECENT WORKSPACES
 [cpp-kl48](#)
 [java-gradle-lgen](#)
 [cpp-bvyl](#)
 [java-mongo-e5yc](#)
 [java-gradle-kifm](#)

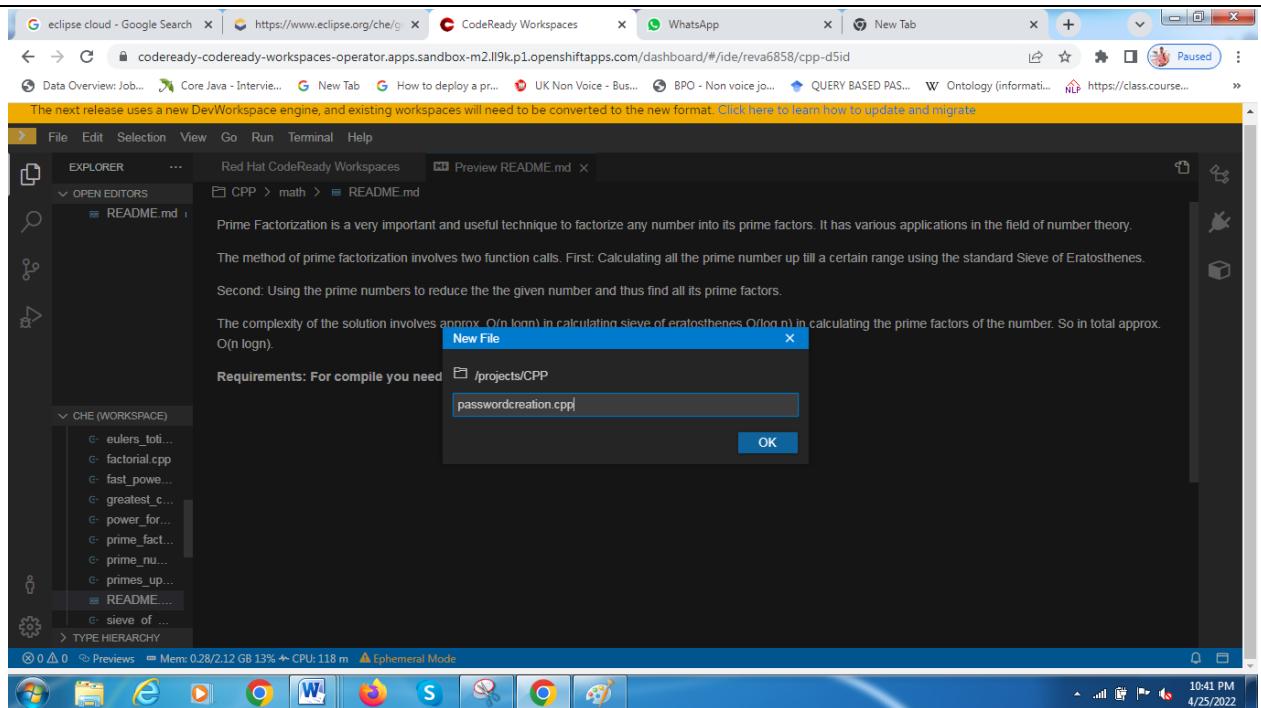
The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. [Click here to migrate](#)

Starting workspace cpp-kl48 
[Progress](#)
[Logs](#)
 1 Initializing

 2 Waiting for workspace to start

3 Open IDE

Step 7: Go to File menu → select new → Name the file name (.cpp)



Step 8: Type the password creation C++ program

```
#include<iostream>
#include<string.h>
using namespace std;
int main()
{
    char pass[20], ePass[20];
    int numOne, numTwo, sum;
    cout<<"Create a Password: ";
    cin>>pass;
    cout<<"\nEnter Two Numbers to Add: ";
    cin>>numOne>>numTwo;
    cout<<"\nEnter the Password to See the Result: ";
    cin>>ePass;
    if(!strcmp(pass, ePass))
    {
        sum = numOne + numTwo;
        cout<<endl<<numOne<<" + "<<numTwo<<" = "<<sum;
    }
    else
        cout<<"\nSorry! You've entered a Wrong Password!";
    cout<<endl;
    return 0;
}
```

Step 9: Run the program using terminal

The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. Click here to learn how to update and migrate.

File Edit Selection View Go Run Terminal Help

EXPLORER OPEN EDITORS passwordcreation.cpp

CPP > Run Task... Run Build Task... Run Test Task... Run Last Task Ctrl+Shift+K

Show Running Tasks... Restart Running Task... Terminate Task... Attach Task...

Configure Tasks... Run Selected Text

```

1 #inc
2 #inc
3 usir
4 int
5 [
6
7
8
9
10
11
12
13
14 if(!strcmp(pass, ePass))
15 {
16     sum = numOne + numTwo;
17     cout<<endl<<numOne<<" + "<<numTwo<<" = "<<sum;
18 }
19 else
20     cout<<"\nSorry! You've entered a Wrong Password!";
21 cout<<endl;
22 return 0;
23 ]

```

Ln 23, Col 2 LF UTF-8 Spaces: 4 C++ 10:42 PM 4/25/2022

Step 10: Select the cpp-dev

The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. Click here to learn how to update and migrate.

File Edit Selection View Go Run Terminal I Select container to create new terminal

EXPLORER OPEN EDITORS passwordcreation.cpp

CPP > password creation.cpp

Developer Container Tooling Containers

```

1 #include<iost
2 #include<str
3 using namespac
4 int main()
5 [
6     char pass[ ]
7     int numOne, numTwo, sum;
8     cout<<"Create a Password: ";
9     cin>>pass;
10    cout<<"\nEnter Two Numbers to Add: ";
11    cin>>numOne>>numTwo;
12    cout<<"\nEnter the Password to See the Result: ";
13    cin>>ePass;
14    if(strcmp(pass, ePass))
15    {
16        sum = numOne + numTwo;
17        cout<<endl<<numOne<<" + "<<numTwo<<" = "<<sum;
18    }
19    else
20        cout<<"\nSorry! You've entered a Wrong Password!";
21    cout<<endl;
22    return 0;
23 ]

```

Ln 23, Col 2 LF UTF-8 Spaces: 4 C++ 10:42 PM 4/25/2022

Step 11: Type the command “make filename”

```

Problems      [jboss@workspace49c8v38h2g2451c8 CPP]$ make passwordcreation
[jboss@workspace49c8v38h2g2451c8 CPP]$ 

```

Step 12: Run the program using “./filename”

```

Problems      [jboss@workspace49c8v38h2g2451c8 CPP]$ ./passwordcreation
[jboss@workspace49c8v38h2g2451c8 CPP]$ 

```

OUTPUT:

```
[jboss@workspace49c8v38h2g2451c8 CPP]$ ./passwordcreation
Create a Password: 1234

Enter Two Numbers to Add: 1 5

Enter the Password to See the Result: 1234

1 + 5 = 6
[jboss@workspace49c8v38h2g2451c8 CPP]$ █
```

Results:

Using the above steps services offered by Eclipse with Redhat cloud environment for develop and run the C++ Program.

Aim:

To explore the features provided by Eclipse with Redhat cloud account and develop the java program for pattern creation.

Concept:

Eclipse Che hosted by Red Hat is an open source product based on Eclipse Che that is running on OpenShift Dedicated. The new service is part of the Developer Sandbox for Red Hat OpenShift offering, and is using CodeReady Workspaces, which is built upon Eclipse Che and is optimized for Red Hat OpenShift and Red Hat Linux.

Process/Procedure:

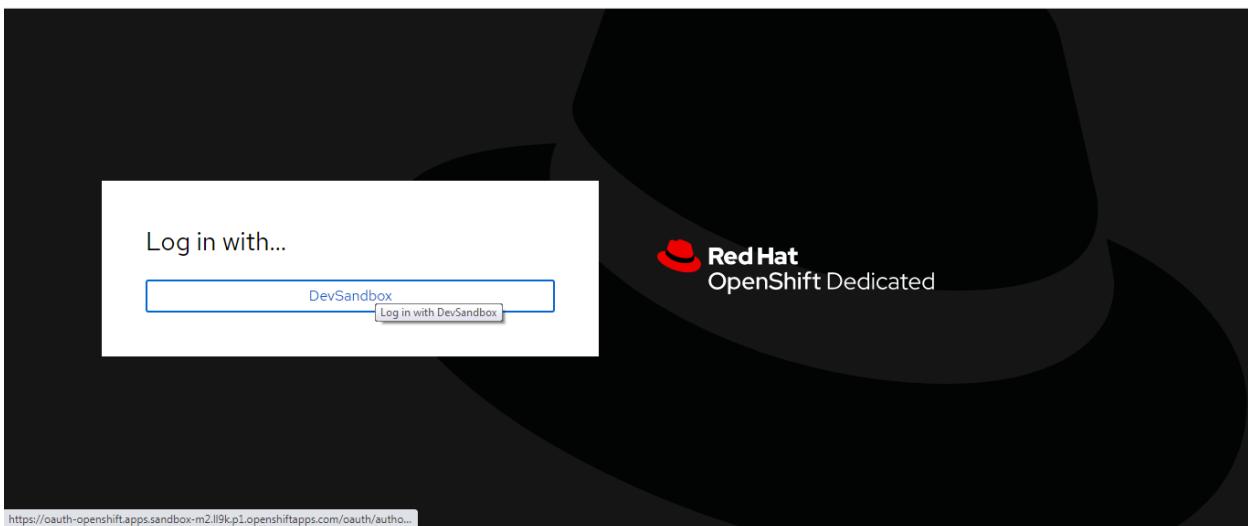
24. Start <https://www.eclipse.org/che/getting-started/cloud/>
25. Try eclipse che online in cloud platform
26. Click any programming language icon.
27. Create an account in redhat sandbox
28. Launch our developer sandbox
29. Create and run the Java Program.

Experiment

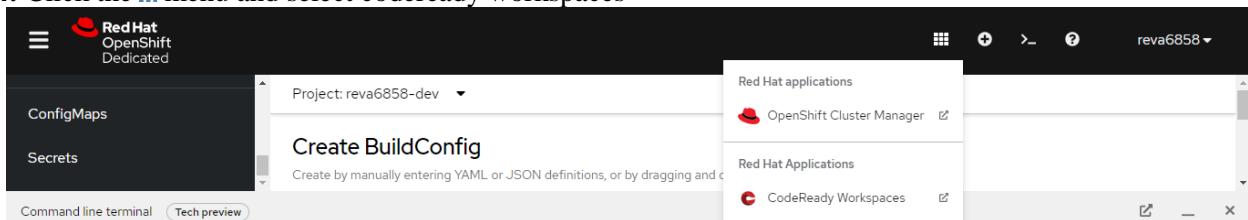
Step 1: Go to the link <https://www.eclipse.org/che/getting-started/cloud/>

Step 2: Select any programming language.

Step 3: Click the DevSandbox



Step 4: Click the menu and select codeready workspaces



Step 5: Click create workspace in the left side menu

The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. [Click to migrate](#)

Starting workspace **java-gradle-lgen** (Starting)

Progress Logs

Step 6: Select the Java language

Stack with Go 1.14 IntelliJ IDEA Community Edition based editor stack Java stack with OpenJDK 11, Maven 3.6.3, and Gradle 6.1

Java Java Lombok Java Lombok

Java Gradle Java stack with OpenJDK 11 and Gradle 6.2.1 Java stack with OpenJDK 11, Maven 3.6.3 and Lombok 1.18.18

X Java Maven QUARKUS

Java Maven Java Quarkus

Java stack with OpenJDK 11, Maven 3.6.3 and Vert.x demo Java stack with OpenJDK 11, Maven 3.6.0 Java stack with OpenJDK 11, Maven 3.6.3, Gradle 6.1 and Quarkus Tools

The next release uses a new DevWorkspace engine, and existing workspaces will need to be converted to the new format. [Click to migrate](#)

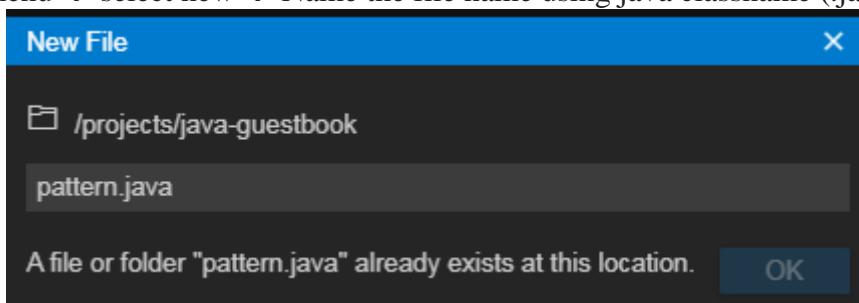
Starting workspace **cpp-kl48** (Starting)

Progress Logs

1 Initializing
2 Waiting for workspace to start
3 Open IDE

cpp-kl48
java-gradle-lgen
cpp-bvyl
java-mongo-e5yc java-mongo-e5yc
java-gradle-k1fm

Step 7: Go to File menu → select new → Name the file name using java classname (.java)



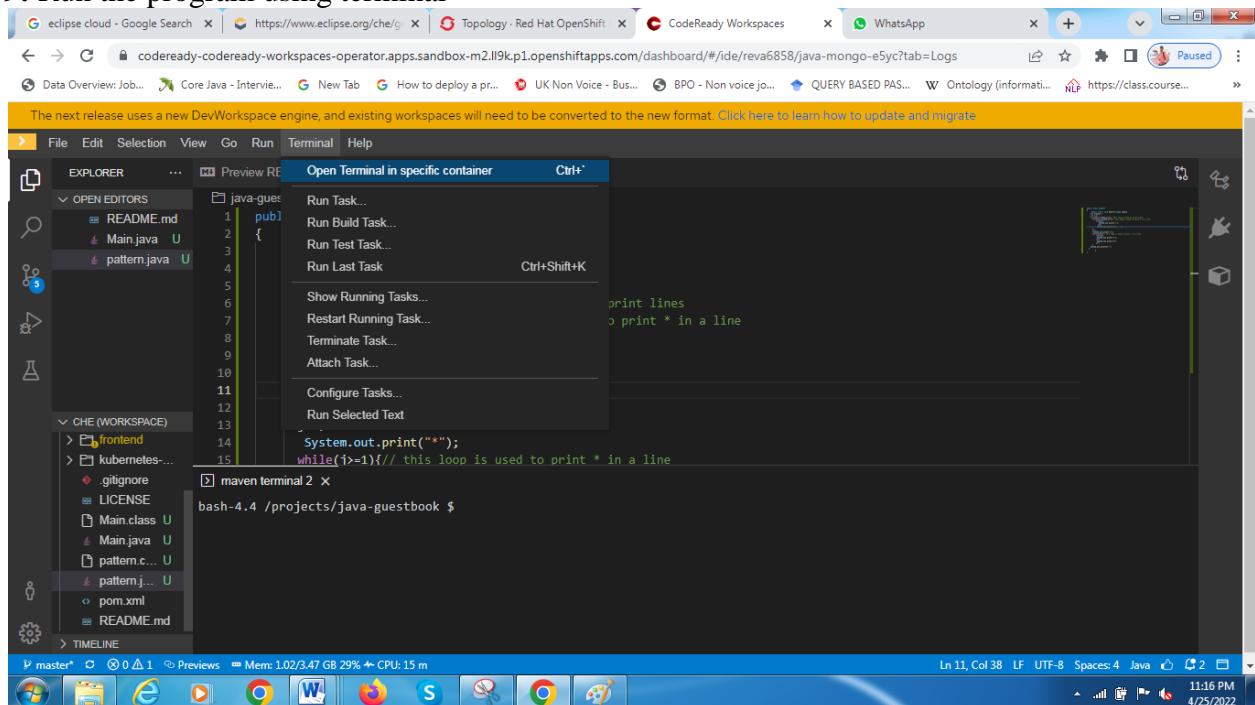
Step 8: Type the pattern creation java program

```

public class pattern
{
    public static void main(String[] args){
        int lines=4;
        int i,j;
        for(i=1;i<=lines;i++){// this loop is used to print lines
            for(j=1;j<=lines;j++){// this loop is used to print * in a line
                if(i==j)
                    System.out.print("*");
                else
                    System.out.print("0");
            }
            j--;
            System.out.print("*");
        while(j>=1){// this loop is used to print * in a line
            if(i==j)
                System.out.print("*");
            else
                System.out.print("0");
            j--;
        }
        System.out.println("");
    }
}
}

```

Step 9: Run the program using terminal



Step 10: Select the mango developer

Step 11: Type the compile command “javac filename.java”

```

bash-4.4 /projects/java-guestbook $ javac pattern.java
bash-4.4 /projects/java-guestbook $ java pattern

```

Step 12: Run the program using “java filename”

OUTPUT:

```
bash-4.4 /projects/java-guestbook $ java pattern
*000*000*
0*00*00*0
00*0*0*00
000***000
```

Results:

Using the above steps services offered by Eclipse with Redhat cloud environment for developed and run the Java Program.