**Slip no .-5**

**Q. Working of Google drive to make spreadsheet and notes.**

**Create a Spreadsheet and Fill It With Data**

The best part about Google Sheets is that it's free and it works on any device—which makes it easy to follow along with the tutorials in this book. All you'll need is a web browser (or the [Google Sheets app](https://www.google.com/sheets/about/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier#section-download) on your iOS or Android device), and a free Google account. On your Mac or PC, head over to [sheets.google.com](https://sheets.google.com/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier), and you're ready to get started.

There are 3 ways to create a new spreadsheet in Google Sheets:

1. Click the red "NEW" button on your your [Google Drive dashboard](https://drive.google.com/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier) and select "Google Sheets"
2. Open the menu from within a spreadsheet and select "File > New Spreadsheet"
3. Click "Blank" or select a template on the [Google Sheets](https://sheets.google.com/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier) homepage

This will create a new blank spreadsheet (or a pre-populated template if you choose one of those). For this tutorial, though, you should start with a blank spreadsheet.

Look around the white-and-grey grid that occupies most of your screen, and the first thing you’ll notice is a blue outline around the selected cell or cells.

As soon as you open a new spreadsheet, if you just start typing you’ll see that your data starts populating the selected cell immediately—usually the top left cell. There's no need to double click cells when you add information, and not much need to use your mouse.

Feel free to select any cell you’d like, then go ahead and type something in. When you’re done entering data into a cell, you can do one of 4 things:

1. Press **ENTER** to save the data and move to the **beginning** of the next row
2. Press **TAB** to save the data and move **to the right** in the same row
3. Use the **ARROW KEYS** on your keyboard (up, down, left, and right) to **move 1 cell in that direction**
4. **Click any cell** to jump **directly to that cell**

If you don’t want to type in everything manually, you can also add data to your Sheet *en masse* via a few different methods:

1. Copy and paste a list of text or numbers into your spreadsheet
2. Copy and paste an HTML table from a website
3. Import an existing spreadsheet in csv, xls, xlsx and [other formats](https://support.google.com/docs/answer/40608?hl=en&utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier&utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier&utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier)
4. Copy any value in a cell across a range of cells via a *click and drag*

**Slip No.6**

**Q.** **Working and Implementation of identity and Access management.**

##### **STEP 1: ASSESS YOUR CURRENT IAM SITUATION.**

* Take inventory of your current cloud versus on-premises applications
* Estimate the extent of non-sanctioned apps deployed, i.e. “Shadow IT”
* Take inventory of your end-user access preferences

Your first step is to review the gaps within your current [IAM (Identity and Access Management)](https://www.onelogin.com/learn/iam) environment and use that information to define your future requirements. Take inventory of your current policies and infrastructure so you can better understand the costs and benefits of a new IAM solution.

##### **STEP 2: EVALUATE WHAT IAM APPROACH IS RIGHT FOR YOU.**

* Think about what drives your strategy
* Consider security, productivity, compliance concerns
* Cloud vs. on-premises deployments
* Out-of-box ready vs. custom deployment
* Open standards vs. proprietary interfaces, e.g. SAML, SCIM, NAPPS
* Review the cost benefits of different solutions, e.g. both money and rollout time

Once you’ve answered the questions from step one, you’ll have most of what you need to start researching solutions.

##### **STEP 3: DEFINE A STRATEGY FOR IMPLEMENTING YOUR IAM PLAN.**

* Assemble key stakeholders
* Define a cloud vendor onboarding certification policy (CVOC)
* Define deployment plan
  + Requirement
  + Dependencies
  + Milestones
  + Timelines
  + Metrics
* Implement IAM solution
* Gain user acceptance

The key to a successful implementation includes engaging the right stakeholders early, driving toward achievable milestones supporting early successes, and then expanding the reach and scope of your solution. Stakeholders might include representatives from your line of business, security, network, compliance, and human resource teams.

**Slip No. 7**

**Q. Write a program for web feed.**

### 1) Define the source, by the channel tag

### The channel will be the same for all your RSS feeds. These tags are required: - title: the title of your website, may be the one in the title tag of the home page. - link: the URL of your website: example: https://www.xul.fr

### 2) Add an image

This is optional. Design a small image (88x31 for example) in a common format (gif, jpg, png) and put in into the same directory that the RSS file.  
The "image" tag is a sub-element of the channel tag  
- **url** is the address of the image itself.  
- **link** is the address of the page displayed when one clicks on the image.

### 3) Add a new

Now, we will add a web page to display an information. This is an "item" tag, a sub-element of channel, and this components are required:  
- **title**: the title of the article.  
- **link**: the URL of the page.

### 4) Add more items

More items may be added to this channel.

### 5) Upload the file

Put the **feed.xml**file into your website, among other web pages.

### 6) Validate the file

You may use this online [**RSS feed validator**](http://www.feedvalidator.org/).

### 7) Make it available

This is accomplished by adding an RSS button on the home page. A click on the button should display the XML file you have created

### 8) Updating the feed

To publish further articles, you have just to add items, and remove older ones to keep the number of articles constant.

<?xml version="1.0" ?>

<rss version="2.0">

<channel>

  <title>Ajax and XUL</title>

  <link>[**https://www.xul.fr/en/**](https://www.xul.fr/en/)</link>

  <description>XML graphical interface etc...</description>

  <image>

      <url>[**https://www.xul.fr/xul-icon.gif**](https://www.xul.fr/xul-icon.gif)</url>

      <link>[**https://www.xul.fr/en/index.php**](https://www.xul.fr/en/index.php)</link>

  </image>

  <item>

      <title>News  of today</title>

      <link>[**https://www.xul.fr/en-xml-rss.html**](https://www.xul.fr/en-xml-rss.html)</link>

      <description>All you need to know about RSS</description>

  </item>

  <item>

      <title>News of tomorrows</title>

      <link>[**https://www.xul.fr/en-xml-rdf.html**](https://www.xul.fr/en-xml-rdf.html)</link>

      <description>And now, all about RDF</description>

    </item>

</channel>

</rss>

**Slip No. 8**

**Q. Execute the step to Demonstrate and implementation of cloud on single sign on.**

You can implement Single Sign On in your **custom applications** that use Auth0 by just flipping a switch in each app.

1. In the management dashboard, click [Apps / APIs](https://manage.auth0.com/#/applications).
2. Click the application that you want to enable Single Sign On.
3. In the **Settings** tab, scroll down until you see the **Use Auth0 instead of the IdP to do Single Sign On** switch.
4. Flip the switch! and save the changes.

Steps are as Follows:

The login flow usually looks like this:

1. A user browses to the application or website they want access to, aka, the Service Provider.
2. The Service Provider sends a token that contains some information about the user, like their email address, to the SSO system, aka, the Identity Provider, as part of a request to authenticate the user.
3. The Identity Provider first checks to see whether the user has already been authenticated, in which case it will grant the user access to the Service Provider application and skip to step 5.
4. If the user hasn’t logged in, they will be prompted to do so by providing the credentials required by the Identity Provider. This could simply be a username and password or it might include some other form of authentication like a [One-Time Password (OTP)](https://www.onelogin.com/learn/otp-totp-hotp).
5. Once the Identity Provider validates the credentials provided, it will send a token back to the Service Provider confirming a successful authentication.
6. This token is passed through the user’s browser to the Service Provider.
7. The token that is received by the Service Provider is validated according to the trust relationship that was set up between the Service Provider and the Identity Provider during the initial configuration.
8. The user is granted access to the Service Provider.

**Slip No. 10**

**Q. Installing and Developing Application Using Google App Engine.**

* Install the [Google Cloud SDK](https://cloud.google.com/sdk) command-line tool.
* Install and configure the App Engine plugin for Maven.  
  App Engine also provides plugins for [Gradle](https://cloud.google.com/appengine/docs/standard/java/using-gradle), [IntelliJ IDEA](https://cloud.google.com/code/docs/intellij/create-standard" \t "_blank), and [Eclipse](https://cloud.google.com/eclipse/docs), but this guide demonstrates the Maven plugin.
* Create a new App Engine project.
* Give permission to Cloud Build to enable building and deploying your app.

## Installing required SDKs

1. If you haven't already installed Java SE 8 SDK, [install the Java SE 8 Development Kit (JDK)](http://www.oracle.com/technetwork/java/javase/downloads/index.html).
2. Install and initialize the latest version of the Google Cloud SDK.

[Install and Initialize Google Cloud SDK](https://cloud.google.com/sdk/docs)

1. Install the App Engine Java component:

gcloud components install app-engine-java

1. Authorize your user account:

gcloud auth application-default login

## Configuring Maven in your development environment

1. [Download](http://maven.apache.org/download.cgi) Maven 3.5 or later from the Maven website.
2. [Install](http://maven.apache.org/install.html) Maven 3.5 or later on your local machine.

## Creating a new App Engine project

To create an App Engine App:

1. Create a new directory.
2. Initialize the new project in the created directory:

 mvn archetype:generate -Dappengine-version=1.9.59 -Djava8=true -DCloudSDK\_Tooling=true -Dapplication-id=your-app-id -Dfilter=com.google.appengine.archetypes:

* 1. Set -Dappengine-version to the most recent version of the App Engine SDK for Java, and application-id to the ID of your Cloud project.
  2. Set -Djava8=true to deploy the project in the Java 8 runtime.
  3. Set -DCloudSDK\_Tooling=true to use Cloud SDK tooling.

1. When prompted to choose an archetype, choose the value 2 for the App Engine skeleton archetype. This creates an empty project that contains the required directory structure and files.
2. When prompted for version, press ENTER to select the default most recent version.
3. When prompted to Define value for property 'groupId', supply the desired namespace for your app; for example, com.mycompany.myapp.
4. When prompted to Define value for property 'artifactId', supply the project name; for example, myapp.
5. When prompted to Define value for property 'version', accept the default value.
6. When prompted to Define value for property 'package', supply your preferred package name (or accept the default). The generated Java files will have the package name you specify here.
7. When prompted to confirm your choices, accept the default value (Y).
8. Wait for the project to finish generating, then change directories to the new project directory, for example myapp/.
9. Add the Cloud SDK-based plugin to your project's pom.xml file.

<build>  
    <plugins>  
       [...]  
          <plugin>  
            <groupId>com.google.cloud.tools</groupId>  
            <artifactId>appengine-maven-plugin</artifactId>  
            <version>2.2.0</version>  
            <configuration>  
                <projectId>your-project-ID-goes-here</projectId>  
                <version>1</version>  
            </configuration>  
          </plugin>  
       [...]  
    </plugins>  
</build>

1. Specify the target Google Cloud Platform project ID and the [service and version](https://cloud.google.com/appengine/docs/standard/java/an-overview-of-app-engine#versions_and_instances) in the plugin configuration.

**Slip No. 13**

**Q. Implementation of Cloud Failure Cluster.**

## Setting up failover clustering

### Reserve an IP address for the cluster in Compute Engine

When you create the failover cluster, you assign an IP address to create an administrative access point. In a production environment, you might use an IP address from a separate subnet. However, in this tutorial you reserve an IP address from the subnet you already created. Reserving the IP address prevents conflicts with other IP assignments.

1. Open a terminal on a host VM or open Cloud Shell.

[Go to Cloud Shell](https://console.cloud.google.com/?cloudshell=true)

1. Reserve an IP address. For this tutorial, use 10.0.0.8:

gcloud compute addresses create cluster-access-point --region [YOUR\_REGION] --subnet wsfcnetsub1 --addresses 10.0.0.8

1. To confirm the reservation of the IP address:

gcloud compute addresses list

### Create the cluster

To create and configure the failover cluster:

1. Use RDP to connect wsfc-1 and wsfc-2.
2. Follow the steps in the Microsoft instructions below, with these additional notes:
   * Install the Failover Clustering feature on wsfc-1 and wsfc-2. Don't install the Failover Clustering feature on wsfc-dc.
   * Run the Failover Cluster Manager app as the domain user WSFC.TEST\cluster-admin. Otherwise, you might encounter permissions issues. It's a good idea to always run Failover Cluster Manager this way or to connect to a server as cluster-admin to ensure you have the required permissions.
   * Add wsfc-1 and wsfc-2 to the cluster as nodes.
   * When validating the configuration:
     + On the **Testing Options**​ page, select **Run only tests I select**​, ​and then click **Next​**.
     + On the **Test Selection**​ page, clear **Storage** because the **Storage​** option will fail when running on Compute Engine (as it would for separate standalone physical servers).

Common issues you might encounter during cluster validation include:

* + - * **Only one network interface between replicas**. You can ignore this one, because it doesn't apply in a cloud-based setup.
      * **Windows Updates not the same on both replicas**. If you configured your Windows instances to apply updates automatically, one of the nodes might have applied updates that the other hasn't downloaded yet. You should keep the servers in identical configurations.
      * **Pending reboot**. You've made changes to one of the servers, and it needs a reboot to apply. Don't ignore this one.
      * **The servers do not all have the same domain role**. You can ignore this one.
      * **The servers are not all in the same Organizational Unit (OU)**. This tutorial doesn't use an OU at all, but in a production system consider putting your cluster in its own OU. The Microsoft instructions describe this best practice.
      * **Unsigned drivers were found**. You can ignore this one.
  + On the **Summary** page, you can select **Create the cluster now using the validated nodes** to continue on to create the cluster, rather than closing the wizard and reopening it.
  + In the Create Cluster Wizard, on the **Access point** page:
    - Name your cluster "testcluster".
    - In the **Address** field, enter the IP address that you reserved earlier, 10.0.0.8

**Slip No. 14**

**Q. Managing and working of cloud xen server.**

Citrix Hypervisor is an industry leading, open-source platform for cost-effective desktop, server, and cloud virtualization infrastructures. Citrix Hypervisor enables organizations of any size or type to consolidate and transform compute resources into virtual workloads for today’s data center requirements. Meanwhile, it ensures a seamless pathway for moving workloads to the cloud.

The key features of Citrix Hypervisor are:

* Consolidating multiple virtual machines (VMs) onto a physical server
* Reducing the number of separate disk images to be managed
* Allowing for easy integration with existing networking and storage infrastructures
* Enabling you to schedule zero downtime maintenance by live migrating VMs between Citrix Hypervisor hosts
* Assuring availability of VMs by using high availability to configure policies that restart VMs on another server in case one fails
* Increasing portability of VM images, as one VM image works on a range of deployment infrastructures
* The Xen Project hypervisor is an open-source type-1 or bare-metal hypervisor. It allows many instances of an operating system or different operating systems to run in parallel on a single machine (or host). Xen hypervisor is used as the basis for many different commercial and open-source applications, such as: server virtualization, Infrastructure as a Service (IaaS), desktop virtualization, security applications, embedded, and hardware appliances.
* Citrix Hypervisor is based on the Xen Project hypervisor, with extra features and supports provided by Citrix. Citrix Hypervisor 8.2 uses version 4.13 of the Xen hypervisor.

### Control domain

* The **Control Domain**, also called Domain 0, or dom0, is a secure, privileged Linux VM that runs the Citrix Hypervisor management toolstack known as XAPI. This Linux VM is based on a CentOS 7.5 distribution. Besides providing Citrix Hypervisor management functions, dom0 also runs the physical device drivers for networking, storage, and so on. The control domain can talk to the hypervisor to instruct it to start or stop guest VMs.

### Guest domain (VMs)

Guest domains are user-created virtual machines that request resources from dom0. For a detailed list of the supported distributions, see [**Supported Guests, Virtual Memory, and Disk Size Limits**](https://docs.citrix.com/en-us/citrix-hypervisor/system-requirements/guest-os-support.html).

## Key concepts

### Resource pool

Citrix Hypervisor allows you to manage multiple servers and their connected shared storage as a single entity by using resource pools. Resource pools enable you to move and run virtual machines on different Citrix Hypervisor hosts. They also allow all servers to share a common framework for network and storage. A pool can contain up to 64 servers running the same version of Citrix Hypervisor software, at the same patch level, and with broadly compatible hardware. For more information, see [**Hosts and resource pools**](https://docs.citrix.com/en-us/citrix-hypervisor/hosts-pools.html).

### Storage repository

Citrix Hypervisor storage targets are called storage repositories (SRs). A storage repository stores Virtual Disk Images (VDIs), which contains the contents of a virtual disk. SRs are flexible, with built-in support for SATA, SCSI, NVMe, and SAS drives that are locally connected, and iSCSI, NFS, SAS, SMB3, and Fibre Channel remotely connected. The SR and VDI abstractions allow advanced storage features such as thin provisioning, VDI snapshots, and fast cloning to be exposed on storage targets that support them.