



Oracle Autonomous Transaction Processing Hands-on Lab Guide



Oracle Autonomous Transaction Processing Hands on Lab Guide

Lab Introduction	4
<i>Lab software prerequisites</i>	4
Section 1 - Introduction to Autonomous Transaction Processing	9
Lab 1. Connecting to the Oracle Cloud	10
<i>Connecting to your Lab Virtual Machine</i>	11
<i>Signing in to the OCI Console</i>	12
<i>Locating your ATP Instance</i>	14
Lab 2. Securely Connecting to Autonomous Transaction Processing	19
<i>Managing Priorities on Autonomous Transaction Processing</i>	20
<i>Downloading the credentials wallet</i>	21
<i>Connecting to the database using SQL Developer</i>	24
<i>(Optional) Secure access to your Autonomous Database using Access Control Lists</i>	26
Section 2 - Focus on DBAs.....	37
Lab 3. Scaling an Automatic Transaction Processing Instance	38
<i>Scaling your instance using the console</i>	39
<i>Scaling the database using REST</i>	42
<i>Uploading the Key to Oracle Cloud Infrastructure</i>	43
<i>Execute the script to scale the database</i>	47
Lab 4. Managing and Monitoring Autonomous Transaction Processing	51
<i>About Backup and Recovery on Autonomous Transaction Processing</i>	51
<i>Exploring Service Console</i>	52
<i>Stopping your ATP instance</i>	57
<i>Starting your ATP instance</i>	58
Lab 5. Data Loading into Autonomous Transaction Processing	61
<i>Loading a file from local storage</i>	61
<i>Loading a file from Object Storage</i>	66
<i>Importing a Data Pump Export</i>	70
Lab 6. DBA Exploration of ATP with SQL Developer.....	76
Lab 7: (Optional) Exploring Auto-Scaling using Swingbench.....	87
<i>Prepare the workload</i>	87
<i>Run the workload with Auto Scaling Disabled</i>	89
<i>Enable Auto Scaling</i>	92
<i>Run the workload with Auto Scaling enabled</i>	94
<i>Disable Auto Scaling in your Database</i>	97
<i>Want to learn more about Swingbench?</i>	99
Section 3- Focus on Developers	100
Microservices on Kubernetes and Autonomous Database	102
Want to Learn More?	103
<i>Recommended Reading</i>	103
<i>Additional Resources</i>	103
Appendix A – Provisioning Process Walkthrough	105
<i>Creating your Autonomous Transaction Processing Database</i>	105
Appendix B – Creating and preparing a user to access Object Storage	112
Appendix C – Connecting to the database using Oracle Machine Learning Notebook	115

Oracle Autonomous Transaction Processing Hands-on Lab Guide

<i>Creating an OML user</i>	115
Appendix D – Configuring Oracle Cloud Infrastructure for REST	122
<i>Create the keys.....</i>	122
<i>Uploading the Key to Oracle Cloud Infrastructure.....</i>	122
<i>Download and customize the script</i>	124

Lab Introduction

In this hands-on lab you will get first-hand experience with Oracle's new Autonomous Transaction Processing (ATP) service. Oracle ATP delivers a self-driving, self-securing, self-repairing database service that can instantly scale to meet demands of mission critical transaction processing and mixed workload applications.

The lab is structured in three sections

- Section 1 – Introduction to ATP
- Section 2 – Focus on DBA activities
- Section 3 – Focus on Developer features

The audience for this hands-on-lab is expected to be database administrators and application developers. The lab expects you to understand the basic concepts of an Oracle database, the UNIX command line and be comfortable using the vi editor or the built in gedit editor.

All attendees must complete the 'Introduction to ATP' section, but then can select labs from the other sections.

Lab software prerequisites

This lab requires you to install the following desktop application so that you can complete this hands-on lab:

- An PDF File reader.
- A VNC viewer. There are many VNC viewer packages both commercial and open source. Suggested packages are TigerVNC viewer or TightVNC Viewer but if you already have a preferred VNC viewer you can use this. TigerVNC viewer has a simpler install process, as it is a standalone executable, but has fewer features.

Some pdf viewers may have trouble correctly copying and pasting the format required for some of the scripts within this guide. All long SQL statements and scripts can be found in the **/home/oracle/labScripts** directory of the supplied VM. Where appropriate the script or SQL filename will be provided by the output of its listing within the guide.

Installing TigerVNC Viewer

Follow these steps to if you want to download and install TigerVNC Viewer.

macOS

A better VNC viewer for macOS is realVNC which can be obtained from
<https://www.realvnc.com/en/connect/download/viewer/macos/>

Windows 64-bit

Download '[vncviewer64-1.9.0.exe](#)' from
<https://bintray.com/tigervnc/stable/tigervnc/1.9.0#files> and save it to your desktop.
It is a self-contained executable file, which requires no further installation.

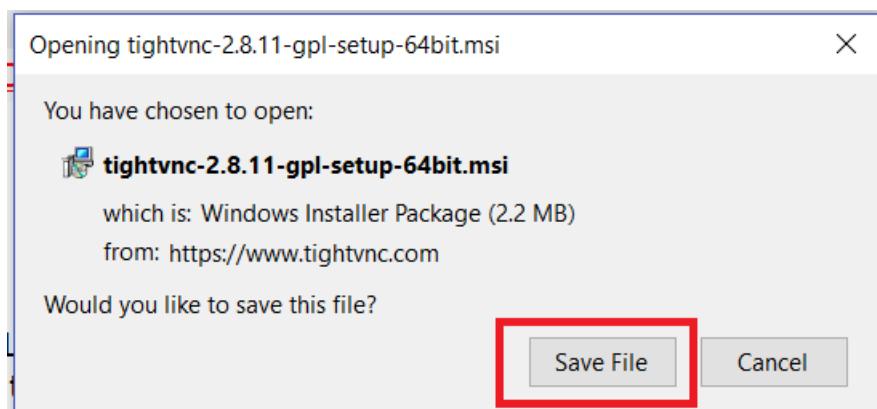
Installing TightVNC Viewer

Follow these steps if you want to download and install TightVNC Viewer.

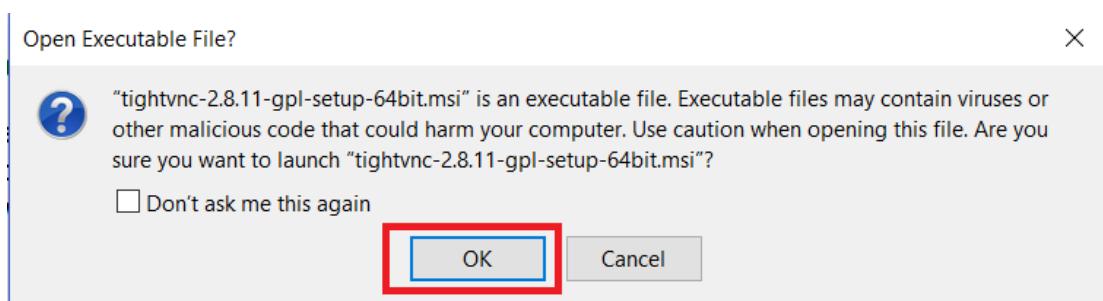
Windows 64-bit

Select the 'Installer for Windows (64-bit)' from
<https://www.tightvnc.com/download.php>

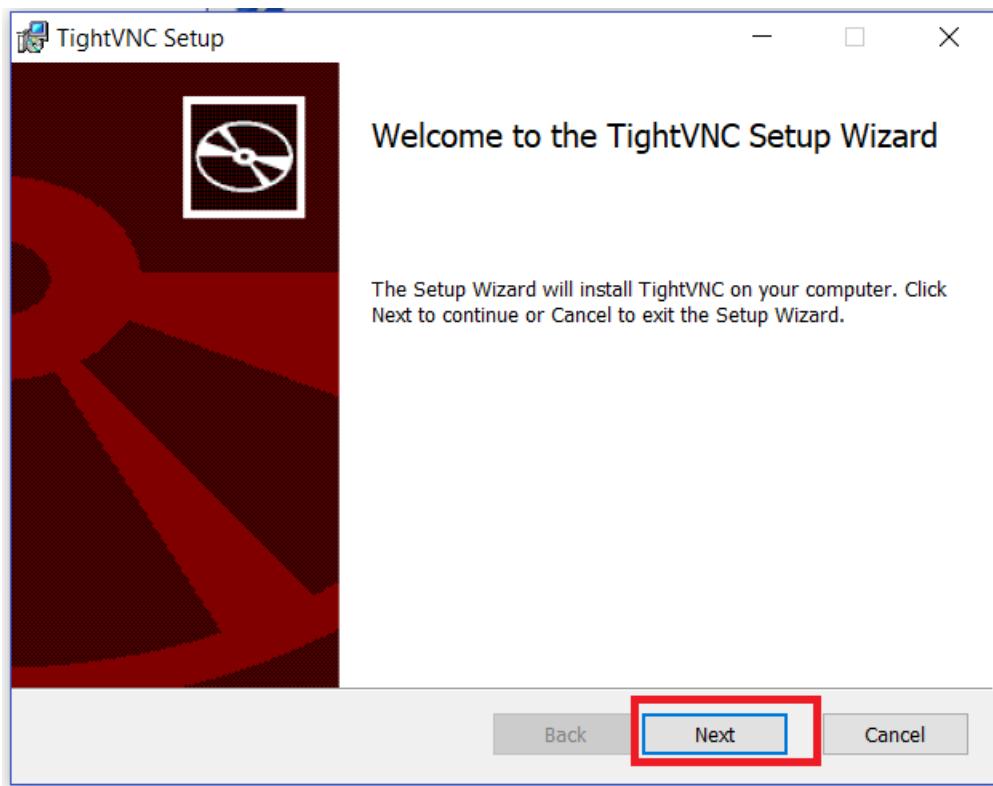
When prompted, select to save the file.



Locate the saved file on your machine and double click to execute. Windows will prompt you to check if you want to open the executable file. Select ok

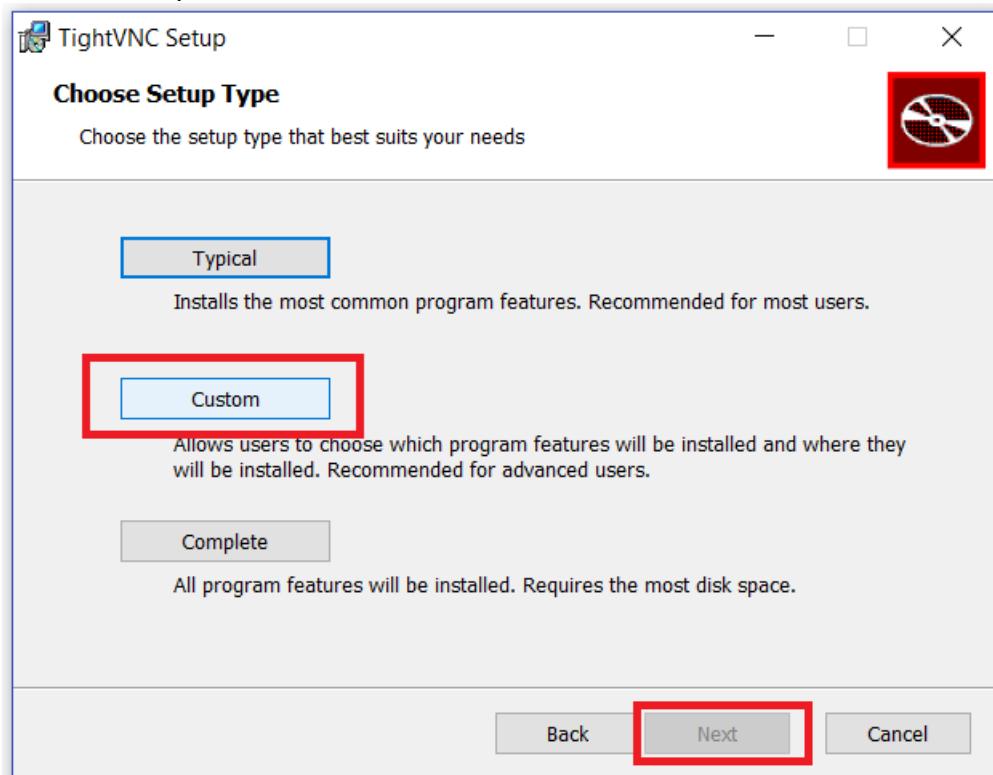


This will start the TightVNC install wizard. Select 'Next' to continue the install.

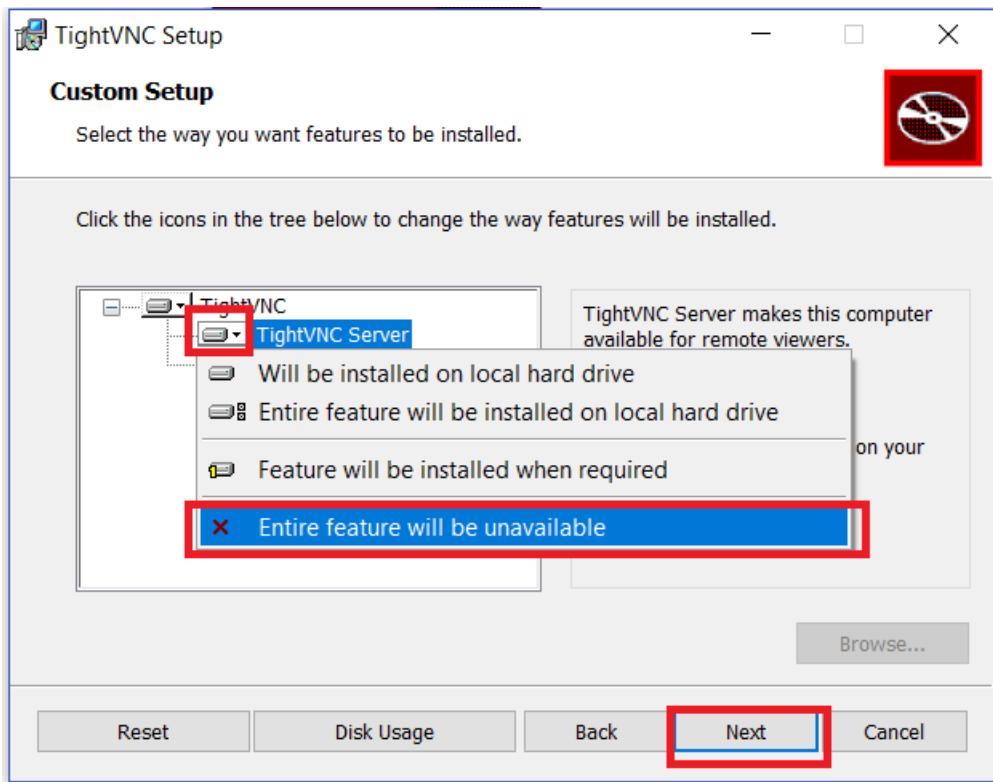


Review and accept the license terms if you agree. Select **Next** to continue.

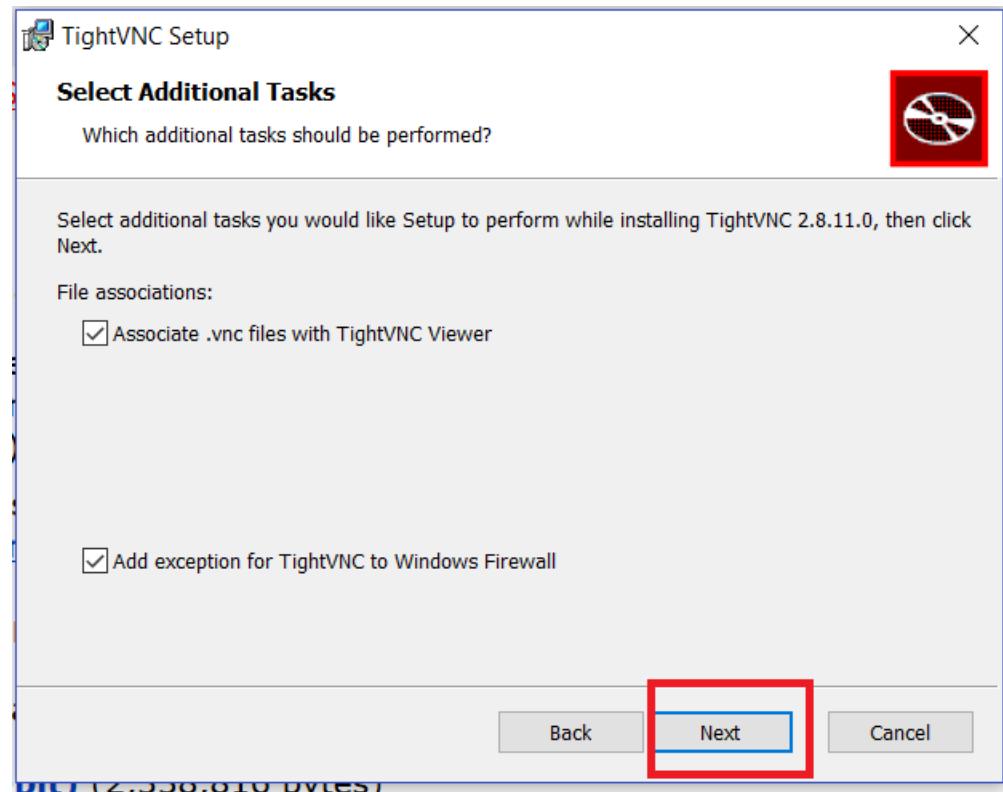
For this lab we only require the TightVNC Viewer, rather than the entire package. Select to carry out a **Custom** installation and select **Next**.



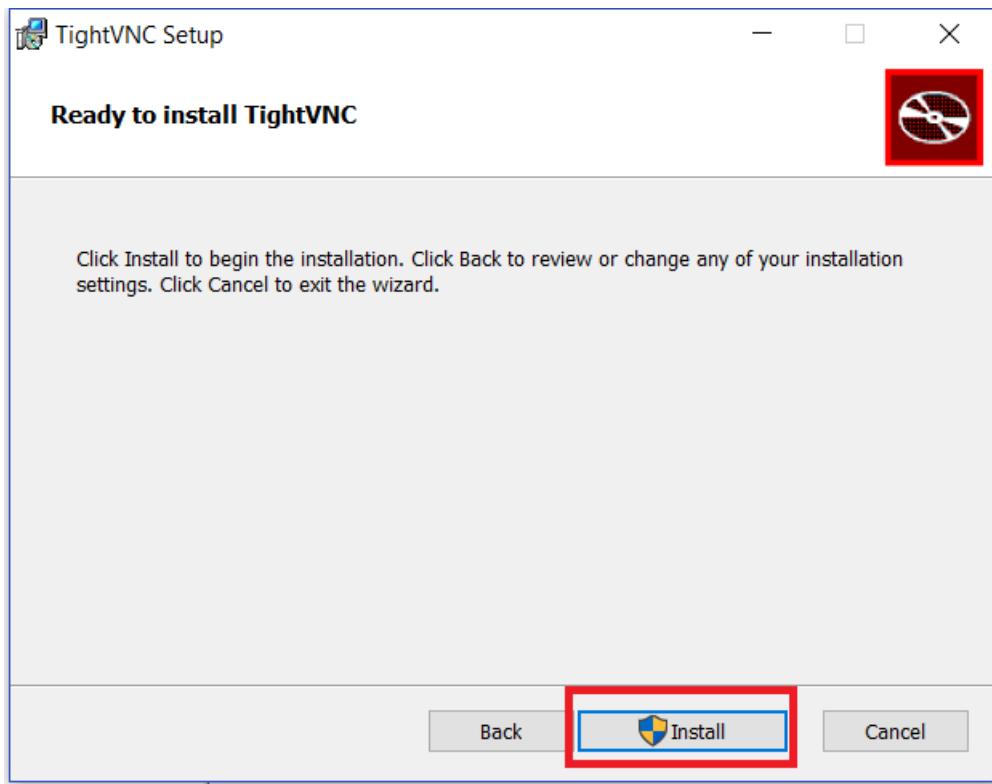
If you do not wish to install the server component, select the drop down next to **TightVNC Server** and select 'X Entire Feature will be unavailable'.



On the 'Select Additional Tasks' screen, choose if you want to change the file associations and the Firewall rules and select 'Next'



Select 'Install'



Section 1 - Introduction to Autonomous Transaction Processing



Lab 1 – Connecting to the Oracle Cloud

Lab 1. Connecting to the Oracle Cloud

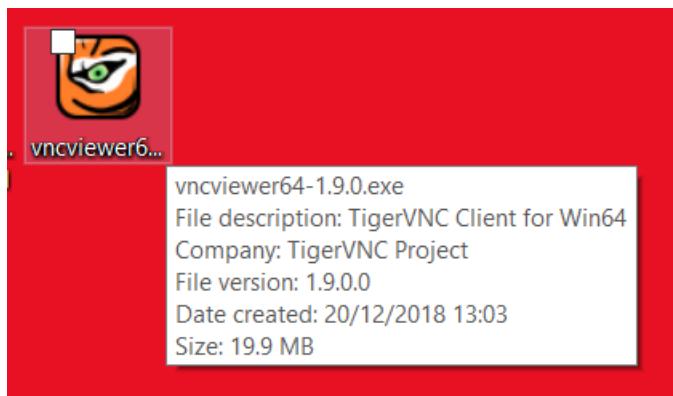
Objectives:

- Learn how to login to the Oracle Cloud Console
- Learn how to provision a new Autonomous Transaction Processing (ATP) Instance

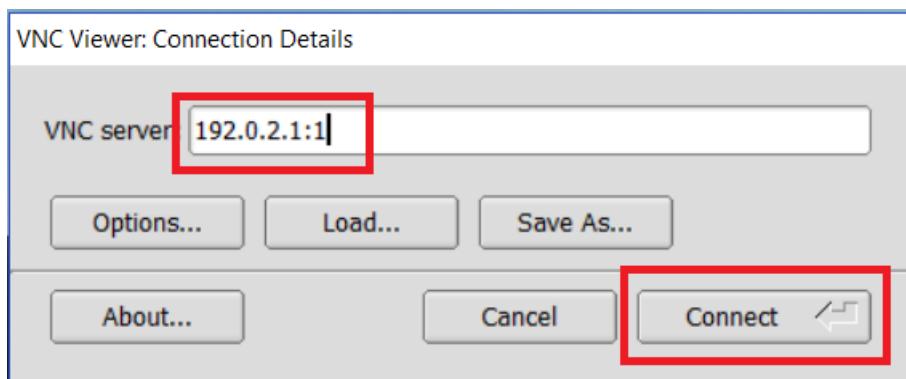
Connecting to your Lab Virtual Machine

To reduce the amount of software that needs to be installed on your local machine we have prepared a Linux virtual machine to act as your desktop.

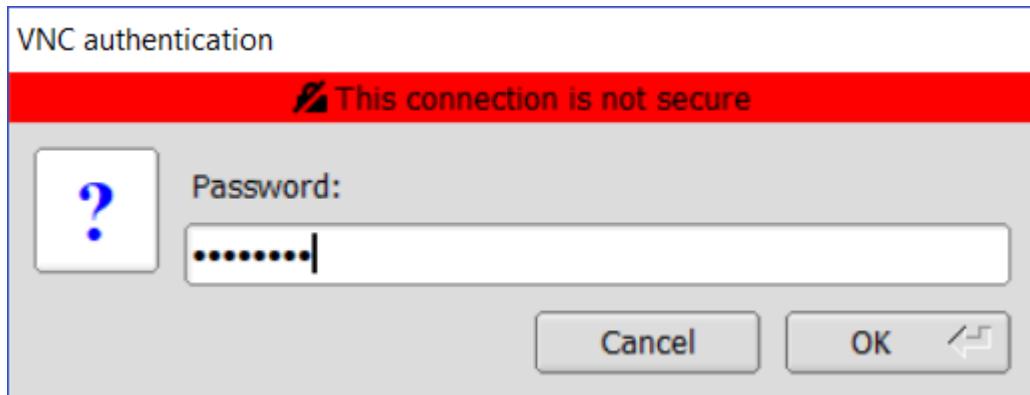
Start TigerVNC viewer (or your choice of VNC viewer) by double clicking on the icon on your local desktop.



Enter connection information provided by your lab leader and select **Connect**. You will use a different IP address than the example in the screenshot below.



Enter the supplied password and hit OK



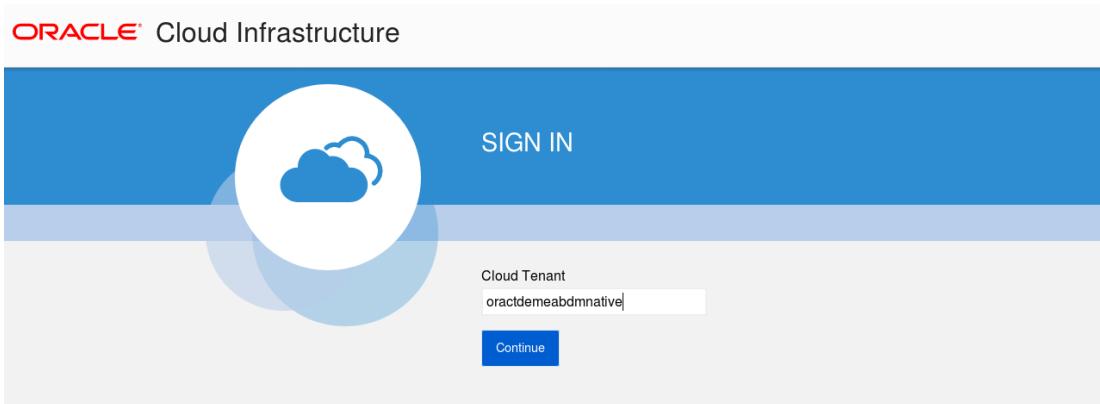
The VNC session will open, and an Oracle Linux desktop session will be displayed.



Signing in to the OCI Console

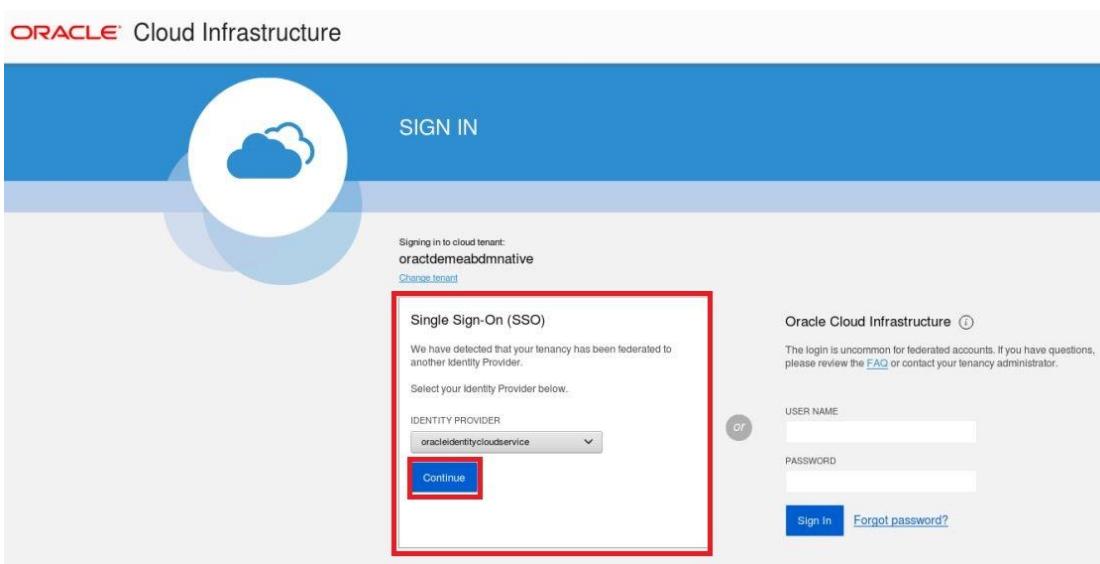
Start Firefox on your Lab VM by double clicking on the Firefox Logo.

Go to the [OCI console](https://console.eu-frankfurt-1.oraclecloud.com). <https://console.eu-frankfurt-1.oraclecloud.com> For this lab we are using the Frankfurt region, even though other regions may be geographically closer as this is where our tenancy has capacity to support the workshop.

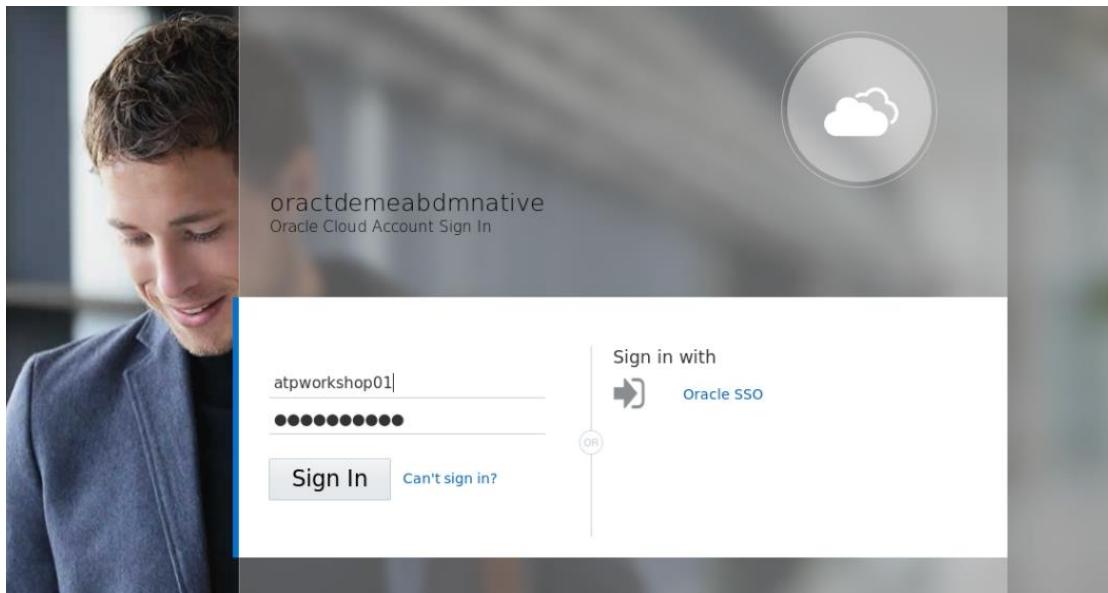


Enter a cloud tenancy of "**oractdemeabdmnative**" and click continue.

On the left-hand section of the page under the title "Single Sign-On" press the '**Continue**' button.



Enter your username and password. Click "**Sign In**".



This will bring you to the console home page.

Verify that your region is set to eu-frankfurt-1 on the top right hand side of your screen, if not, use the drop down list to select 'eu-frankfurt-1'

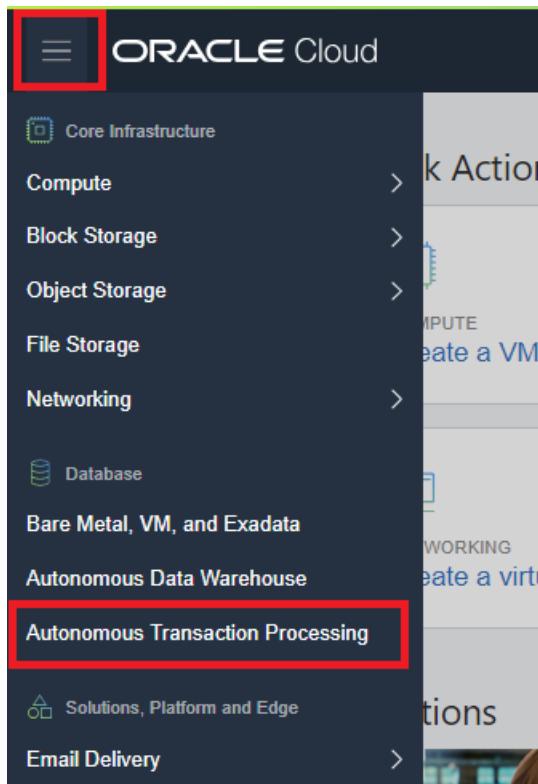


Locating your ATP Instance

As part of the preparation for this lab, ATP instances have been pre-created. Your lab leader will have demonstrated a provisioning process live and the process is documented using screenshots in Appendix A.

Click on the **MENU** link at the top left of the page.

This will produce a drop-down menu, where you should select "**Autonomous Transaction Processing**"



This will take you to the management console page for ATP in the root compartment of the tenancy. The red warning icon "Forbidden" indicates that the Policies within the tenancy does not allow your user to create ATP Instances in the root compartment.

The screenshot shows the 'Autonomous Database' management console. On the left, there are filters for 'List Scope' (COMPARTMENT: oractdemeabdmautodb (root)), 'Filters' (STATE: Any state), and 'WORKLOAD TYPE' (ATP). On the right, the main area displays a table titled 'Autonomous Databases in oractdemeabdmautodb (root) Compartment'. The table has columns: Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Created. A single row is present, showing a red exclamation mark icon and the word 'Forbidden' under the 'State' column. At the bottom, it says 'No Autonomous Databases < Page 1 >'.

To locate your ATP instance in this tenancy you need to select a compartment.

Click on the pulldown menu marked **Compartment**. We will be using the **ATP_Workshop** compartment. Please select this from the drop down list

Autonomous Transaction Processing

List Scope

COMPARTMENT

oractdemeabdmautodb (root)

Search compartments

- oractdemeabdmautodb (root)
 - ATP_Workshop
 - ATP_Delegate

Any state

Auto
oracle

Create

Name

Note – Your list of compartments may be different to the one shown above. In these lab notes we will use the compartment called ATP_Delegate in all the screenshots/examples. Your compartment may be different.

The main page will now change to show the list of ATP instances within your compartment, as shown below:

Note - this lab uses the same tenancy and compartment for all lab attendees. You will see ATP instances listed on this page which have already been provisioned in preparation for this lab.

Autonomous Databases in ATP_Delegate Compartment						
Create Autonomous Database						
Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
dd25ATP	Available	dd25ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
dd24ATP	Available	dd24ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:57 GMT
dd23ATP	Available	dd23ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
dd22ATP	Available	dd22ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
dd20ATP	Available	dd20ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
dd19ATP	Available	dd19ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
dd18ATP	Available	dd18ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT
dd17ATP	Available	dd17ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:41:50 GMT
dd16ATP	Available	dd16ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:37:49 GMT
dd15ATP	Available	dd15ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:33:48 GMT

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Note: the list of instances is sorted by provisioning date, not by name. Refer to your lab information sheet to check which ATP instance has been assigned to you for this lab.

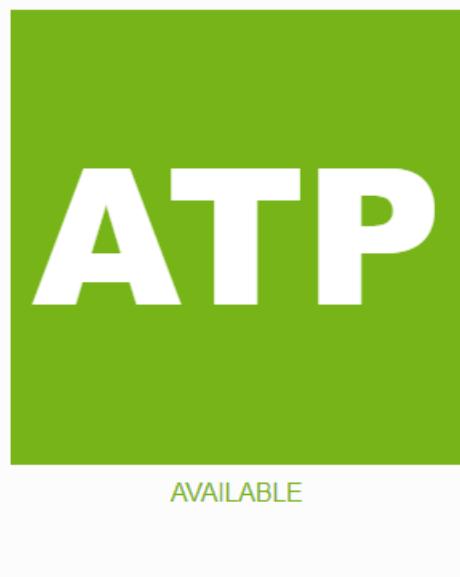
Click on the name of your ATP instance from the list of instances. You may have to scroll down to locate it in the list.

The screenshot shows the Oracle Autonomous Database interface. On the left, there is a sidebar with 'List Scope' and filters for 'COMPARTMENT' (set to 'ATP_Delegate'), 'STATE' (set to 'Any state'), and 'WORKLOAD TYPE' (set to 'All'). In the center, a table titled 'Autonomous Databases in ATP_Delegate Compartment' lists ten instances. The first instance, 'MELATPTRAIN01', is highlighted with a red box. The table columns are: Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Created. The data is as follows:

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Fri, 08 Mar 2019 08:38:38 GMT
dd11ATP	Available	dd11ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
dd12ATP	Available	dd12ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:09:57 GMT
dd13ATP	Available	dd13ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
dd14ATP	Available	dd14ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
dd15ATP	Available	dd15ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
dd16ATP	Available	dd16ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
dd17ATP	Available	dd17ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
dd18ATP	Available	dd18ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT

This will display more information about your instance and you should notice the various menu buttons that help you manage your new instance

A summary of your instance lifecycle status is shown in the large box on the left.



Additional summary information about your instance is displayed, including the workload type. You can also see the Lifecycle Status reported in this region.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2
OCID: ...ago6ma Show Copy
Created: Tue, 18 Jun 2019 07:32:10 GMT
CPU Core Count: 1
Storage (TB): 1
License Type: Bring Your Own Licence
Database Version: 18c
Auto Scaling: Disabled ⓘ
Lifecycle State: Available

Infrastructure

Dedicated Infrastructure: No
Backup
Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT

Return to the main page which list all your ATP instances by clicking on the Autonomous Database link at the top of the page:

Autonomous Database x Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console

Autonomous Database Information T

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb
OCID: ...ago6ma Show Copy
Created: Tue, 18 Jun 2019 07:32:10 GMT
CPU Core Count: 1

You are now ready to start Lab 2.



Lab 2 - Securely Connecting to Autonomous Transaction Processing

Lab 2. Securely Connecting to Autonomous Transaction Processing

Objectives:

- Learn about the different Consumer Groups in Autonomous Transaction Processing (ATP)
- Learn how to download the credential wallet for your ATP instance
- Learn how to securely connect desktop tools to ATP

Applications and tools connect to ATP databases by using Oracle Net Services (also known as SQL*Net). SQL*Net supports a variety of connection types to ATP databases, including Oracle Call Interface (OCI), ODBC drivers, JDBC OC, and JDBC Thin Driver. Unlike other cloud services you do not get a UNIX command line interface on the system hosting your ATP instance, this reduces the complexity and need of UNIX skills required to administer it.

The sample SQL scripts for this lab are available in your VM under the directory `/home/oracle/labScripts/lab2`.

Managing Priorities on Autonomous Transaction Processing

The priority of user requests in ATP is determined by the database service the user is connected to. Users are required to select a service when connecting to the database. The service names are in the format:

- *database_name_tpurgent*
- *database_name_tp*
- *database_name_low*
- *database_name_medium*
- *database_name_high*

These services map to the LOW, MEDIUM, HIGH, TP and TPURGENT consumer groups. For example, a user connecting to *database_name_low* service uses the consumer group LOW.

The basic characteristics of these consumer groups are:

- **tpurgent**: The highest priority application connection service for time critical transaction processing operations. This connection service supports manual parallelism.
- **tp**: A typical application connection service for transaction processing operations. Queries run serially.

- **high:** A high priority application connection service for reporting and batch operations with low concurrency requirements. All operations run in parallel (if you have multiple OCPU assigned to your instance) and are subject to queuing.
- **medium:** A typical application connection service for reporting and batch operations. All operations run in parallel and are subject to queuing.
- **low:** A lowest priority application connection service for high concurrency reporting or batch processing operations. Queries run serially.

By default, the CPU/IO shares assigned to the consumer groups TPURGENT, TP, HIGH, MEDIUM, and LOW are 12, 8, 4, 2, and 1, respectively. The shares determine how much CPU/IO resources a consumer group can use with respect to the other consumer groups. With the default settings the consumer group TPURGENT will be able to use 12 times more CPU/IO resources compared to LOW, when needed. The consumer group TP will be able to use 4 times more CPU/IO resources compared to MEDIUM, when needed.

To change the default values for the shares you can use the PL/SQL procedure `cs_resource_manager.update_plan_directive`.

As a database administrator and an application developer you need to select the database service based on your performance, concurrency and parallelism requirements.

Downloading the credentials wallet

As ATP only accepts secure connections to the database, you need to download the wallet file containing your credentials first.

The wallet is downloaded from the ATP service console, or from the "**DB Connection**" button on the instance details page. To access the ATP Service console, find your database on the table listing ATP instances and click on the three vertical dots on the right-hand side.

In the pop-up menu select **Service Console**.

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATTRAIN01	Available	MELATTRAIN01	1	1	Transaction Processing	Fri, 08 Mar 2019 08:38:38 GMT
dd25ATP	Available	dd25ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
dd24ATP	Available	dd24ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:09:57 GMT
dd23ATP	Available	dd23ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
dd22ATP	Available	dd22ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
dd20ATP	Available	dd20ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
dd19ATP	Available	dd19ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
dd18ATP	Available	dd18ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT

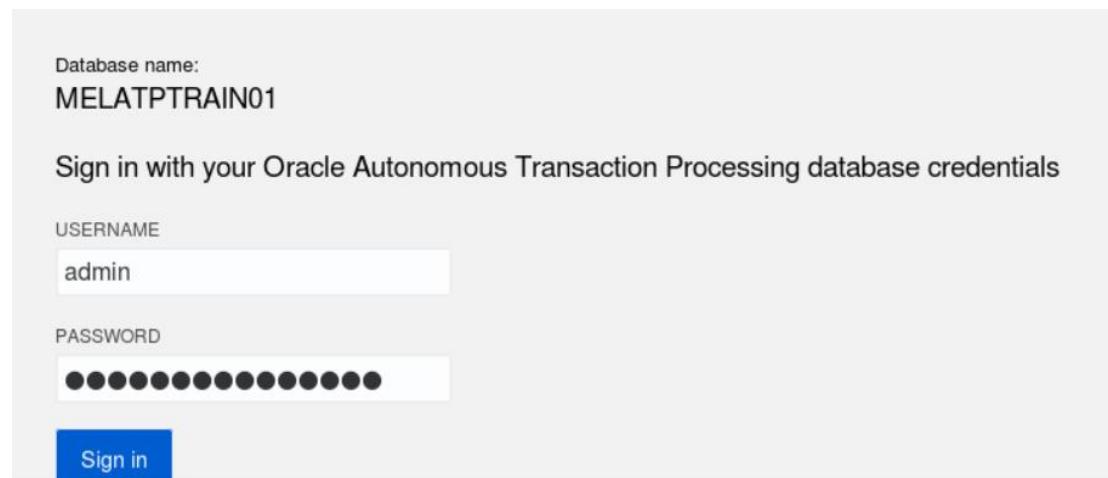
Oracle Autonomous Transaction Processing Hands-on Lab Guide

This will open a new browser tab for the Service Console.

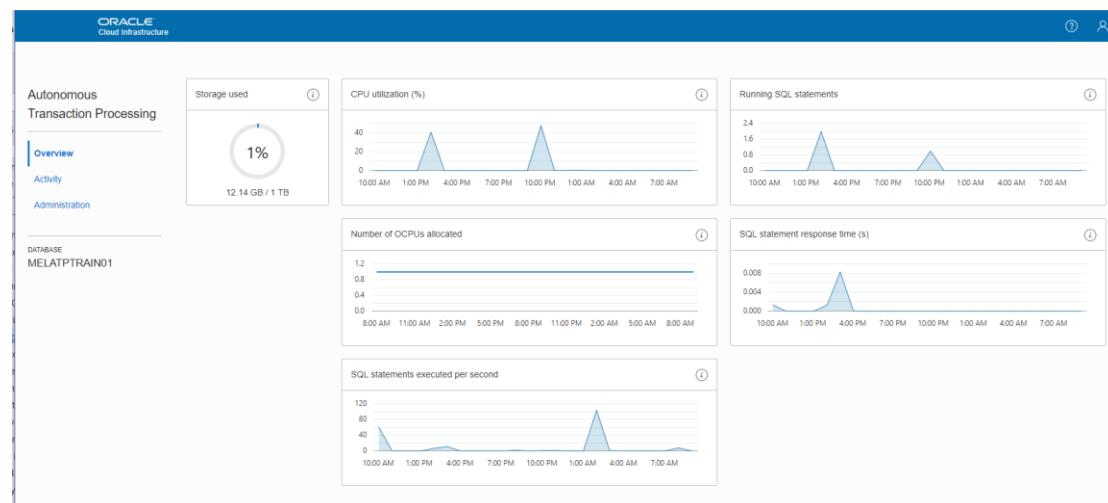
If you are prompted to do so, sign in to the service console with the following information.

Username: admin

Password: The administrator password you specified during provisioning



You will now see the main **dashboard** page for your instance. As we have not generated any load into the instance, it may have 'No data to display' in the information panes. Later lessons will explore the **Overview** and **Activity** tabs in more detail.



Click the "**Administration**" link in the left-hand side menu and click "**Download Client Credentials**" to download the wallet.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

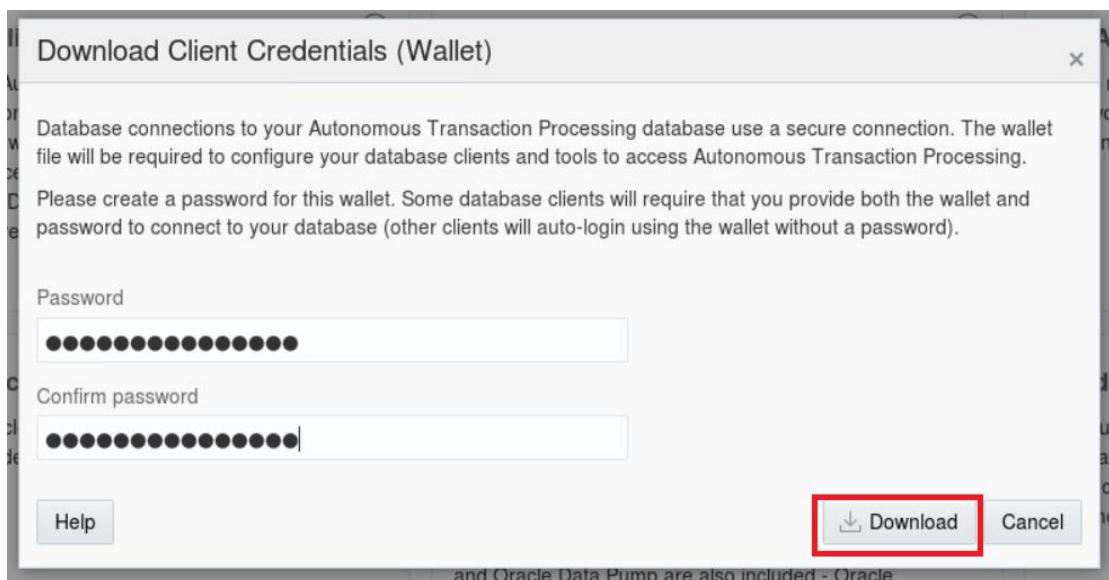
The screenshot shows the Oracle Cloud Infrastructure console with the Autonomous Transaction Processing service selected. The 'Administration' tab is highlighted with a red box. The 'Download Client Credentials (Wallet)' section is also highlighted with a red box, containing the following text:

Download Client Credentials (Wallet) ⓘ
Connections to Autonomous Transaction Processing use a secure connection. Your existing tools and applications will need to use this wallet file to connect to your Autonomous Transaction Processing instance. If you are familiar with using an Oracle Database within your own data center, you may not have previously used these secure connections.

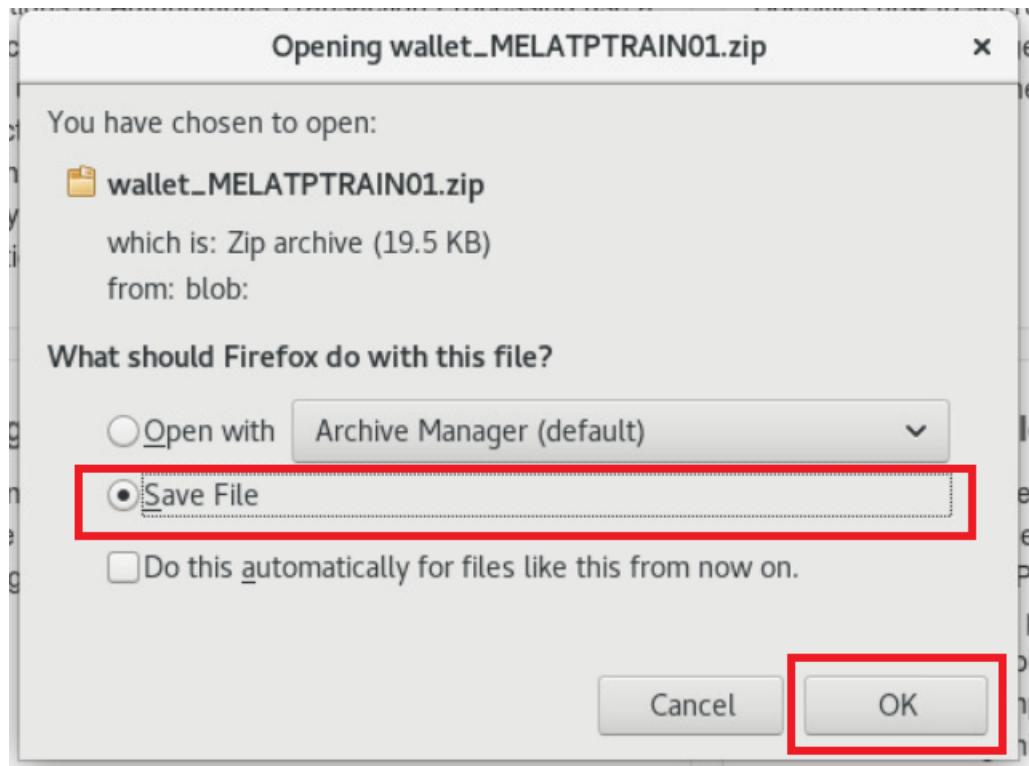
Other sections visible include 'Set Resource Management Rules', 'Set Administrator Password', 'Manage Oracle ML Users', 'Download Oracle Instant Client', and 'Send Feedback to Oracle'.

Specify a password for the wallet. Some applications require this password when connecting to the database, for example some JDBC thin applications will require this password to use as the keystore password. Note that this password is separate from the admin password and can be set to a different value. For this lab, you could use the value ATPwelcome-1234 or another memorable password of your choice.

Click Download to download the wallet file to your lab virtual machine.



Select to Save the file, and then click ok. This will save the file in the default downloads location \$HOME/Downloads



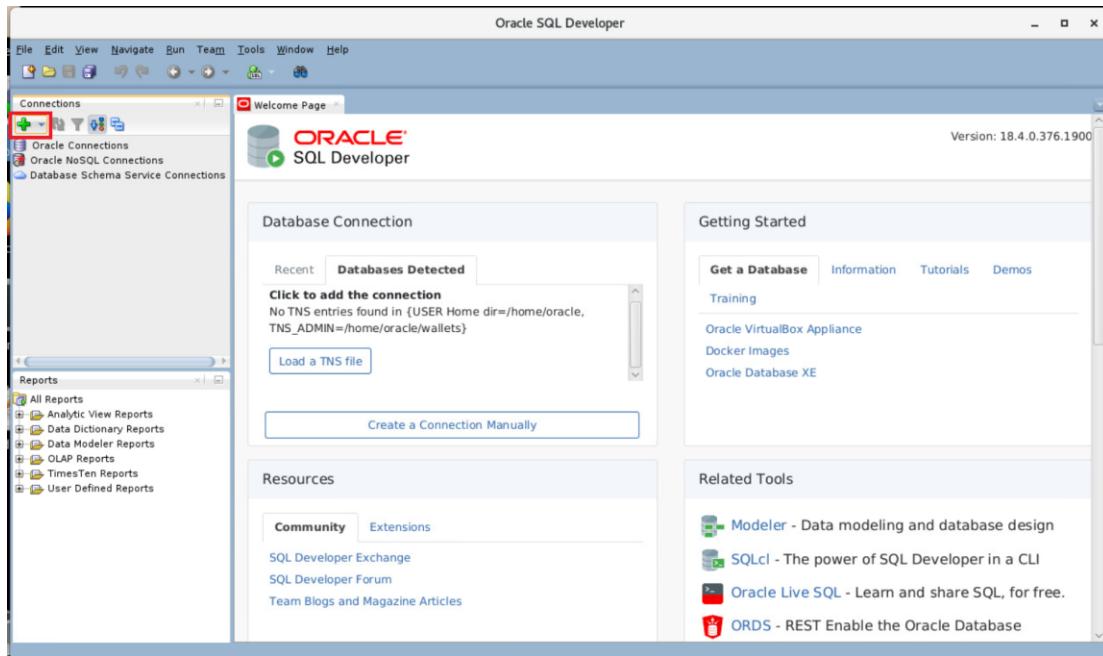
Connecting to the database using SQL Developer

Minimise your Firefox window, and on your Lab VM desktop, start SQL Developer by double clicking on the icon.

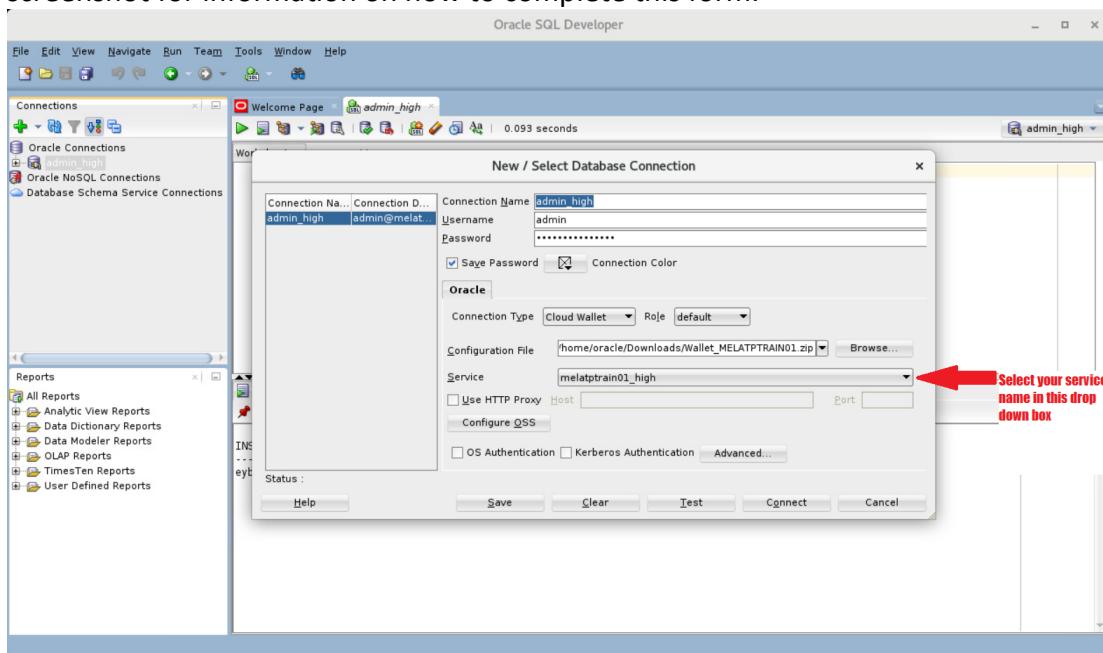


Click the Create Connection icon in the Connections toolbox on the top left of the SQL Developer homepage.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



This will open the **New>Select Database Connection Screen**. See below the screenshot for information on how to complete this form.



Connection Name: admin_high

Username: admin

Password: The admin password you specified during the provisioning process.

Connection Type: Cloud Wallet

Configuration File: Enter the full path for the wallet file you downloaded earlier in the lab or click the Browse button to point to the locate the file (by default it will be under your Downloads directory).

Service: The Wallet will contain the service names for all the ATP databases in the tenancy so this list could be long. Please make sure that you are selecting your database. As discussed previously there are 5 pre-configured

database services for each database. Pick `<your databasename>_high` for this lab. For example, if you created a database named **melatptrain01** select **melatptrain01_high** as the service.

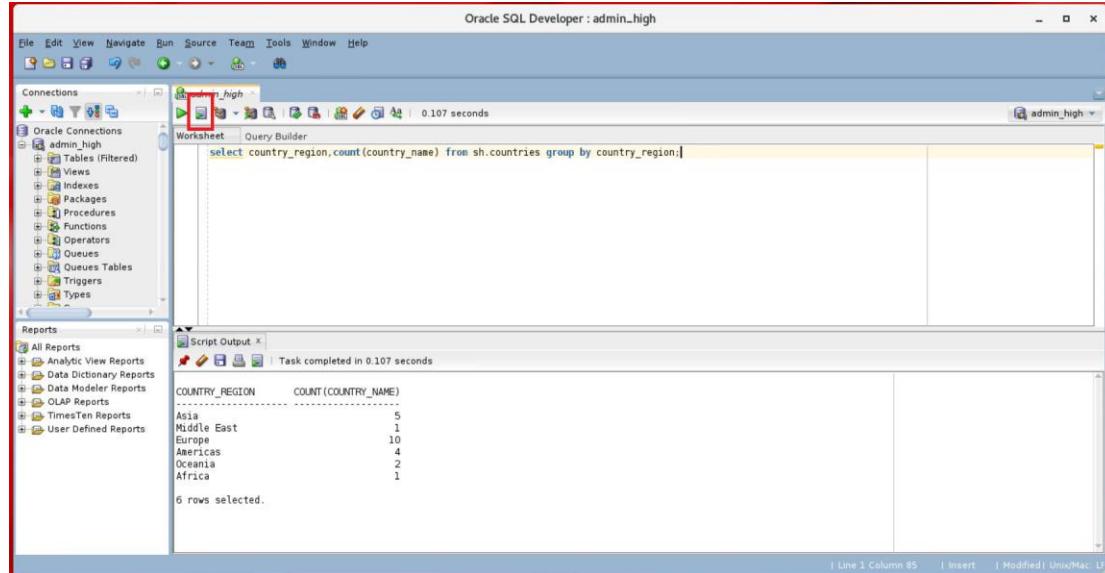
Test your connection by clicking the Test button, if it succeeds save your connection information by clicking Save, then connect to your database by clicking the Connect button.

You can now run a test query using the sample data in the SH schema.

In the SQL Worksheet enter the following SQL:

```
select country_region, count(country_name) from sh.countries group by country_region;
```

and press the ‘Run Script’ button or press F5.



Note – You do not have to use GUI tools to access an ATP instance. Other Oracle Client utilities such as SQL Plus can connect to the ATP instance using a wallet.

(Optional) Secure access to your Autonomous Database using Access Control Lists

An Access Control List (ACL) provides additional protection for your Autonomous Database by allowing only the IP addresses in the list to connect to the database.

When you provision a new Autonomous Database, it does not have an initial ACL. You can use the Oracle Cloud Infrastructure Console, API, or CLI to create an ACL for the database by adding a minimum of one entry to the list. An entry can be a

Oracle Autonomous Transaction Processing Hands-on Lab Guide

comma-separated list of CIDR blocks or public IP addresses. You can modify the list at any time. Setting the ACL for the Autonomous Database does not block administration activities via the Service Console or the Oracle Cloud Infrastructure Console. Removing all entries from the list makes the database accessible to all clients with the applicable credentials.

In this exercise you will:

- Create an ACL that will block your access to your Autonomous Database.
- Alter this ACL to only allow your lab VM access to your Autonomous Database.

It is important to complete this exercise successfully to allow subsequent labs to complete.

Creating your initial ACL

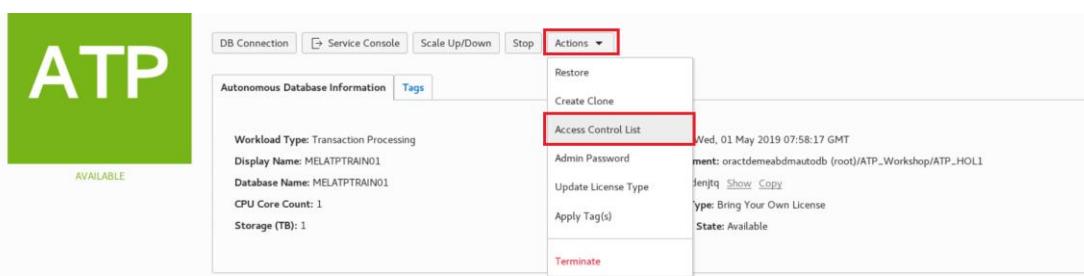
Navigate back to the **Autonomous Transaction Processing** page in the lab compartment.

Click on the name of your database to open the instance details screen.

Create Autonomous Database						
Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Wed, 01 May 2019 07:58:17 GMT
ss06ATP	Available	ss06ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 13:03:53 GMT
ss05ATP	Available	ss05ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:59:51 GMT
ss04ATP	Available	ss04ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:55:50 GMT
ss03ATP	Available	ss03ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:51:49 GMT
ss02ATP	Available	ss02ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:47:48 GMT
ss01ATP	Available	ss01ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:43:47 GMT

Displaying 7 Autonomous Databases < Page 1 >

In the **Actions** drop down select "Access Control List"



In the pop up box enter the IP Address **192.168.28.1** and select "Update". This is a non-routable IP address that is not in use in the lab environment. Once you set an ACL only new connections from IP addresses that match the ACL will be allowed to connect.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Access Control List [help](#) [cancel](#)

Specify the IP addresses allowed to access this database. You can use a comma-separated list to enter multiple IP addresses. An access control list blocks all IP addresses that are not in the list from accessing the database.

IP NOTATION TYPE IP ADDRESSES [x](#)

[+ Additional Entry](#)

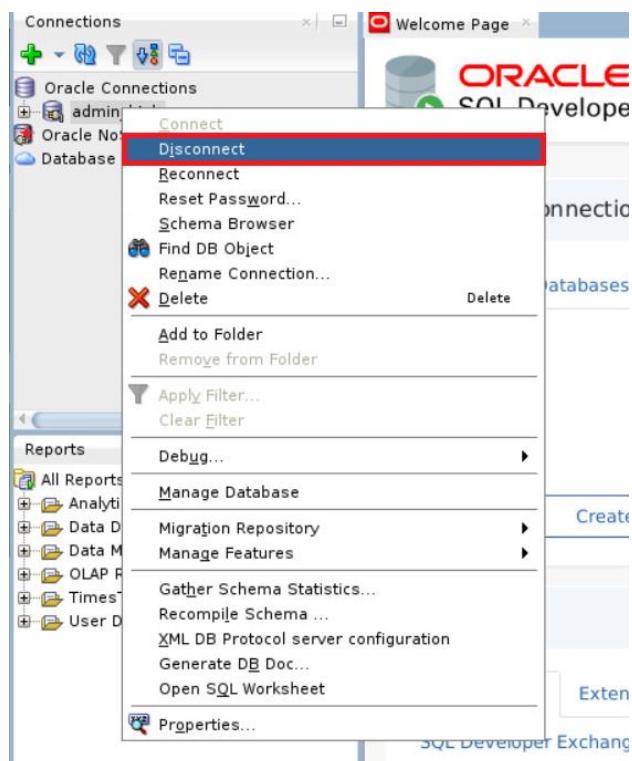
The Lifecycle status will change to "**Updating**".



Wait until the Lifecycle status is "**Available**"

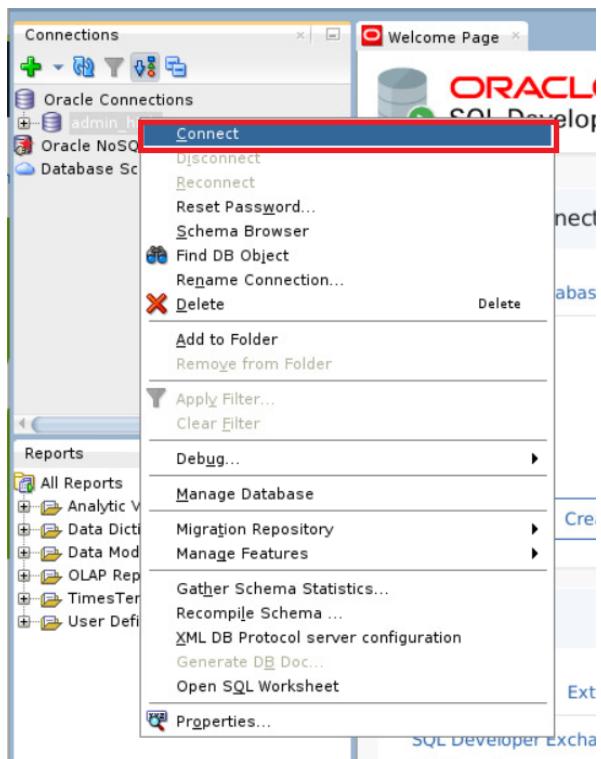


Return to your SQL Developer window. Select your "**admin_high**" connection, right click to bring up the menu and select "**Disconnect**".

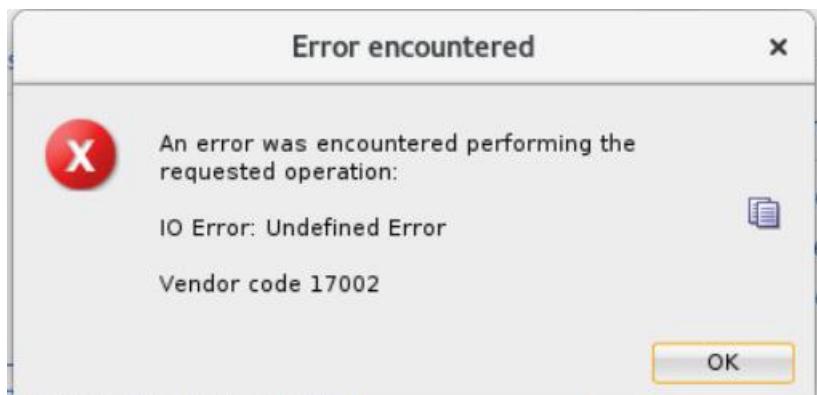


Connect to your database again. Select your "**admin_high**" connection, right click to bring up the menu and select "**Connect**".

Oracle Autonomous Transaction Processing Hands-on Lab Guide



The connection to the database should fail with "IO Error: Undefined Error"



Note – If your connection to the database succeeds, this usually means that there is a problem with the definition of your admin_high connection and the 'Service Name' was not set to your database.

Correcting the ACL to allow connections

To ensure Return to your Firefox browser and navigate back to the **Autonomous Transaction Processing** page in the lab compartment.

Click on the name of your database to open the instance details screen.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Create Autonomous Database						
Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Wed, 01 May 2019 07:58:17 GMT
ss06ATP	Available	ss06ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 13:03:53 GMT
ss05ATP	Available	ss05ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:59:51 GMT
ss04ATP	Available	ss04ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:55:50 GMT
ss03ATP	Available	ss03ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:51:49 GMT
ss02ATP	Available	ss02ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:47:48 GMT
ss01ATP	Available	ss01ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:43:47 GMT

Displaying 7 Autonomous Databases < Page 1 >

In the **Actions** drop down select "Access Control List"

The screenshot shows the Oracle Cloud Infrastructure console. On the left, there is a large green button labeled 'ATP' with 'AVAILABLE' below it. To the right, there is a card for the Autonomous Database 'MELATPTRAIN01'. At the top of the card, there is a 'Actions' dropdown menu with several options: 'Restore', 'Create Clone', 'Access Control List', 'Admin Password', 'Update License Type', 'Apply Tag(s)', and 'Terminate'. The 'Access Control List' option is highlighted with a red box.

In the pop up box use the small x next to the IP address to delete the existing entry.

The screenshot shows the 'Access Control List' dialog box. It has a header 'Access Control List' with 'help' and 'cancel' links. Below the header is a text area: 'Specify the IP addresses allowed to access this database. You can use a comma-separated list to enter multiple IP addresses. An access control list blocks all IP addresses that are not in the list from accessing the database.' There are two input fields: 'IP NOTATION TYPE' (set to 'IP Address') and 'IP ADDRESSES' (containing '192.168.28.1'). To the right of the 'IP ADDRESSES' field is a small red box containing a white 'x' icon, which is typically used for deletion. At the bottom are 'Update' and '+ Additional Entry' buttons.

Select the "**Additional Entry**" button to create a new ACL entry.

The screenshot shows the 'Access Control List' dialog box again. It has a header 'Access Control List' with 'help' and 'cancel' links. Below the header is the same text area: 'Specify the IP addresses allowed to access this database. You can use a comma-separated list to enter multiple IP addresses. An access control list blocks all IP addresses that are not in the list from accessing the database.' The 'IP ADDRESSES' field is empty. To the right of the field is a red box around the '+ Additional Entry' button. At the bottom are 'Update' and '+ Additional Entry' buttons.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Enter the IP Address of your Lab Virtual Machine. This is the address you used to connect using the VNC viewer, without the ':1' on the end. Each Virtual Machine has its own IP address.

For example: if your VNC connection was to 192.0.2.1:1 the IP address you enter will be 192.0.2.1.

Access Control List [help](#) [cancel](#)

Specify the IP addresses allowed to access this database. You can use a comma-separated list to enter multiple IP addresses. An access control list blocks all IP addresses that are not in the list from accessing the database.

IP NOTATION TYPE IP ADDRESSES

Select "**Update**".

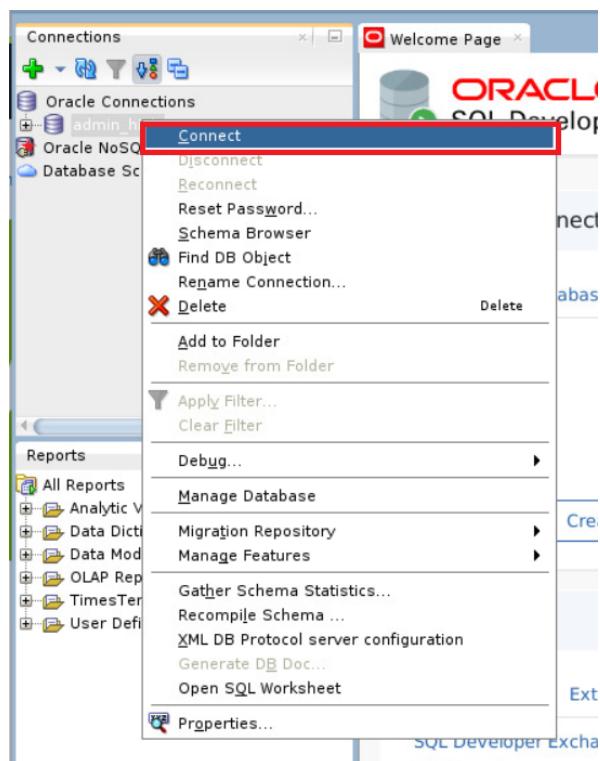
The Lifecycle status will change to "**Updating**".



Wait until the Lifecycle status is "**Available**"



Return to your SQL Developer window. Connect to your database again. Select your "**admin_high**" connection, right click to bring up the menu and select "**Connect**".



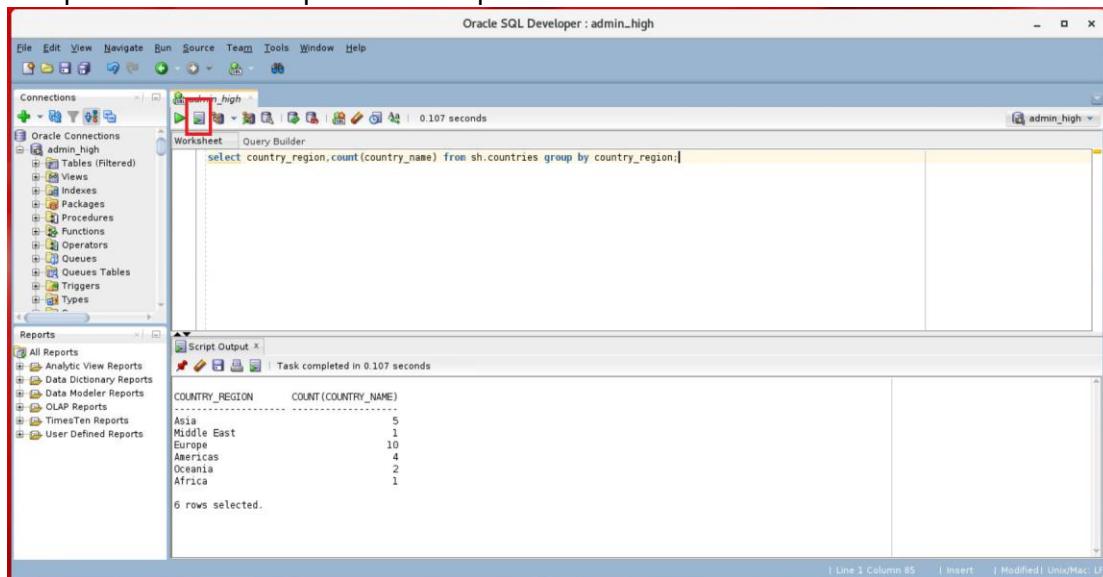
The connection should succeed. Test this by running a simple SQL query in SQL Worksheet.

In the SQL Worksheet enter the following SQL:

Oracle Autonomous Transaction Processing Hands-on Lab Guide

```
select country_region, count(country_name) from sh.countries group by country_region;
```

and press the 'Run Script' button or press F5.



Note – If your connection to the database still fails, verify that the IP address that you have specified for your ACL is correct.

Remove all the ACL entries

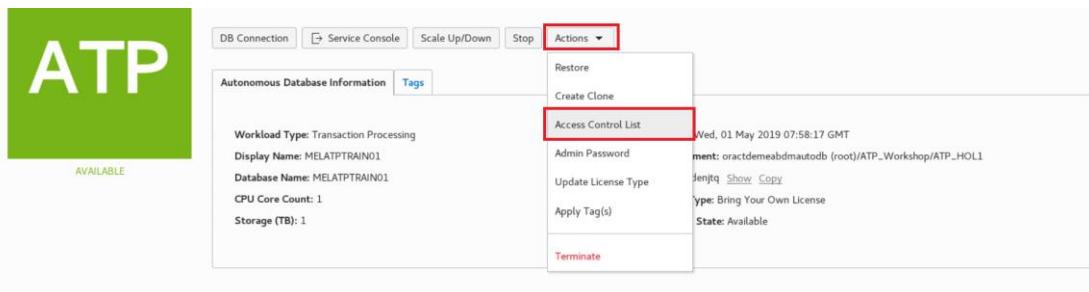
Return to your Firefox browser and navigate back to the **Autonomous Transaction Processing** page in the lab compartment.

Click on the name of your database to open the instance details screen.

Create Autonomous Database						
Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Wed, 01 May 2019 07:58:17 GMT
ss06ATP	Available	ss06ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 13:03:53 GMT
ss05ATP	Available	ss05ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:59:51 GMT
ss04ATP	Available	ss04ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:55:50 GMT
ss03ATP	Available	ss03ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:51:49 GMT
ss02ATP	Available	ss02ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:47:48 GMT
ss01ATP	Available	ss01ATP	1	1	Transaction Processing	Fri, 26 Apr 2019 12:43:47 GMT

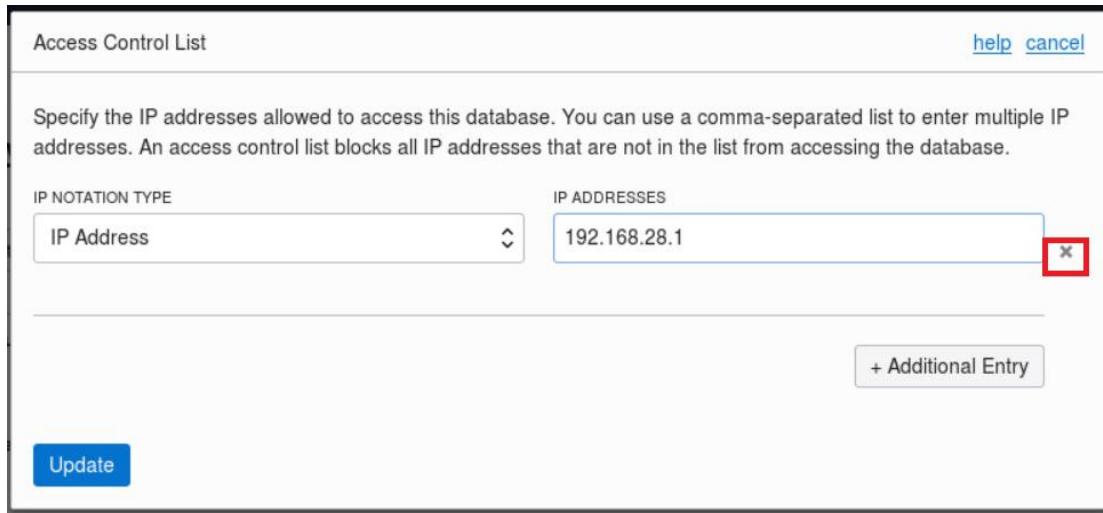
In the **Actions** drop down select "Access Control List"

Oracle Autonomous Transaction Processing Hands-on Lab Guide



The screenshot shows the ATP console interface. At the top, there are buttons for 'DB Connection', 'Service Console', 'Scale Up/Down', 'Stop', and 'Actions'. The 'Actions' button is expanded to show a dropdown menu with options: 'Restore', 'Create Clone', 'Access Control List' (which is highlighted with a red box), 'Admin Password', 'Update License Type', 'Apply Tag(s)', and 'Terminate'. Below the dropdown, there is some detailed database information: Workload Type: Transaction Processing, Display Name: MELATPTRAIN01, Database Name: MELATPTRAIN01, CPU Core Count: 1, Storage (TB): 1. To the right of the dropdown, there is a timestamp: Wed, 01 May 2019 07:58:17 GMT, and a note: 'Management: oractdemebdautodb (root)/ATP_...Workshop/ATP_HOL1'.

In the pop up box use the small x next to the IP address to delete the existing entry.



The screenshot shows the 'Access Control List' dialog box. It has a header 'Access Control List' with 'help' and 'cancel' links. The main area contains a message: 'Specify the IP addresses allowed to access this database. You can use a comma-separated list to enter multiple IP addresses. An access control list blocks all IP addresses that are not in the list from accessing the database.' Below this, there are two sections: 'IP NOTATION TYPE' (set to 'IP Address') and 'IP ADDRESSES' (containing '192.168.28.1'). A red box highlights the delete 'x' icon next to the IP address. At the bottom are 'Update' and '+ Additional Entry' buttons.

The pop-up will now have no IP Addresses or CIDR blocks specified. Select "Update" to save this change.



The screenshot shows the 'Access Control List' dialog box again. The 'IP ADDRESSES' field is now empty. The 'Update' button at the bottom left is highlighted with a red box. The rest of the interface is identical to the previous screenshot.

The Lifecycle status will change to "**Updating**".



ATP

UPDATING...

Wait until the Lifecycle status is "Available"



ATP

AVAILABLE

Once you have successfully removed all the ACL entries you have completed Lab 2.

Section 2 -

Focus on DBAs



Lab 3 - Scaling an ATP Instance

Lab 3. Scaling an Automatic Transaction Processing Instance

Objectives:

- Learn how to scale your Autonomous Transaction Processing (ATP) instance.

Scaling in the context of an ATP database means increasing or decreasing the amount of CPU or storage resources allocated to the service. Scaling an ATP instance is easy, flexible and can be done without any downtime so your applications continue to run unaffected while the scaling operation is in progress.

The sample SQL scripts for this lab are available in your VM under the directory **/home/oracle/labScripts/lab3**.

Scaling your instance using the console

To demonstrate that scaling an instance does not cause downtime we will run a long running query in SQL Developer and alter the number of CPUs available to the ATP instance while the SQL is executing.

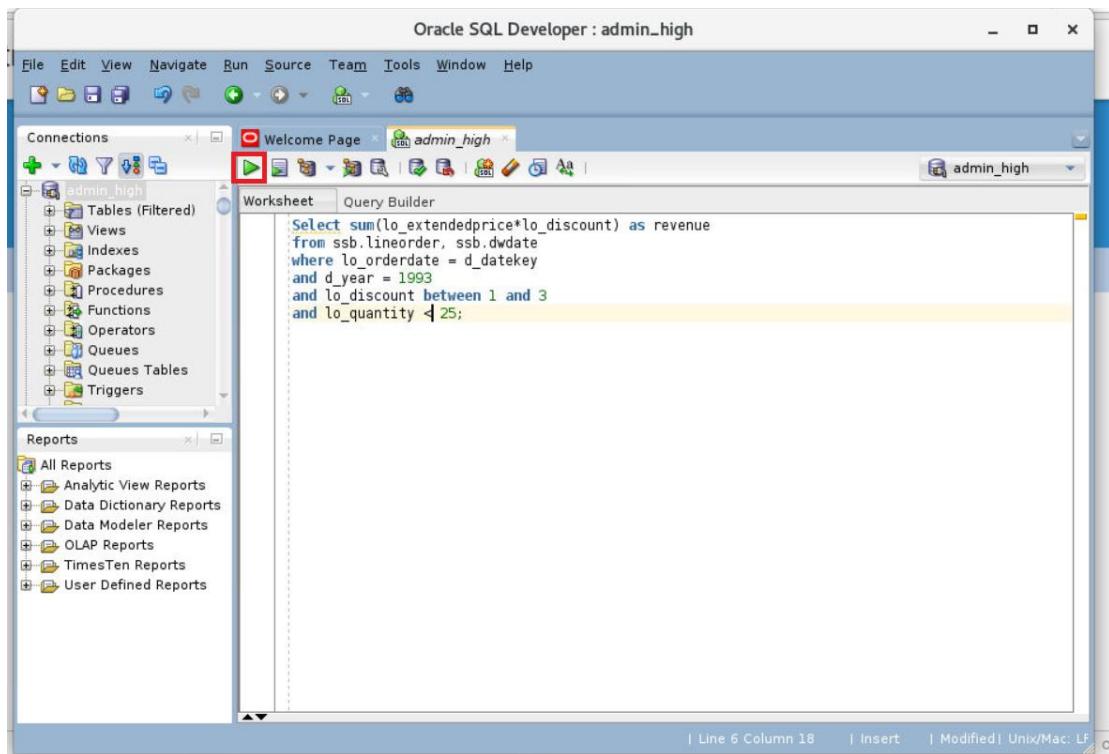
Start SQL Developer and open your **admin_high** connection.

Paste the following long running SQL into the SQL Worksheet and select **Run Statement**. It will run for at least 10 minutes, allowing time to carry out the scaling operations while it is still executing. The sql can be found in the file **/home/oracle/labScripts/lab3/lab3_long_running.sql**

```
select sum(lo_extendedprice*lo_discount) as revenue
from ssb.lineorder, ssb.dwdate
where lo_orderdate = d_datekey
and d_yearmonthnum = 199401
and lo_discount between 4 and 6
and lo_quantity between 26 and 35;
```

Leave this window open and the statement executing.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



Return to your Firefox window that you used to during the provisioning exercise.

Navigate back to the **Autonomous Database** page. Ensure you have selected the correct sub compartment under ATP_Workshop where your database is listed.

Click on the name of your database to open the instance details screen.

A screenshot of the "Autonomous Databases in ATP_Delegate Compartment" page. The left sidebar shows "List Scope" set to "ATP_Delegate" and "Filters" for "STATE" (Any state) and "WORKLOAD TYPE" (All). The main table lists databases with the following columns: Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Created. One row, "MELATPTRAIN01", is highlighted with a red box. The table data is as follows:

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Fri, 08 Mar 2019 08:38:38 GMT
dd25ATP	Available	dd25ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
dd24ATP	Available	dd24ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:09:57 GMT
dd23ATP	Available	dd23ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
dd22ATP	Available	dd22ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
dd20ATP	Available	dd20ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
dd19ATP	Available	dd19ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
dd18ATP	Available	dd18ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT

Select the **Scale Up/Down** button.

A screenshot of the "Autonomous Database Details" page for "MELATPTRAIN01". The top navigation bar shows "Autonomous Database" and "Autonomous Database Details". The main content area has a green banner on the left with "ATP" and "AVAILABLE". The database name "MELATPTRAIN01" is displayed prominently. Below it is a toolbar with buttons for "DB Connection", "Service Console", "Scale Up/Down" (which is highlighted with a red box), "Stop", and "Actions". There are two tabs at the bottom: "Autonomous Database Information" and "Tags". The "General Information" section is visible on the left, and the "Infrastructure" section is on the right.

Fill in the form with the following information.

Change the **CPU core count** from 1 to 2 (i.e. you are going to add 1 OCPU)
Leave the **storage capacity** set at 1

The screenshot shows a 'Scale Up/Down' dialog box. It has two main input fields: 'CPU CORE COUNT' containing '2' and 'STORAGE (TB)' containing '1'. Below these fields are descriptive labels: 'The number of CPU cores to enable. Available cores are subject to your tenancy's service limits.' and 'The amount of storage to allocate.'. There is also a checkbox for 'AUTO SCALING' which is unchecked. At the bottom is a blue 'Update' button, which is highlighted with a red rectangular border.

Click “**Update**” after setting the values in the form as shown above.

This will return you to the instance page and the status will now show “**Scaling in progress**” and the “**Scale Up/Down**” button is grayed out.

The screenshot shows the 'Autonomous Database Details' page for database 'MELATPTRAIN01'. On the left is a large green square with the letters 'ATP' in white. Below it, the text 'SCALING IN PROGRESS...' is displayed. At the top right are several buttons: 'DB Connection', 'Service Console', 'Scale Up/Down' (which is grayed out), 'Stop', and 'Actions ▾'. Below these buttons is a tab bar with 'Autonomous Database Information' and 'Tags' (which is selected). The main content area is divided into sections: 'General Information' (Database Name: MELATPTRAIN01, Workload Type: Transaction Processing, Compartment: oraclemeasabnautodb (root)/ATP_Workshop/ATP_HOL2, OCID: ..., Created: Tue, 18 Jun 2019 07:32:10 GMT, CPU Core Count: 1, Storage (TB): 1, License Type: Bring Your Own Licence, Database Version: 18c, Auto Scaling: Disabled) and 'Infrastructure' (Dedicated Infrastructure: No, Backup: Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT).

The page will automatically refresh once the scale operation has completed. The Lifecycle state will return to “**Available**”

Note that the “ATP” square remains green during the scaling process. This means that applications can continue running during the scale operation without downtime.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

MELATPTRAIN01

Autonomous Database Information Tags

General Information

- Database Name: MELATPTRAIN01
- Workload Type: Transaction Processing
- Compartment: oractomedabatadadb (root)/ATP_Workshop/ATP_HOL2
- OCID: ...ag06ma Show Code
- Created: Tue, 18 Jun 2019 07:32:10 GMT
- CPU Core Count: 2
- Storage (TB): 1
- License Type: Bring Your Own Licence
- Database Version: 18c
- Auto Scaling: Disabled ⓘ
- Lifecycle State: Available

Infrastructure

- Dedicated Infrastructure: No
- Backup
- Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT

Bring your SQL Developer window back to the foreground. Verify that the query is still executing and has not been interrupted. You can see this by looking at the Query Result Tab, and the Query Result bar will be moving right to left and there will be rotating symbols around the icon. Once you have verified the query has not been interrupted, proceed to scaling your instance back to 1 OCPU.

Oracle SQL Developer : admin_high

File Edit View Navigate Run Source Team Tools Window Help

Connections

- admin_high
- Tables (Filtered)
- Views
- Indexes
- Packages
- Procedures
- Functions
- Operators
- Queues
- Queues Tables
- Triggers

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Worksheet Query Builder

```
Select sum(lo_extendedprice*lo_discount) as revenue
from ssb.lineorder, ssb.dwdate
where lo_orderdate = d_datekey
and d_year = 1993
and lo_discount between 1 and 3
and lo_quantity < 25;
```

Query Result

Scaling the database using REST

During the previous exercise we used the Console to scale the ATP instance without downtime. It is also possible to control the database via other methods such as the OCI CLI or REST APIs. In this exercise you will scale your instance back to 1 OCPU using REST.

A RESTful API is an application programming interface (API) that uses HTTPS requests to GET, PUT, POST and DELETE data. The Autonomous Transaction Processing Cloud has an extensive REST API to help you provision, manage, and delete your instances.

To help you get started with REST and Oracle Cloud Infrastructure, Oracle has published a series of scripts in different languages that help construct REST requests. As we are using a Linux VM we will use the Bash script that has been provided in the documentation. See the link below for other samples and detailed information on their use.

<https://docs.cloud.oracle.com/iaas/Content/API/Concepts/signingrequests.htm#Bash>

We have downloaded the Oracle provided script Bash script to your lab virtual machine. It is named `/home/oracle/labScripts/lab3/oci-curl`.

The downloaded script helps to construct the REST command that will be sent to the database. This includes formatting the headers and Oracle cloud authentication parameters correctly. We have preconfigured this script to save time in this lab.

To allow REST scripts to be able to communicate with the Oracle cloud, we need to authenticate them to ensure the security of the cloud environment. A REST script uses an already created Oracle cloud user. The REST script can only perform actions that the selected cloud user can perform, these permissions are part of the policies which apply to that user. If a REST script tries to perform an action that the cloud user is not authorized to perform the REST script will fail.

To ensure authentication between the REST script and the cloud user we create an API signing key that the REST script will use to authenticate itself when running. We have generated a key pair (public / private) under the `/home/oracle/labScripts/lab3/key` directory

Uploading the Key to Oracle Cloud Infrastructure

We need to upload the public key to your user in the OCI console

The screenshot shows the Oracle Cloud OCI console interface. At the top, there's a navigation bar with the Oracle Cloud logo, a search bar, and a profile icon. The profile icon is highlighted with a red box. To the right of the profile icon, it says "Profile atpworkshop01" and "Tenancy: oscemea001". Below the navigation bar, the main content area has a sidebar on the left with sections for "Autonomous Database", "List Scope", "COMPARTMENT" (set to "ATP_Delegate"), and "Filters". The main content area displays a list titled "Autonomous Databases in ATP_Delegate Compartment". The list includes two entries:

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Last Updated
MELATPTRAIN1	Available	MELATPTRAIN1	1	1	Transaction Processing	Wed, 29 May 2019 09:04:35 GMT
RESTADB1	Available	RESTADB1	1	1	Transaction Processing	Fri, 24 May 2019 19:42:33 GMT

At the bottom of the page, there are links for "Terms of Use and Privacy" and "Cookie Preferences", and a copyright notice: "Copyright © 2019, Oracle and/or its affiliates. All rights reserved."

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Select the API keys from the left-hand menu, and then “Add Public Key”

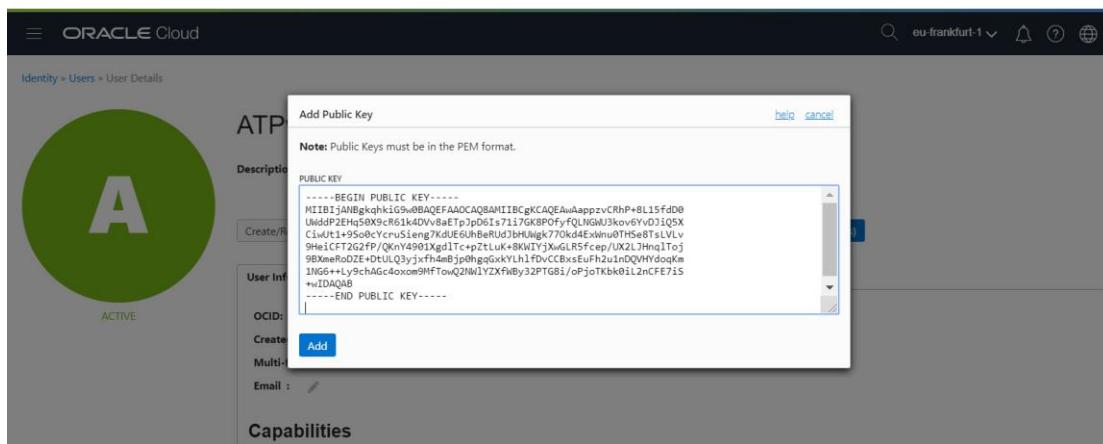
The screenshot shows the Oracle Cloud Identity User Details page for a user named ATPworkshop01. The user has a green circular profile picture with a white letter 'A' and the word 'ACTIVE' below it. The user's name is ATPworkshop01 and the description is 'ATP Workshop user account - ATPWorkshop01'. Below the user information, there are tabs for 'User Information' and 'Tags', with 'User Information' selected. Under 'User Information', the OCID is listed as 'OCID: ...j0mitq' with 'Show Copy' and 'Edit' buttons. The user was created on 'Tue, 28 May 2019 17:14:51 GMT'. Multi-factor authentication is disabled. The email field is empty. To the right, the status is 'Active' and federated is 'No'. Below this, under 'Capabilities', Local password, API keys, and Auth tokens are all set to 'Yes'. SMTP credentials and Customer secret keys are also 'Yes'. On the left sidebar, under 'Resources', 'API Keys (0)' is highlighted with a red box. The main content area shows the 'API Keys' section with a sub-header 'Add Public Key'. A message states 'There are no API Keys for this User.' with a large red-bordered 'Add Public Key' button.

From your VM desktop open a terminal and navigate to `/home/oracle/labScripts/lab3/key`

Using you mouse, copy and paste the contents of the **public** key into the ‘Add Public Key’ dialogue box

```
[oracle@atpvm01 ~]$ cd /home/oracle/labScripts/lab3/key
[oracle@atpvm01 key]$ ls
oci_api_key.pem  oci_api_key_public.pem
[oracle@atpvm01 key]$ cat oci_api_key_public.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEAAQCAQ8AMIIBCgKCAQEAtsaFAs1rGE7bhN3JX0xh
sgcSaxKE65KiCFlf0jKB0jeWyQ4USY3/gdgsfM/lEdUcCA/dF9oQAGt/lmmCt9h
4NgknP4xcX7TsqtZETkW+1maHo0G9uSD8E4x0JXPi0QwZVqnj8aabM7riwNiAfQ9
MQSs/vvqukN0ZoPKI2i99sGsqfSG0E2ycsTu6vbmYpj55oGJAJPh9p1P3hwyHvuc
C4c/C+P75fa60hTjEV06U2DWGPgeET77Z0VBGuPEEQaA062B2IUoNnjMXkZrAepI
CTqr9ciobCtl3s51VFQGDJq1o6eglUFJvz22KhGG/Q4MG+xuv2zHiq4U0kzif9yk
fwIDAQAB
-----END PUBLIC KEY-----
```

Oracle Autonomous Transaction Processing Hands-on Lab Guide



Press “Add” in the dialogue box once the public key has been pasted in.

The key is added to the user and the fingerprint is displayed in the console. Each user can have up to three API keys.

A screenshot of the 'API Keys' section. It shows a single API key named 'PK' with a green circular icon. The details for the key are: Fingerprint: 7a:77:17:62:46:f9:e9:37:11:1b:04:29:1b:2b:a2:d1, Time Created: Thu, 20 Jun 2019 10:54:39 GMT. At the bottom right, there are three small black dots indicating more keys.

If you are interested in this topic, Appendix D contains a step by step guide on how to set up REST for your Autonomous Transaction Processing instance.

The REST script will use the private key of the pair to authenticate itself against your OCI user, which now has the public key loaded.

We have written a wrapper script to call the oci-curl script. You will need to provide the wrapper script with two arguments, which are the OCID of your user and the OCID of your database.

Locating your user OCID

The User OCID can be located by selecting the user icon at the top right of the cloud console.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Cloud interface for managing Autonomous Databases. On the left, there's a sidebar with 'Autonomous Database' and 'List Scope' sections. Under 'List Scope', 'COMPARTMENT' is set to 'ATP_Delegate' and 'Tenancy' is 'oscemea001 (root)/ATP_Workshop/ATP_Delegate'. Below this is a search bar with placeholder text 'Don't see what you're looking for? (i)'. The main area displays a table titled 'Create Autonomous Database' with columns: Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Last Modified. Two databases are listed: 'MELATPTRAIN1' (Available, 1 core, 1 TB, Transaction Processing, last modified Wed, 29 May 2019 09:04:35 GMT) and 'RESTADB1' (Available, 1 core, 1 TB, Transaction Processing, last modified Fri, 24 May 2019 19:42:33 GMT). A red box highlights the 'Profile' section in the top right, which shows 'atpworkshop01' and 'Tenancy: oscemea001'. Other options in the profile menu include 'Change Password', 'User Settings', and 'Sign Out'. At the bottom of the page are links for 'Terms of Use and Privacy' and 'Cookie Preferences', and a copyright notice: 'Copyright © 2019, Oracle and/or its affiliates. All rights reserved.'

On the following page you can use the 'Copy' link to copy the OCID of the user into the script when needed.

The screenshot shows the Oracle Cloud 'User Details' page for a user named 'ATPworkshop01'. On the left is a large green circular icon with a white letter 'A' and the word 'ACTIVE' below it. The main title is 'ATPworkshop01'. Below the title is a 'Description' field containing 'ATP Workshop user account - ATPworkshop01' with an edit icon. To the right are several buttons: 'Create/Reset Password', 'Enable Multi-Factor Authentication', 'Edit User Capabilities', 'Unblock', 'Delete' (red), and 'Apply Tag(s)'. Below these buttons is a 'User Information' section with tabs for 'User Information' (selected) and 'Tags'. Under 'User Information', the 'OCID' field contains the value '...lomltq' with a 'Show' and a 'Copy' link (both highlighted with a red box). Other details shown include 'Created: Tue, 28 May 2019 17:14:51 GMT', 'Multi-factor authentication: Disabled', 'Email: ...', 'Status: Active', and 'Federated: No'.

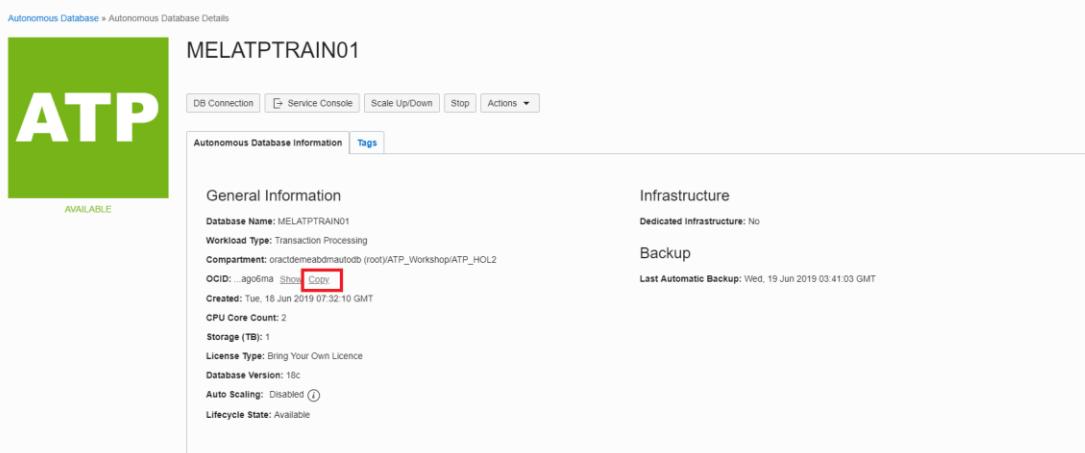
The OCID of a user is a unique reference for that user. It is a long string. An example of one is below:

ocid1.user.oc1..aaaaaaaaaha4hzarjri6f457ke46bd13gevqb6ogyjlvwzx4skbzws2lomltq

Locating the database OCID

The database OCID can be located from your database's details page. When ready you can use the 'Copy' link to copy the value and paste it into the script.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



MELATPTRAIN01

Autonomous Database Information Tags

General Information

- Database Name: MELATPTRAIN01
- Workload Type: Transaction Processing
- Compartment: oractomedesabmautodb (root)/ATP_Workshop/ATP_HOL
- OCID: ...ag6ma Show **Copy**
- Created: Tue, 18 Jun 2019 03:32:10 GMT
- CPU Core Count:
- Storage (TB):
- License Type: Bring Your Own Licence
- Database Version: 18c
- Auto Scaling: Disabled ⓘ
- Lifecycle State: Available

Infrastructure

- Dedicated Infrastructure: No
- Backup
- Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT

The OCID of a database is a unique reference for that database. It is a long string. An example of one is below:

```
ocid1.autonomousdatabase.oc1.eu-frankfurt-1.abtheljsciv7ipmz6tpqlqs3poqp7v62z75utn466bvbq6lfxq5x3hfn2roq
```

Execute the script to scale the database



From your VM desktop open a terminal, navigate to /home/oracle/labScripts/lab3.

```
$ cd /home/oracle/labScripts/lab3
```

We will run the wrapper script and pass the parameters of the user OCID and the OCID of the database to scale. The wrapper script is set to reduce the database back to 1 OCPU.

```
$ ./RESTscale.sh <your user OCID> <your DB OCID>
```

You should be able to paste the OCID values into the terminal using **control-v** or mouse **right click paste**

```
[oracle@atpvm01 lab3]$ cd /home/oracle/labScripts/lab3
[oracle@atpvm01 lab3]$ ./RESTscale.sh ocid1.user.oc1..aaaaaaaaaha4hzarjri6f457ke
46bd13gevqb6ogyjlvwzx4skbzws2lomltq    ocid1.autonomousdatabase.oc1.eu-frankfu
rt-1.abtheljsciv7ipmz6tpqlqs3poqp7v62z75utn466bvbq6lfxq5x3hfn2roq
```

As well as the user OCID and database OCID provided the script uses a json file that contains how the database should be scaled. The script displays this information to the screen. The script then calls the oci-curl routine, which generates the actual curl command that will be used.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The script will call the following JSON file (/home/oracle/labScripts/lab3/scale.json) that details how to scale the DB instance
Displaying contents of scale.json file

```
{  
    "cpuCoreCount" : 1  
}
```

Calling the oci-curl script with the json file and database OCID provided

```
oci-curl database.eu-frankfurt-1.oraclecloud.com put /home/oracle/labScripts/lab3/scale.json /20160918/autonomousDatabases/ocid1.autonomousdatabase-frankfurt-1.abtheljsciv7ipmz6tpqlqs3poqp7v62z75utn466bvbq6lfxq5x3hfn2roq
```

Curl command constructed... Ready to execute:

```
curl --data-binary @/home/oracle/labScripts/lab3/scale.json -Ss -X PUT https://database.eu-frankfurt-1.oraclecloud.com/20160918/autonomousDatabases/ocid1.autonomousdatabase.oc1.eu-frankfurt-1.abtheljsciv7ipmz6tpqlqs3poqp7v62z75utn466bvbq6lfxq5x3hfn2roq -H date: Thu, 30 May 2019 12:15:27 GMT -H x-cs-ha256: X1ErUhX02lqo/mafa0f/GioJ+awXx4BxCs5Z/9y1BaY= -H content-type: application/json -H content-length: 24 -H Authorization: Signature vei_bgpw3aps47a/61:85:ae:a6:03:3b:8e:da:27:76:03:1f:a5:f9",algorithm="rsa-sha256",headers="(request-target) date host x-content-sha256 content-type",signature="qqT9wyZgBo1XvJtAT4syESUhf/TSGCsLmvyFkkKyVL0rG1Usr65EoK/5gt9jmp+uAomjp460d52g3Jc2gCow/YwLbjuxH1ZvHZZqaDKY3xW/7zoECG3imPUoufrk7KkuGq8jtrPfVnb7d5+C0ncf0vEyIBeuzafrWPVjYn1LOtTntZ2h8tPQ6ydmMge81KFLSgcptRFAn83CEly1ZotMH0tOZ99j5vc6g14x0KVaHy+LFqsDpIglbqfius5gpG1vxF0dzz9EbGfrR1r0MNruG91N/C4UuoG2lHDx0W0#FXaheY6IYXTR3ScMJsCR8JwMQ=="
```

Do you want to send this command? [y/N]

At this point type 'Y' to allow the REST command to be executed.

The script will output that the database is being scaled:

```
-----  
{  
    "additionalDatabaseStatus" : null,  
    "autonomousContainerDatabaseId" : null,  
    "compartmentId" : "ocid1.compartment.oc1..aaaaaaaaa67l77h7jtc7d4zlwp4hyrhkkurysui07fg3s62z3llhc7lxqxa",  
    "connectionStrings" : {  
        "allConnectionStrings" : {  
            "HIGH" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_high.atp.oraclecloud.com",  
            "MEDIUM" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_medium.atp.oraclecloud.com",  
            "LOW" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_low.atp.oraclecloud.com",  
            "TPURGENT" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_tpurgent.atp.oraclecloud.com",  
            "TP" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_tp.atp.oraclecloud.com"  
        },  
        "dedicated" : null,  
        "high" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_high.atp.oraclecloud.com",  
        "low" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_low.atp.oraclecloud.com",  
        "medium" : "adb.eu-frankfurt-1.oraclecloud.com:1522/a0wer55a85ido6t_melatptrain1_medium.atp.oraclecloud.com"  
    },  
    "connectionUrls" : null,  
    "cpuCoreCount" : 2,  
    "dataStorageSizeInTBs" : 1,  
    "dbName" : "MELATPTRAIN1",  
    "dbVersion" : "18.4.0.0",  
    "dbWorkload" : "OLTP",  
    "definedTags" : { },  
    "displayName" : "MELATPTRAIN1",  
    "freeformTags" : { },  
    "id" : "ocid1.autonomousdatabase.oc1.eu-frankfurt-1.abtheljsciv7ipmz6tpqlqs3poqp7v62z75utn466bvbq6lfxq5x3hfn2roq",  
    "isAutoscalingEnabled" : false,  
    "isDedicated" : false,  
    "isPreview" : false,  
    "licenseModel" : "LICENSE_INCLUDED",  
    "lifecycleDetails" : null,  
    "lifecycleState" : "SCALE_IN_PROGRESS",  
    "serviceConsoleUrl" : "https://adb.eu-frankfurt-1.oraclecloud.com/console/index.html?tenant_name=OCID1.TENANCY.OC:PIPE4LMOW7FRRN5F3EGPG3XGUR6V2Q2WGVB3ID4EHWUJNPUSMB5Q&database_name=MELATPTRAIN1&service_type=ATP",  
    "sqlWebDeveloperUrl" : null,  
    "timeCreated" : "2019-05-29T09:04:35.261Z",  
    "usedDataStorageSizeInTBs" : 1,  
    "whitelistedIps" : null  
}
```

The script will poll every ten seconds to see if the scale operation has completed.
Once the operation has completed the script will show the updated database details via a further REST call, and then exit.

The database console will also show that the database is being scaled:

Oracle Autonomous Transaction Processing Hands-on Lab Guide

MELATPTRAIN01

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

- Database Name: MELATPTRAIN01
- Workload Type: Transaction Processing
- Compartment: oractdemesadbautodb (root)/ATP_Workshop/ATP_HOL2
- OCID: ...ag06ma [Show](#) [Copy](#)
- Created: Tue, 18 Jun 2019 07:32:10 GMT
- CPU Core Count: 2
- Storage (TB): 1
- License Type: Bring Your Own Licence
- Database Version: 18c
- Auto Scaling: Disabled ⓘ
- Lifecycle State: **Scaling In Progress**

Infrastructure

- Dedicated Infrastructure: No
- Backup
- Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT

After a short while the database scaling operation will complete and your database will be utilizing 1 OCPU.

MELATPTRAIN1

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

Workload Type: Transaction Processing

Display Name: MELATPTRAIN1

Database Name: MELATPTRAIN1

CPU Core Count: 1

Storage (TB): 1

Created: Wed, 29 May 2019 09:04:35 GMT

Compartment: oscemea001 (root)/ATP_Workshop/ATP_Delegate

OCID: ...fn2roq [Show](#) [Copy](#)

License Type: License Included

Lifecycle State: **Available**

This concludes lab 3 on scaling an Autonomous Transaction Processing Cloud Instance. If the long running SQL statement, that we started at the beginning of this lab, has not completed, just let it continue to run as you move to the next lab.



Lab 4 - Managing and monitoring ATP



Lab 4. Managing and Monitoring Autonomous Transaction Processing

Objectives:

- Learn how to use the Autonomous Transaction Processing (ATP) Cloud Service Console
- Learn how to monitor the performance of your ATP instance
- Learn how to monitor individual SQL statements
- Learn how to stop and start your ATP instance
- Learn about the ATP backup and restore processes

In this section you will use the Cloud Console and the service console to manage and monitor your ATP instance. You will learn how you can stop and start your instance, and how you could recover your database if needed.

You will use the service console to monitor the performance of your ATP instance.

About Backup and Recovery on Autonomous Transaction Processing

ATP automatically backs up your database for you. The retention period for automatic and manual backups is 60 days. You can restore and recover your database to any point-in-time in this retention period.

Manual Backups

You do not have to do any manual backups for your database as ATP backs up your database automatically. You can do manual backups using the cloud console; for example, if you want to take a backup before a major change to make restore and recovery faster. The manual backups are put in a Cloud Object Storage bucket in your tenancy, which you must configure before you carry out manual backups. When you initiate a point-in-time recovery ATP decides which backup to use for faster recovery.

Recovery

You can initiate recovery for your ATP database using the cloud console. ATP automatically restores and recovers your database to the point-in-time you specify.

You can see what backups have been created for your instance on the ATP Database Details page. As your instances were only provisioned today, there will be no backups. A sample output is below.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Cloud Service Console for the MELATPTRAIN01 ATP instance. The top navigation bar includes 'DB Connection', 'Service Console' (which is highlighted), 'Scale Up/Down', 'Stop', and 'Actions'. Below the navigation is a tabs section with 'Autonomous Database Information' and 'Tags'. The main content area is divided into two sections: 'General Information' and 'Infrastructure'. 'General Information' displays details like Database Name (MELATPTRAIN01), Workload Type (Transaction Processing), Compartment (oraclemeabdautodb (root)/ATP_Workshop/ATP_HOL2), OCID (ag06ma...), and Creation Date (Tue, 18 Jun 2019 07:32:10 GMT). 'Infrastructure' shows Dedicated Infrastructure (No) and a redboxed 'Backup' section with the message 'Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT'. Below this is a 'Backups' section with a table showing one backup entry: Name (Jun 19, 2019 03:41:03 AM UTC), State (Active), Type (Incremental, Initiated by Auto Backup), and Dates (Created: Wed, 19 Jun 2019 03:30:15 GMT, Closed: Wed, 19 Jun 2019 03:41:03 GMT).

Exploring Service Console

There are two routes to the Service Console:-

1. Connect to the Service Console for your instance by selecting it from the menu next to the name of your ATP instance on the list of ATP Instances in your compartment.

The screenshot shows the 'Autonomous Database' list page within the 'ATP_Delegate' compartment. On the left, there are filters for 'COMPARTMENT' (set to 'ATP_Delegate') and 'STATE' (set to 'Any state'). The main table lists several autonomous databases: MELATPTRAIN01, dd25ATP, dd24ATP, dd23ATP, dd22ATP, and dd21ATP. Each row includes columns for Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Created date. A context menu is open for the first row (MELATPTRAIN01), with options including 'View Details', 'Service Console' (which is highlighted with a red box), 'Copy OCID', 'Apply Tag(s)', and 'Terminate'.

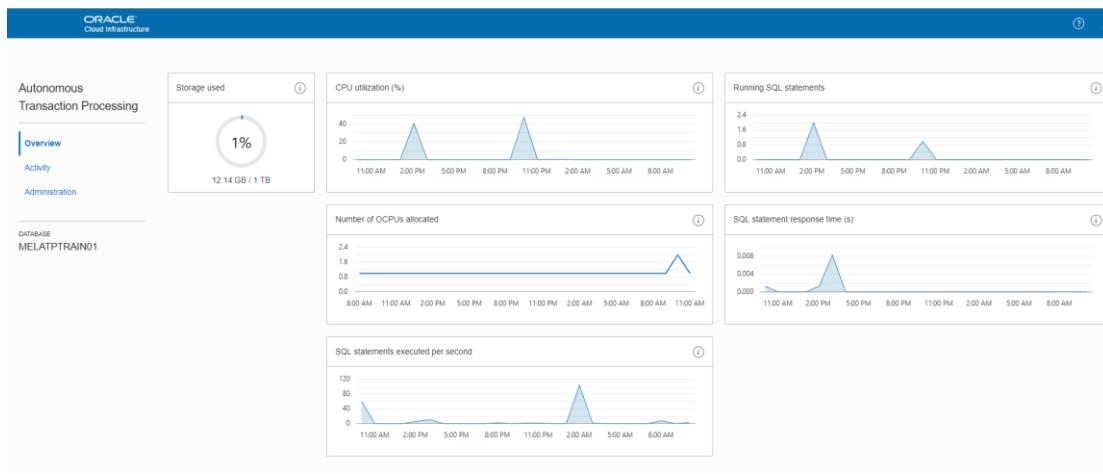
2. Or you can click the name of your instance to open the instance page and select the 'Service Console' button.

The screenshot shows the detailed view for the MELATPTRAIN01 ATP instance. The top navigation bar includes 'DB Connection', 'Service Console' (highlighted with a red box), 'Scale Up/Down', 'Stop', and 'Actions'. Below the navigation is a tabs section with 'Autonomous Database Information' and 'Tags'. The main content area is divided into 'General Information' and 'Infrastructure'. 'General Information' displays details like Database Name (MELATPTRAIN01), Workload Type (Transaction Processing), Compartment (oraclemeabdautodb (root)/ATP_Workshop/ATP_HOL2), OCID (ag06ma...), and Creation Date (Tue, 18 Jun 2019 07:32:10 GMT). 'Infrastructure' shows Dedicated Infrastructure (No) and a 'Backup' section with the message 'Last Automatic Backup: Wed, 19 Jun 2019 03:41:03 GMT'.

Console Overview

The Overview page shows real-time and historical information about the utilization of the service. Your page may have 'No Data to Display' in the monitoring panes as your instance is newly provisioned.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



The components on this page are:

- **Storage:** This chart shows the total and used storage capacity of the service. It indicates what percentage of the space is currently in-use.
- **CPU utilization (%):** This chart shows the historical CPU utilization of the service.
- **Running SQL statements:** This chart shows the average number of running SQL statements historically.
- **Average SQL statement response time (s):** This chart shows the average response time of SQL statements historically.
- **Number of OCPUs allocated**

This chart shows the number of OCPUs allocated:

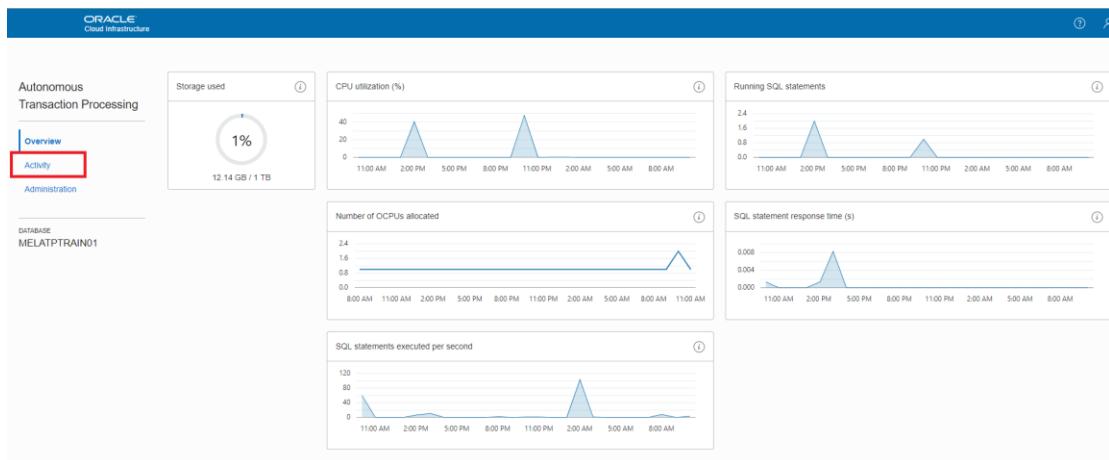
- **Auto Scaling Disabled:** For databases with auto scaling disabled, for each hour the chart shows the number of OCPUs allocated to the database if the database is open for at least some part of the hour.
- **Auto Scaling Enabled:** For databases with auto scaling enabled, for each hour the chart shows the average number of OCPUs used during that hour if that value is higher than the number of OCPUs provisioned. If the number of OCPUs used is not higher than the number of OCPUs provisioned, then the chart shows the number of OCPUs allocated for that hour.
- **Stopped Database:** If the database was stopped for the full hour the chart shows 0 OCPUs allocated for that hour.

The default retention period for performance data is eight days. So, the CPU utilization, running statements, and average SQL response time charts show data for up to the last eight days by default.

Activity Page

Select the **Activity** link on the left-hand side.

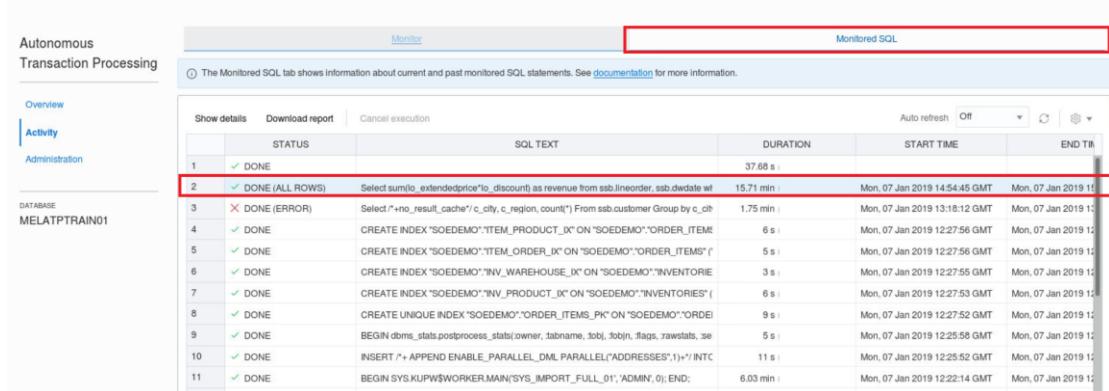
Oracle Autonomous Transaction Processing Hands-on Lab Guide



The **Activity** page defaults to a near real time view, which shows performance data for the last hour. Selecting the '**Time Period**' tab allows you to change the time slider to view historic performance data if required.

Monitored SQL

Select the '**Monitored SQL**' Tab. If you have the long running SQL statement from the previous lab shown in this table, use it to explore the interface, otherwise, just review the provided screen shots.



The **Monitored SQL** tab shows information about current and past monitored SQL statements. By default, Real-Time SQL Monitoring automatically starts when a SQL statement runs in parallel, or when it has consumed at least 5 seconds of CPU or I/O time in a single execution.

To see the detailed SQL monitor report for a statement, select the statement and right click to bring up the menu and click **Show Details**. This page can take a little while to populate with the data.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Autonomous Transaction Processing interface. On the left, there's a sidebar with 'Autonomous Transaction Processing' under 'Overview', 'Activity' (which is selected), and 'Administration'. Below that is 'DATABASE' and 'MELATPTRAIN01'. The main area is titled 'Monitor' and has a 'Monitored SQL' tab highlighted with a red box. A tooltip says: 'The Monitored SQL tab shows information about current and past monitored SQL statements. See [documentation](#) for more information.' Below the tooltip is a table with columns: STATUS, SQL TEXT, DURATION, START TIME, and END TIME. The table lists several statements, with the second one highlighted by a red box. The statement is: 'SELECT sum(l_extendedprice*l_discount) as revenue from ssb.lineorder, ssb.dwddate where lo_orderdate = d_datekey and d_year = 1993 and lo_discount between 1 and 3 and lo_quantity < 25'. The duration is 15.71 min, start time is Mon, 07 Jan 2019 14:54:45 GMT, and end time is Mon, 07 Jan 2019 14:54:45 GMT. There are three buttons for this row: 'Show details' (highlighted with a red box), 'Download report', and 'Cancel execution'.

The **Overview** tab shows general information for the statement.

This screenshot shows the 'Details for SQL ID: 0nhz7hkbcnjqn' page. It has tabs for Overview, Plan Statistics (highlighted with a red box), and Parallel. The Plan Statistics section contains several sub-sections: General, SQL text, Time & Wait Statistics, and I/O Statistics. The General section shows the status as 'DONE (ALL ROWS)', execution started at 01/07/2019 14:54:45, and execution ID 83886080. The SQL text section contains the same query as the previous screenshot. The Time & Wait Statistics section shows Duration (15.71 min), Database Time (15.8 min), and Activity % (100). The I/O Statistics section shows Buffer Gets (21.16 M), I/O Requests (165.5 K), and I/O Bytes (161.45 GB).

The **Plan Statistics** tab shows the runtime execution plan of the statement.

This screenshot shows the full execution plan for the SQL statement. The 'Plan Statistics' tab is highlighted with a red box. The plan consists of 12 operations: SELECT STATEMENT, TEMP TABLE TRANSFORMATION, LOAD AS SELECT (CURSOR DURATION MEMORY), HASH GROUP BY, KEY VECTOR CREATE BUFFERED, TABLE ACCESS STORAGE FULL, SORT AGGREGATE, HASH JOIN, VIEW, VECTOR GROUP BY, HASH GROUP BY, and KEY VECTOR USE. Each operation is shown with its line number, object name, estimated rows, actual rows, cost, timeline, executions, predicates, and pruning information.

If your statement uses parallel processing this will be documented on the **Parallel** tab

Oracle Autonomous Transaction Processing Hands-on Lab Guide

You can download the SQL Monitor report in HTML format by selecting the SQL statement, right clicking to bring up the menu and clicking 'Download report'

STATUS	SQL TEXT	DURATION	START TIME	END TIME
EXECUTING	Select /*+ PARALLEL (4) */ sum(o_extendedprice*o_discount) as revenue from sbt.lineorder, sbt.dateset where l_orderkey = d_orderkey and l_suppkey = s_suppkey and l_shipdate between '1995-01-01' and '1995-01-05' and l_status = 'O' and s_acctbal > 100 and s_name like 'SCOTT%' and d_name like 'SCOTT%' and l_qty < 25;	13 s	Mon, 07 Jan 2019 15:42:18 GMT	
DONE	Select sum(o_extendedprice*o_discount) as revenue from sbt.lineorder, sbt.dateset where l_orderkey = d_orderkey and l_suppkey = s_suppkey and l_shipdate between '1995-01-01' and '1995-01-05' and l_status = 'O' and s_acctbal > 100 and s_name like 'SCOTT%' and d_name like 'SCOTT%' and l_qty < 25;	37.68 s	Mon, 07 Jan 2019 15:44:45 GMT	Mon, 07 Jan 2019 15:44:45 GMT
DONE (ALL ROWS)	Select sum(o_extendedprice*o_discount) as revenue from sbt.lineorder, sbt.dateset where l_orderkey = d_orderkey and l_suppkey = s_suppkey and l_shipdate between '1995-01-01' and '1995-01-05' and l_status = 'O' and s_acctbal > 100 and s_name like 'SCOTT%' and d_name like 'SCOTT%' and l_qty < 25;	15.71 min	Mon, 07 Jan 2019 14:54:45 GMT	Mon, 07 Jan 2019 15:10:12 GMT
DONE (ERROR)	Select /*+ no_reuse */ sum(o_extendedprice*o_discount) as revenue from sbt.lineorder, sbt.dateset where l_orderkey = d_orderkey and l_suppkey = s_suppkey and l_shipdate between '1995-01-01' and '1995-01-05' and l_status = 'O' and s_acctbal > 100 and s_name like 'SCOTT%' and d_name like 'SCOTT%' and l_qty < 25;	1.75 min	Mon, 07 Jan 2019 13:18:12 GMT	Mon, 07 Jan 2019 13:18:12 GMT
DONE	Create Index IX_10 ON "SOEDEMO"."ORDER_ITEMS" ("ITEM_ID");	6 s	Mon, 07 Jan 2019 12:27:56 GMT	Mon, 07 Jan 2019 12:27:56 GMT
DONE	Create Index IX_11 ON "SOEDEMO"."ORDER_ITEMS" ("ITEM_ID");	5 s	Mon, 07 Jan 2019 12:27:56 GMT	Mon, 07 Jan 2019 12:27:56 GMT
UNKNOWN	Create Index IX_12 ON "SOEDEMO"."INVENTORY" ("ITEM_ID");	9 s	Mon, 07 Jan 2019 12:27:56 GMT	Mon, 07 Jan 2019 12:27:56 GMT

This will download a html document giving details about the SQL statement (note – to view the report you need Adobe Flash plugin in your browser). If you want you can download the report and open it in your Firefox browser.

Cancelling Execution

There is a 'Cancel execution' button. This can be used to cancel the execution of currently running SQL statements. This process is shown in screen shot only.
Select the running SQL you want to cancel, and select **Cancel Execution**

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Autonomous Transaction Processing

Monitor

The Monitored SQL tab shows information about current and past monitored SQL statements. See [documentation](#) for more information.

Overview

Activity

Administration

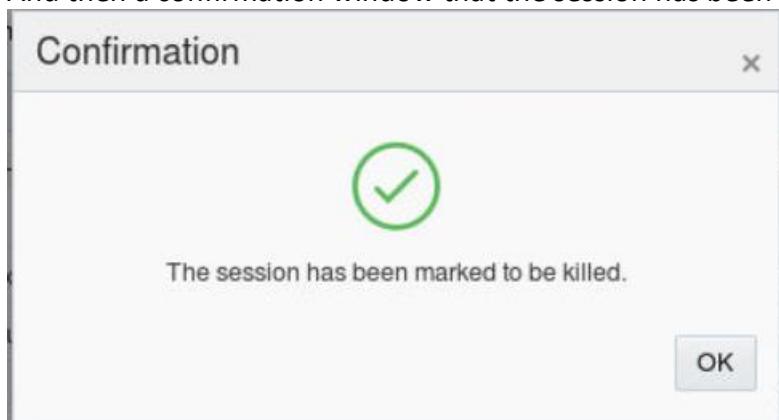
DATABASE MELATPTRAIN01

	Show details	Download report	Cancel execution	Auto refresh	Off	▼
	STATUS	SQL TEXT	DURATION	START TIME	END TIME	
1	<input checked="" type="radio"/> EXECUTING	Select.*# PARALLEL (4) # sum(lt_extendedprice*lt_discount) as revenue from stb.line	13 s	Mon, 07 Jan 2019 14:52:18 GMT		
2	<input checked="" type="radio"/> DONE		37.68 s			
3	<input checked="" type="radio"/> DONE (ALL ROWS)	Select sum(lt_extendedprice*lt_discount) as revenue from stb.lineorder, stb.delivery wif	15.71 min	Mon, 07 Jan 2019 14:54:45 GMT	Mon, 07 Jan 2019 1!	
4	<input checked="" type="radio"/> DONE (ERROR)	Selet.*# nro_result_cache# c, city, c_region, count(*) From stb.customer Group by c _	1.75 min	Mon, 07 Jan 2019 13:18:12 GMT	Mon, 07 Jan 2019 1,	
5	<input checked="" type="radio"/> DONE	CREATE INDEX "SOEDEMO"."ITEM_PRODUCT_IDX" ON "SOEDEMO"."ORDER_ITEMS"	6 s	Mon, 07 Jan 2019 12:27:56 GMT	Mon, 07 Jan 2019 1,	
6	<input checked="" type="radio"/> DONE	CREATE INDEX "SOEDEMO"."ITEM_ORDER_IDX" ON "SOEDEMO"."ORDER_ITEMS"	5 s	Mon, 07 Jan 2019 12:27:56 GMT	Mon, 07 Jan 2019 1,	
7	<input checked="" type="radio"/> DONE	CREATE INDEX "SOEDEMO"."INV_WAREHOUSE_IDX" ON "SOEDEMO"."INVENTORIES"	3 s	Mon, 07 Jan 2019 12:27:55 GMT	Mon, 07 Jan 2019 1,	
8	<input checked="" type="radio"/> DONE	CREATE INDEX "SOEDEMO"."INV_PRODUCT_IDX" ON "SOEDEMO"."INVENTORIES"	6 s	Mon, 07 Jan 2019 12:27:53 GMT	Mon, 07 Jan 2019 1,	
9	<input checked="" type="radio"/> DONE	CREATE UNIQUE INDEX "SOEDEMO"."ORDER_ITEMS_PK" ON "SOEDEMO"."ORD	9 s	Mon, 07 Jan 2019 12:27:52 GMT	Mon, 07 Jan 2019 1,	
10	<input checked="" type="radio"/> DONE	BEGIN_stmt_stabs_postprocess(stabs.ownername, tablename, tobj, tobj_flags, rawstabs, se	5 s	Mon, 07 Jan 2019 12:25:58 GMT	Mon, 07 Jan 2019 1,	
11	<input checked="" type="radio"/> DONE	INSERT.*# APPEND ENABLE_PARALLEL_DML_PARALLEL("ADDRESSES",1)*# INTO	11 s	Mon, 07 Jan 2019 12:25:52 GMT	Mon, 07 Jan 2019 1,	

There will be a Kill session pop up displayed.

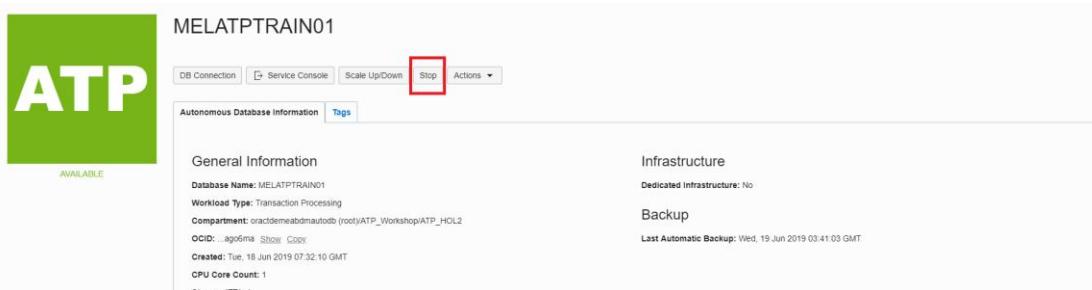


And then a confirmation window that the session has been marked to be killed.



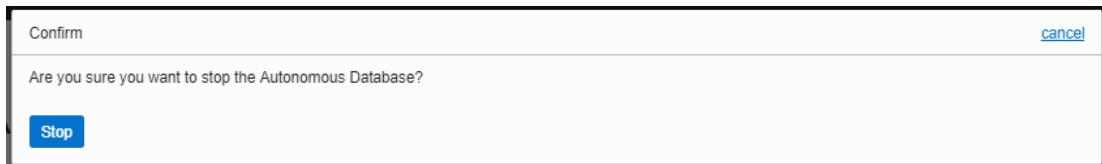
Stopping your ATP instance

Go to the page listing ATP instances in your compartment, which you used during the provisioning exercise and open the name of your ATP instance to open to instance page. Click the **Stop** button.



In the pop-up dialog confirm that you want to stop your instance by clicking on the **Stop** button.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



The instance page will now show that your instance is stopping

A screenshot of the Oracle Cloud Console showing the "Autonomous Database Details" page for an instance named "MELATPTRAIN01". The instance status is shown as "STOPPING". The "General Information" section includes details like Database Name, Workload Type, Compartment, OCID, Creation Date, CPU Core Count, Storage (TB), License Type, Database Version, Auto Scaling, and Lifecycle State. The "Infrastructure" section shows Dedicated Infrastructure set to "No" and the last automatic backup date. The "Actions" menu at the top right includes options like DB Connection, Service Console, Scale Up/Down, Stop, and Actions.

Once the instance has stopped the page will automatically refresh to show the new state



Note that stopping your instance stops metering and charging for your compute resources, based on a full-hour cycle of usage. You can come back later and start your instance instantly anytime.

Starting your ATP instance

Go to the Cloud Console you used during the previous exercise and open the Instances screen. Click the **Start** button.

Oracle Autonomous Transaction Processing Hands-on Lab Guide



The screenshot shows the Oracle Autonomous Database Details page for instance MELATPTRAIN01. The instance status is listed as "STOPPED". Below the status, there is a "Start" button which is highlighted with a red box. Other buttons visible include "DB Connection", "Service Console", "Scale Up/Down", and "Actions". The "General Information" tab is selected, showing the database name "MELATPTRAIN01". The "Infrastructure" tab indicates "Dedicated Infrastructure: No".

In the pop-up dialog confirm that you want to start your instance by clicking on the **Start** button.



The instance page will now show that your instance is starting



STARTING...

Starting the instance will take a few minutes. The Lifecycle state will change to Available when the instance has started.



AVAILABLE



Lab 5 - Data Loading into ATP



Lab 5. Data Loading into Autonomous Transaction Processing

Objectives:

- You will learn how to load a local data file into your Autonomous Transaction Processing (ATP) instance.
- You will learn how to use the DBMS_CLOUD package to load data into your ATP Instance from Object Storage.
- You will use Oracle Data Pump to load data from an export dump file.

You can load data into ATP Database using Oracle Database tools, and Oracle and 3rd party data integration tools.

You can load data:

- From files local to your client computer
- From files stored in a cloud-based object store

For the fastest data loading experience Oracle recommends uploading the source files to a cloud-based object store, such as Oracle Cloud Infrastructure Object Storage, before loading the data into your ATP Database.

To load data from files in the cloud into your ATP database, use the new PL/SQL DBMS_CLOUD package. The DBMS_CLOUD package supports loading data files from the following Cloud sources: Oracle Cloud Infrastructure Object Storage, Oracle Cloud Infrastructure Object Storage Classic, Azure Blob Storage and Amazon AWS S3.

This lab shows how to load data from Oracle Cloud Infrastructure Object Storage using two of the procedures in the DBMS_CLOUD package:

- **Create_credential:** Stores the object store credentials in your ATP database schema. You will use this procedure to create object store credentials in your ATP admin schema.
- **Copy_data:** You will use this procedure to load tables in your admin schema with data from data files staged in the Oracle Cloud Infrastructure.

The sample SQL and shell scripts for this lab are available in your VM under the directory `/home/oracle/labScripts/lab5`.

Loading a file from local storage

In this section you will load a local file from your machine into ATP:

To use as the source file, download this file to your Lab VM using Firefox:

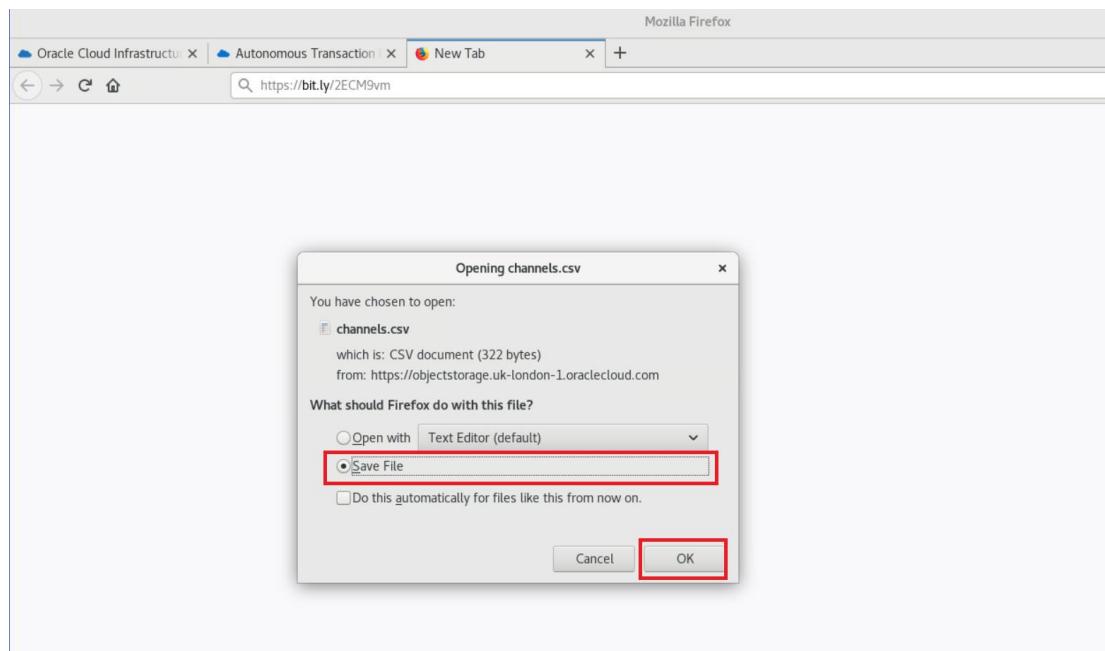
Oracle Autonomous Transaction Processing Hands-on Lab Guide

https://objectstorage.eu-frankfurt-1.oraclecloud.com/p/K4OKkjCdLHgbwegrBAN9Bv0Evy6cmQmAbD5InzeAkS0/n/oractdemeabdmautodb/b/DEMO_DATA/o/channels.csv

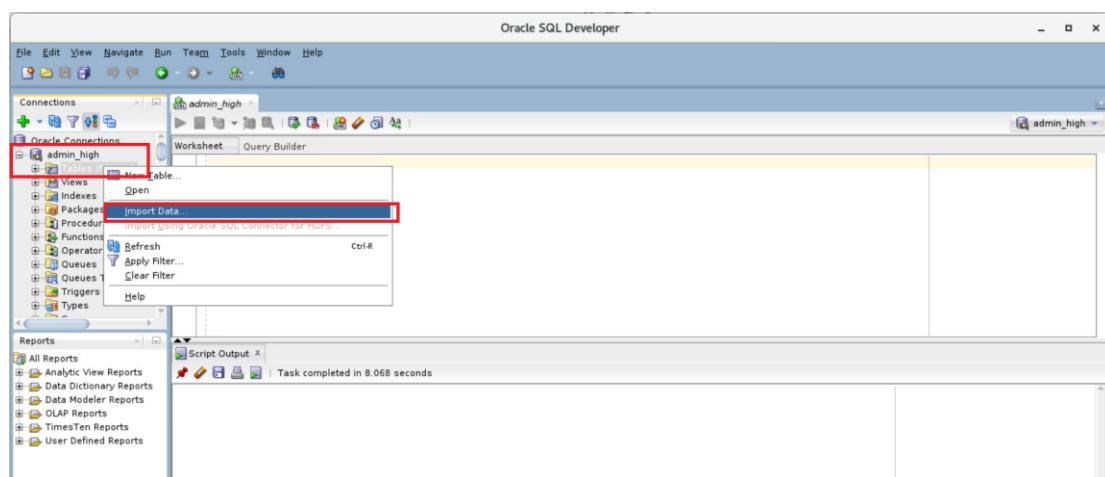
or use this short link

<https://bit.ly/2SWTpKs>

Select **Save File**. This will save the file to default downloads location for the Firefox browser, **\$HOME/Downloads**.

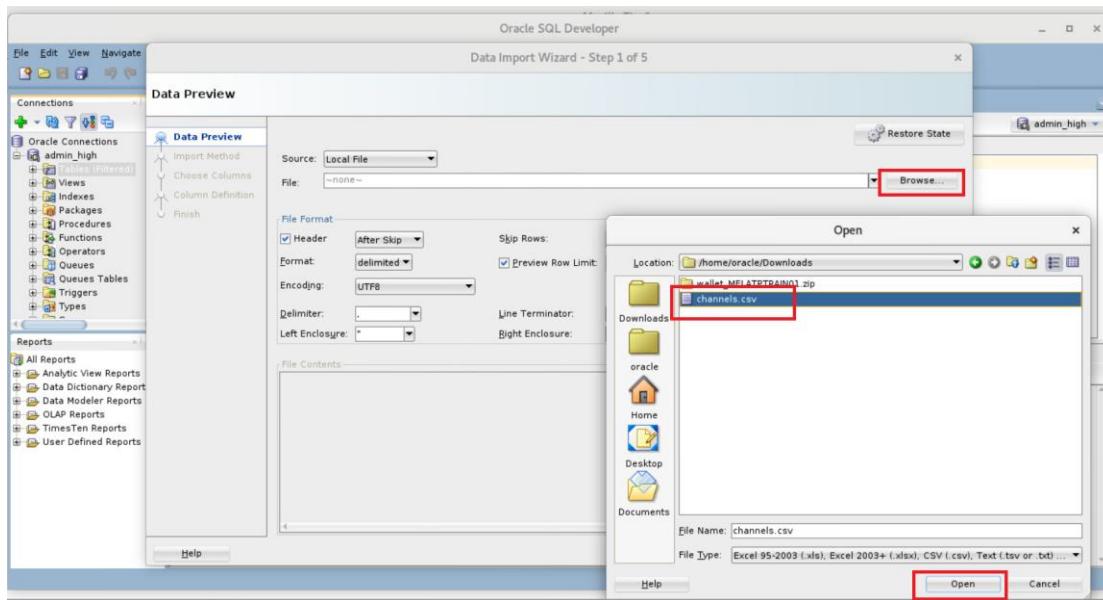


Go to SQL Developer, expand your **admin_high** connection and right click Tables, then click Import Data.

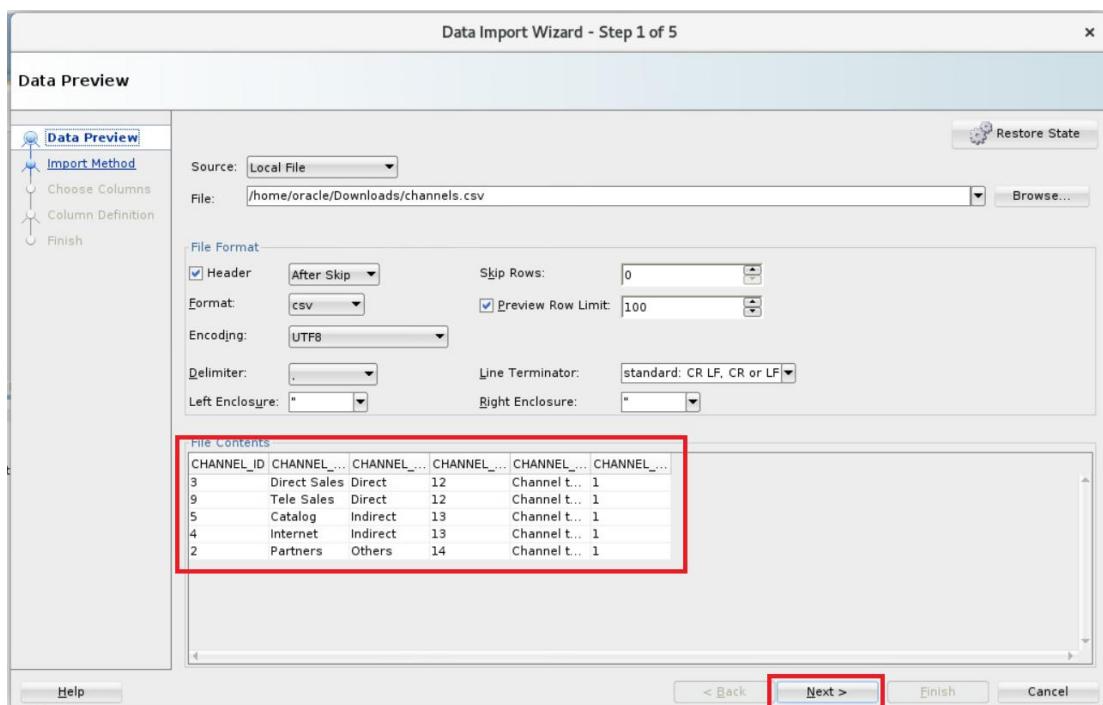


This will open the data import wizard. Click Browse and locate the channels.csv file you downloaded. (It will usually be in the Downloads folder under your home directory).

Oracle Autonomous Transaction Processing Hands-on Lab Guide

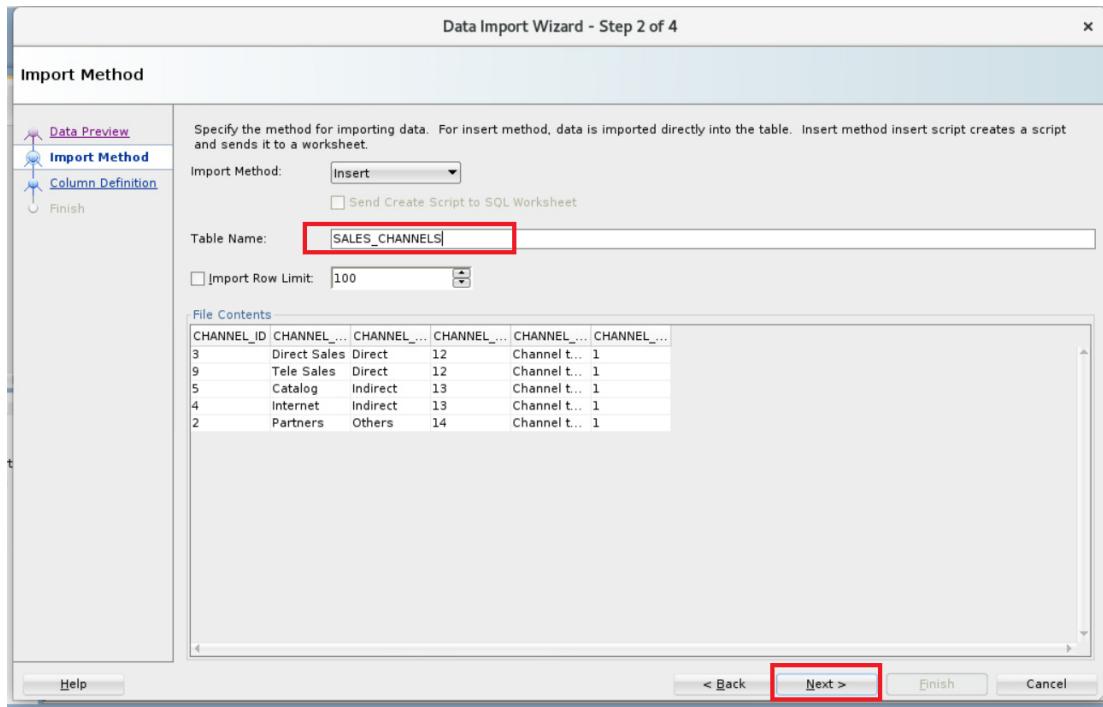


When you select the file you will see the file contents previewed in the import wizard.

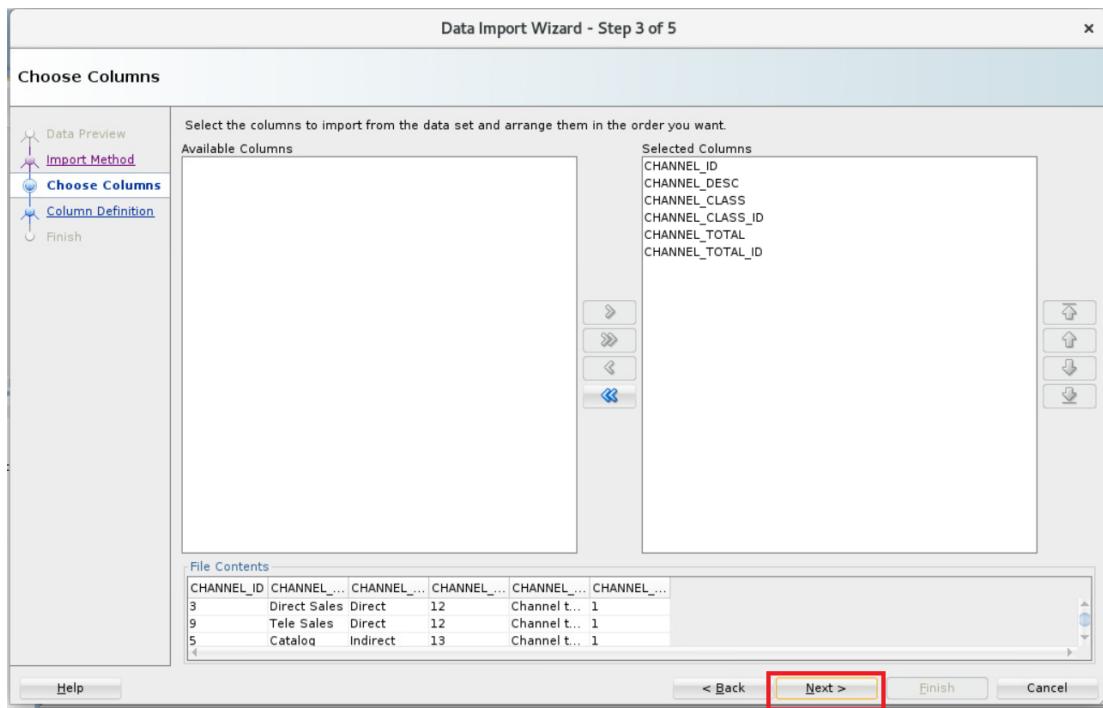


Click Next. In the next screen enter SALES_CHANNELS as the table name you will create and load into.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

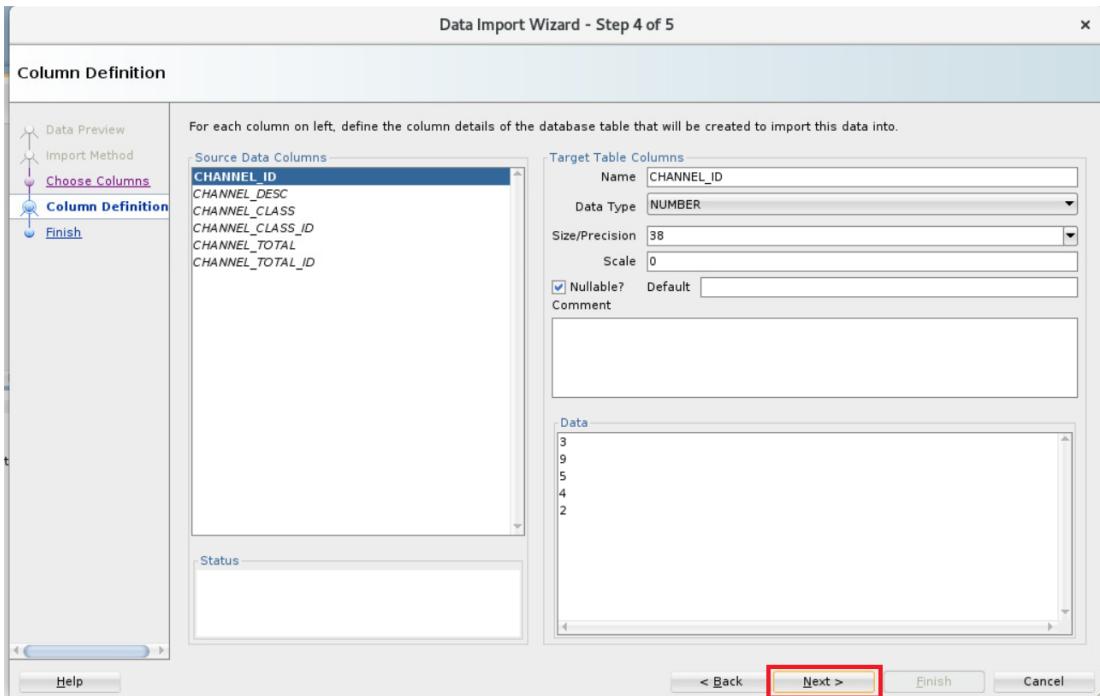


Click Next. The next screen allows you to select the columns you want for this table. For this exercise leave the columns as-is which means the table will have all columns available.

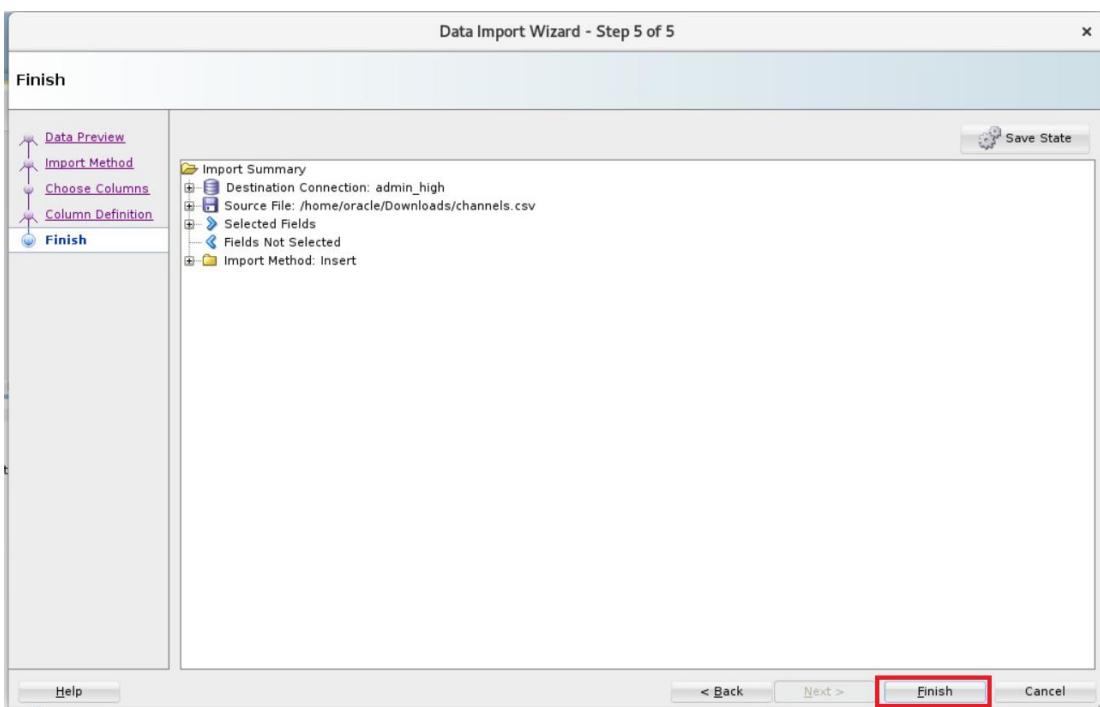


Click Next. The next screen allows you to look at the data types for each column. You can change the data types if you need to. For this exercise leave the data types as default.

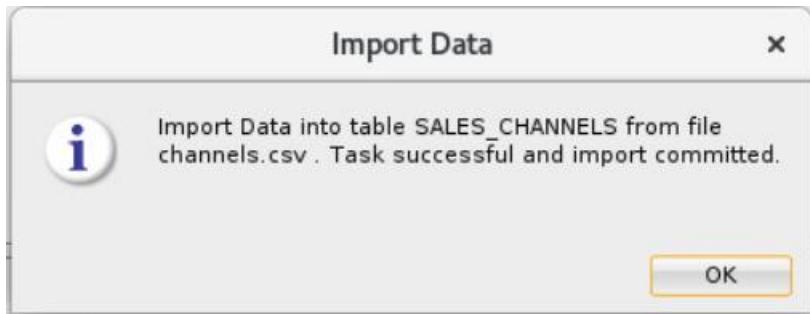
Oracle Autonomous Transaction Processing Hands-on Lab Guide



Click Next. The next page will display a summary for the import operation.



Click Finish to complete the import wizard and start the data load. When the data load finishes you will see a message saying the import was completed.



Your source file is now loaded into ATP. You can run a query on the table in SQL Worksheet to see your data.

```
select * from sales_channels;
```

A screenshot of the Oracle Database SQL Worksheet interface. The top navigation bar shows 'admin_high'. The 'Worksheet' tab is selected. In the editor area, the query 'select * from sales_channels;' is typed. Below the editor is the 'Script Output' tab, which displays the results of the query. The results show five rows of data from the SALES_CHANNELS table:

CHANNEL_ID	CHANNEL_DESC	CHANNEL_CLASS	CHANNEL_CLASS_ID	CHANNEL_TOTAL	CHANNEL_TOTAL_ID
1	3 Direct Sales	Direct		12 Channel total	1
2	9 Tele Sales	Direct		12 Channel total	1
3	5 Catalog	Indirect		13 Channel total	1
4	4 Internet	Indirect		13 Channel total	1
5	2 Partners	Others		14 Channel total	1

Loading a file from Object Storage

Information about how the environment was prepared

To load data from the Oracle Cloud Infrastructure Object Storage you will need a Cloud user with the appropriate privileges to read data from the Object Store. A user 'atp_oss_access' with the right setup and authentication token has been pre-created for you to use in the next step. If you are interested in seeing an outline of how it was setup, this is documented in Appendix B.

Storing your object store authentication token credentials in the database

To access data in the Object Store you must enable your database user to authenticate itself with the Object Store using your object store account and authentication token.

You do this by creating a private CREDENTIAL object for your user that stores this information encrypted in your ATP instance. This encrypted connection information is only usable by your user schema.

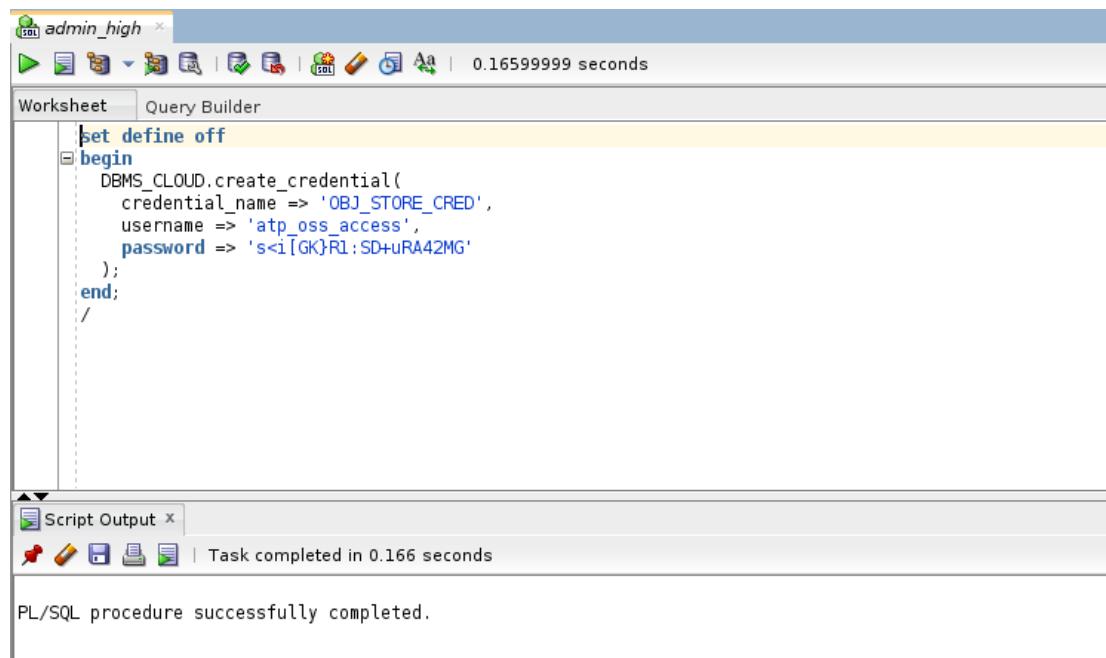
Within the SQL worksheet of SQL Developer in your **admin_high** connection, execute the following code to store the object store credential in the database:

The below sql can be found in the file

/home/oracle/labScripts/lab5/lab5_create_credential.sql

```
set define off
begin
  DBMS_CLOUD.create_credential(
    credential_name => 'OBJ_STORE_CRED',
    username => 'atp_oss_access',
    password => 's<i[GK}R1:SD+uRA42MG'
  );
end;
/

```



The screenshot shows the Oracle SQL Developer interface with the 'admin_high' connection selected. The 'Worksheet' tab is active, displaying the PL/SQL code used to create a credential. The code is as follows:

```
set define off
begin
  DBMS_CLOUD.create_credential(
    credential_name => 'OBJ_STORE_CRED',
    username => 'atp_oss_access',
    password => 's<i[GK}R1:SD+uRA42MG'
  );
end;
/

```

Below the worksheet, the 'Script Output' tab shows the execution results:

```
PL/SQL procedure successfully completed.
```

Now you are ready to load data from Object Store

Loading Data from Object Store

You can use the PL/SQL package DBMS_CLOUD directly to load the data from object store. It can be invoked directly or implicitly by using tools within SQL Developer.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Connect as your admin user in SQL Developer using your **admin_high** connection.

Prepare your destination table by running the create table statement

The sql can be found in **/home/oracle/labScripts/lab5/lab5_create_countries.sql**

```
CREATE TABLE COUNTRIES (
    COUNTRY_ID          NUMBER,
    COUNTRY_ISO_CODE    CHAR(2),
    COUNTRY_NAME        VARCHAR2(40),
    COUNTRY_SUBREGION   VARCHAR2(30),
    COUNTRY_SUBREGION_ID NUMBER,
    COUNTRY_REGION      VARCHAR2(20),
    COUNTRY_REGION_ID   NUMBER,
    COUNTRY_TOTAL        VARCHAR2(30),
    COUNTRY_TOTAL_ID    NUMBER,
    COUNTRY_NAME_HIST   VARCHAR2(40)
);
```

The screenshot shows the Oracle SQL Developer interface. The top menu bar has tabs for 'Welcome Page', 'admin_high', and 'COUNTRIES'. Below the menu is a toolbar with various icons. The main area is divided into two tabs: 'Worksheet' and 'Query Builder', with 'Worksheet' selected. The code in the Worksheet tab is:

```

CREATE TABLE COUNTRIES (
    COUNTRY_ID          NUMBER,
    COUNTRY_ISO_CODE    CHAR(2),
    COUNTRY_NAME        VARCHAR2(40),
    COUNTRY_SUBREGION   VARCHAR2(30),
    COUNTRY_SUBREGION_ID NUMBER,
    COUNTRY_REGION      VARCHAR2(20),
    COUNTRY_REGION_ID   NUMBER,
    COUNTRY_TOTAL       VARCHAR2(30),
    COUNTRY_TOTAL_ID    NUMBER,
    COUNTRY_NAME_HIST   VARCHAR2(40)
);

```

Below the code, the 'Script Output' tab is visible, showing the message 'Task completed in 0.056 seconds'. The status bar at the bottom right indicates '1 Line 24'.

Next, paste the following code to copy data from object storage using the DBMS_CLOUD package. We use the copy_data procedure of the DBMS_CLOUD package to copy the data (countries.csv) staged in your object store.

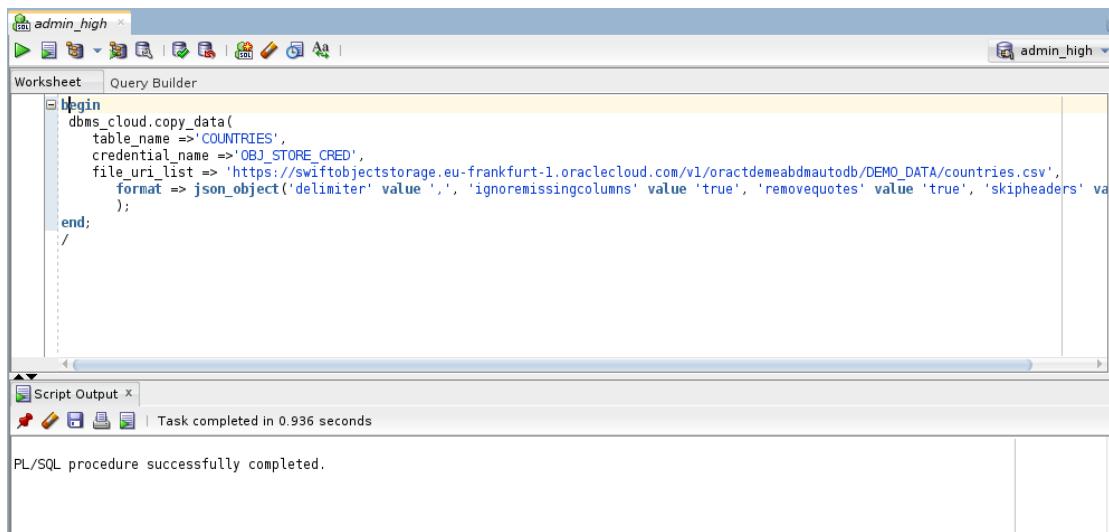
The below sql can be found under
/home/oracle/labScripts/lab5/lab5_dbms_cloud_sample.sql in your vm.

```

begin
  dbms_cloud.copy_data(
    table_name =>'COUNTRIES',
    credential_name =>'OBJ_STORE_CRED',
    file_uri_list => 'https://swiftobjectstorage.eu-frankfurt-
1.oraclecloud.com/v1/oractdemeabdmautodb/DEMO_DATA/countries.csv',
    format => json_object('delimiter' value ',',
    'ignoremissingcolumns' value 'true', 'removequotes' value 'true',
    'skipheaders' value '1')
  );
end;
/

```

Oracle Autonomous Transaction Processing Hands-on Lab Guide

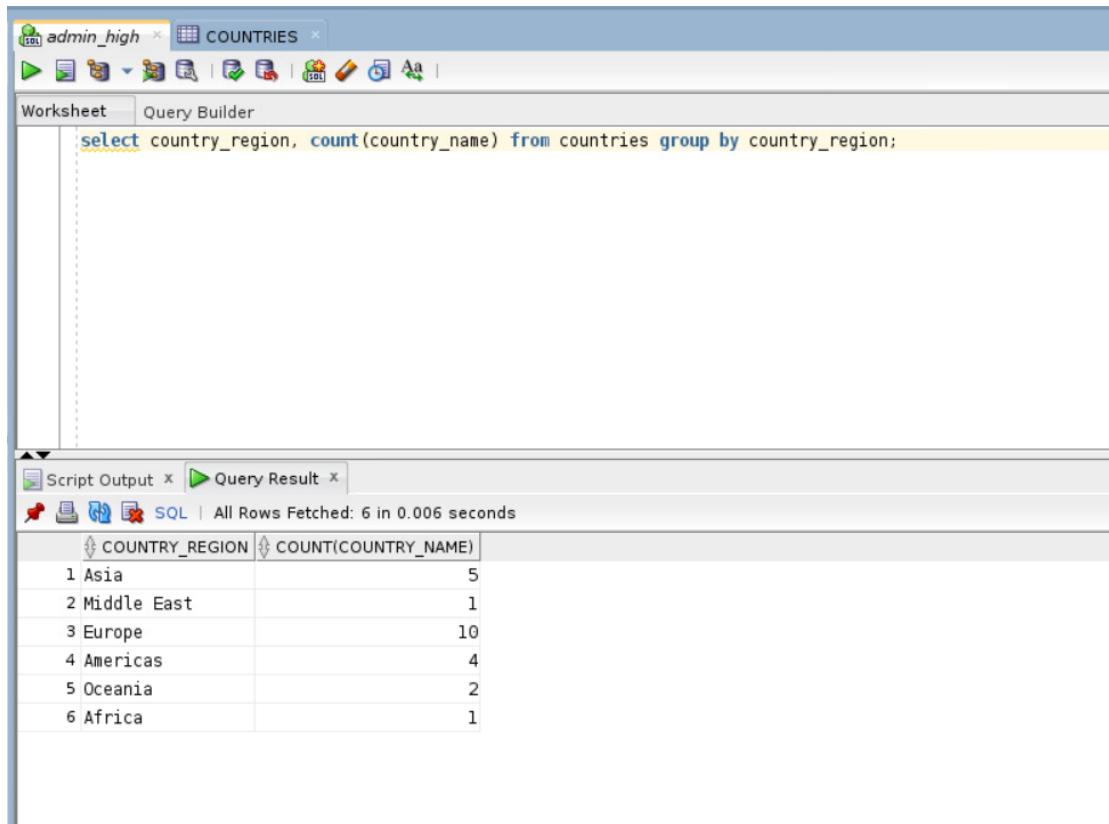


```
begin
  dbms_cloud.copy_data(
    table_name =>'COUNTRIES',
    credential_name =>'OBJ_STORE_CRED',
    file_uri_list => 'https://swiftobjectstorage.eu-frankfurt-1.oraclecloud.com/v1/oractdemeabdmautodb/DEMO_DATA/countries.csv',
    format => json_object('delimiter' value ',', 'ignoremissingcolumns' value 'true', 'removequotes' value 'true', 'skipheaders' value 'true');
  );
end;
/
```

Script Output X | Task completed in 0.936 seconds
PL/SQL procedure successfully completed.

You can verify that the data has loaded by running a simple query on your COUNTRIES table.

```
select country_region, count(country_name) from countries group by country_region;
```



```
select country_region, count(country_name) from countries group by country_region;
```

COUNTRY_REGION	COUNT(COUNTRY_NAME)
1 Asia	5
2 Middle East	1
3 Europe	10
4 Americas	4
5 Oceania	2
6 Africa	1

Importing a Data Pump Export

How the Data Pump Export was Prepared

To simulate migrating from an on-premise platform to the cloud, this export was generated on an Oracle SPARC SuperCluster in an on-premise data center, following the documentation <https://docs.oracle.com/en/cloud/paas/atp-cloud/atpug/load-data.html#GUID-30DB1EEA-DB45-49EA-9E97-DF49A9968E24>

The export command used was:

```
$ expdp system/MyDemoPa55word exclude=cluster,db_link parallel=8
schemas=soe dumpfile=exp_soe_150_%u.dmp directory=mnet
```

This created an export dump consisting of 8 files. The files were uploaded to an Object Storage bucket within the tenancy used for this lab.

The source schema was a Swingbench SOE schema, scaled at 150MB.

Creating the Destination Schema

We are going to re-map the schema from the one in the dumpfiles to a new schema. Within the SQL worksheet of SQL Developer in your **admin_high** connection, execute the following code to drop any existing schema with the same name, and create your new schema. This sql can be found in the file

/home/oracle/labScripts/lab5/lab5_create_user.sql

```
drop user soedemo cascade;
create user soedemo identified by ATPwelcome1234;
grant connect, resource to soedemo;
grant execute on dbms_lock to soedemo;
```

The screenshot shows the Oracle SQL Developer interface. The top window is titled 'admin_high'. In the 'Worksheet' tab, the following SQL script is being run:

```
drop user soedemo cascade;
create user soedemo identified by ATPwelcome1234;
grant connect, resource to soedemo;
grant execute on dbms_lock to soedemo;
```

The bottom window is titled 'Query Result'. The output shows the execution results:

```
Error starting at line : 1 in command -
drop user soedemo cascade
Error report -
ORA-01918: user 'SOEDEMO' does not exist
01918. 00000 - "user '%s' does not exist"
*Cause: User does not exist in the system.
*Action: Verify the user name is correct.

User SOEDEMO created.

Grant succeeded.

Grant succeeded.
```

Running Data Pump

Now go to a terminal window in your TigerVNC session connected to the LAB VM.

The location of the wallet file directory is not mandated by the cloud software. For this lab we will use the directory **/home/oracle/wallets** to store our wallet file. The wallet zip file should still be available on your system under your **\$HOME/Downloads** directory from the previous labs. If not, then you will have to download it again via the admin console.

```
cd $HOME
cp $HOME/Downloads/wallet_*.zip wallets
cd $HOME/wallets
unzip wallet_*.zip

[oracle@demo-long-v4 ]$ cd $HOME
[oracle@demo-long-v4 ]$ cp $HOME/Downloads/wallet_*.zip wallets
[oracle@demo-long-v4 ]$ cd $HOME/wallets
[oracle@demo-long-v4 ]$ unzip wallet_*.zip
Archive: wallet_MELATPTRAIN01.zip
  inflating: cwallet.sso
  inflating: tnsnames.ora
  inflating: truststore.jks
  inflating: ojdbc.properties
  inflating: sqlnet.ora
  inflating: ewallet.p12
  inflating: keystore.jks
[oracle@demo-long-v4 ]$ █
```

Next you edit the sqlnet.ora to set the wallet location to your /home/oracle/wallets directory

```
WALLET_LOCATION = (SOURCE = (METHOD = file) (METHOD_DATA =
(DIRECTORY="/home/oracle/wallets")))
SSL_SERVER_DN_MATCH=yes
```

Verify that LD_LIBRARY_PATH includes the instantclient_18_3 directory

```
echo $LD_LIBRARY_PATH

[oracle@demo-long-v4 wallets]$ echo $LD_LIBRARY_PATH
/home/oracle/instantclient_18_3
```

If not set it using the command

```
export LD_LIBRARY_PATH=/home/oracle/instantclient_18_3
```

Verify the TNS_ADMIN environment variable is set to /home/oracle/wallets

```
echo $TNS_ADMIN
```

```
[oracle@demo-long-v4 ~]echo $TNS_ADMIN  
/home/oracle/wallets  
[oracle@demo-long-v4 ~]■
```

If not set it using the command

```
export TNS_ADMIN=/home/oracle/wallets
```

```
[oracle@demo-long-v4 wallets]$ export TNS_ADMIN=/home/oracle/wallets
```

Go to the Oracle Instant Client directory

```
cd $HOME/instantclient_18_3
```

This installation of Oracle Instant Client includes both the Basic package and the Tools package to allow us to run impdp.

The impdp command that for your import is based on the following information.
The %U in the dump file expands to a 2-digit incrementing integer starting with 01.

Connection string: admin/<your admin password>@<your database>_high
Directory: data_pump_dir

Credential: OBJ_STORE_CRED (the same credential we used to load a file from object_storage)

REMAP_SCHEMA: SOE:SOEDEMO

Dumpfile: https://swiftobjectstorage.eu-frankfurt-1.oraclecloud.com/v1/oractdemeabdmautodb/DEMO_DATA/exp_atp_soe_150_%U.dmp

Parallel: 8

To this lab simpler, there is a wrapper shell script which collects your password and connect string and calls the 'impdp' executable. In the UNIX terminal in your VM change directory to /home/oracle/labScripts/lab5. Then execute the script lab5_import_wrapper.sh, giving the admin password and connect string on the command line.

```
cd /home/oracle/labScripts/lab5  
../lab5_import_wrapper.sh <admin password> <connect string>
```

The script will list the generated impdp command. Review this and enter 'Y' if you are happy to proceed.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

```
[oracle@demo-long-v4 ~]$ cd /home/oracle/labScripts/lab5
[oracle@demo-long-v4 ~]$ ./lab5_import_wrapper.sh ATPwelcome-1234 MELATPTRAIN01_high
This is the import command that has been generated
impdp admin/ATPwelcome-1234@MELATPTRAIN01_high directory=data_pump_dir credential=OBJ STORE CRED REMAP_SCHEMA=SOE:SOEDEMO dumpfile=https://swiftobjectstorage.eu-frankfurt-1.oraclecloud.com/v1/oractdemeabdmautodb/DEMO DATA/exp_atp_soe_150%U.dmp PARALLEL=8
Do you want to run it [Y/N]
Y

Import: Release 18.0.0.0.0 - Production on Fri Feb 1 13:11:39 2019
Version 18.3.0.0.0

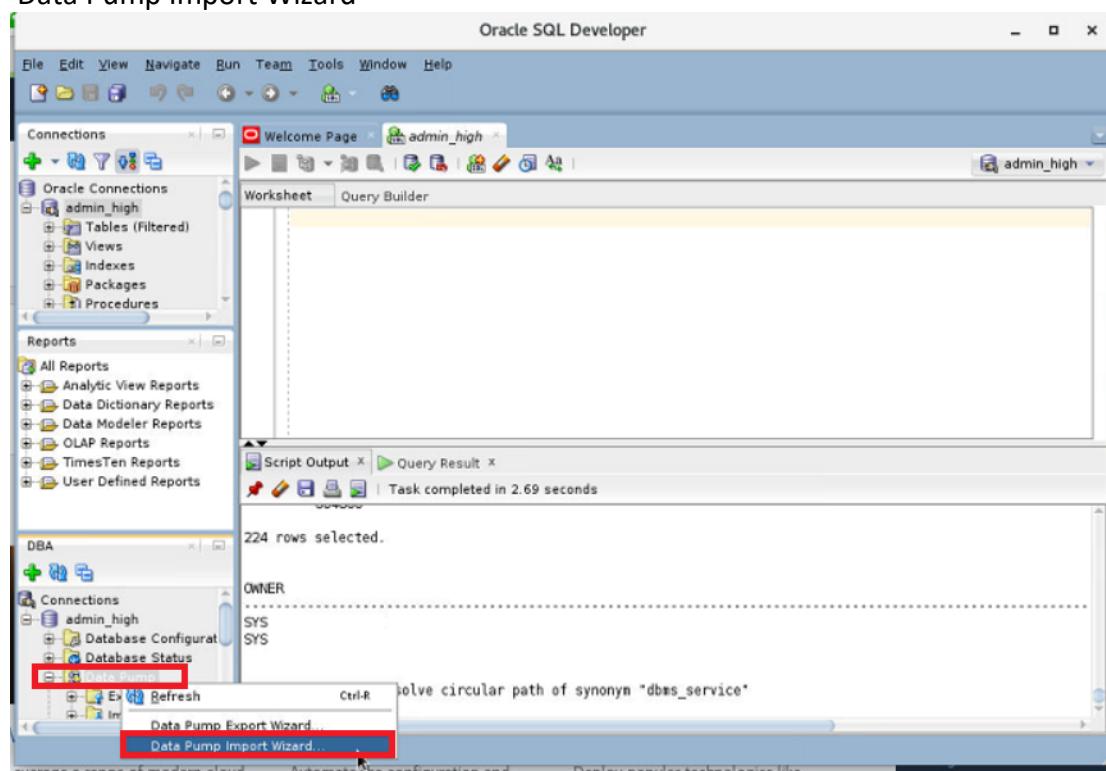
Copyright (c) 1982, 2018, Oracle and/or its affiliates. All rights reserved.

Connected to: Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
```

This import should take under 10 minutes to complete.

You will observe some errors during the import schema creation phase. One is because the schema already exists. The other is the schema originally had the 'manage scheduler' privilege, which cannot be granted by the admin user in your ATP instance. There may also be a compilation error on the Package 'ORDERENTRY'.

Note: You could also have carried out the import using the Oracle SQL Developer 'Data Pump Import Wizard'



You have successfully tested three methods of loading data into your ATP instance. For a list of all the possible methods, see the product documentation <https://docs.oracle.com/en/cloud/paas/atp-cloud/atpug/load-data.html>



Lab 6 - DBA

Exploration of ATP with SQL Developer

Lab 6. DBA Exploration of ATP with SQL Developer

Objectives:

- Explore the DBA view for Autonomous Transaction Processing (ATP) in SQL Developer

In this lab you will get an overview of the DBA view in SQL Developer and how to explore configuration and settings in your ATP environment. As part of the autonomous nature of ATP, some of these settings cannot be adjusted by the DBA.

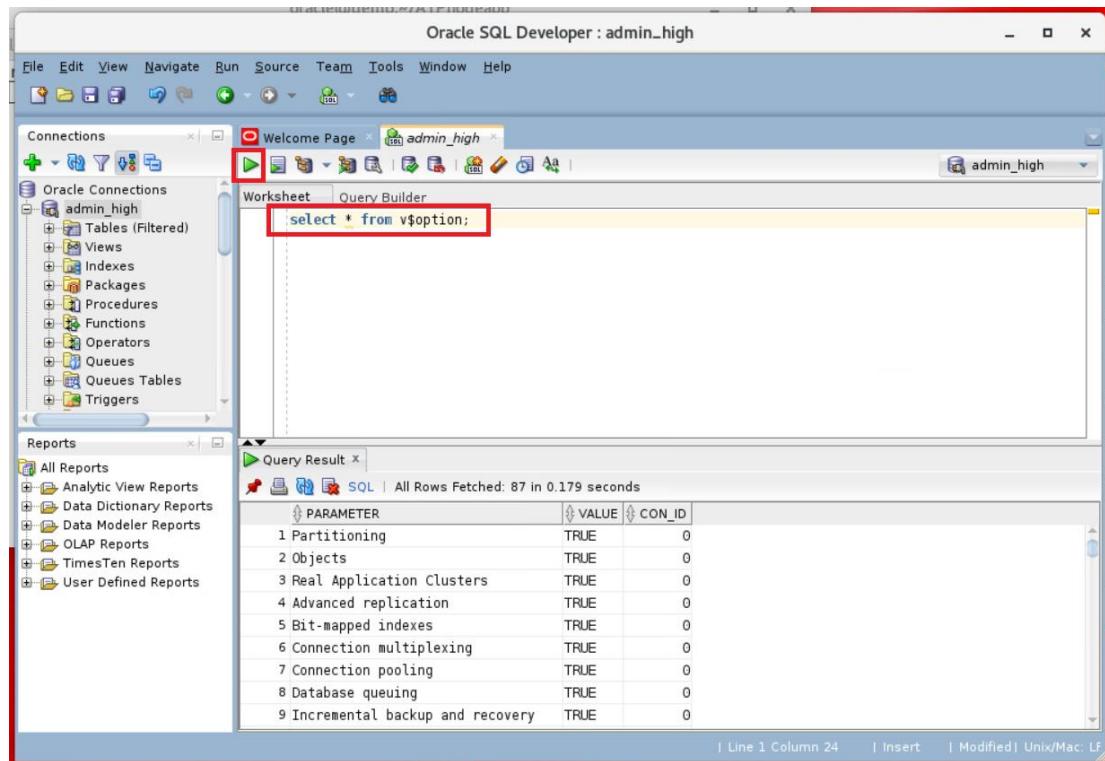
For this lab make sure you use your “ADMIN” user connection. Regular user accounts will not be able to view any system configuration information.

ATP includes many of the database options that are widely used in your databases and applications.

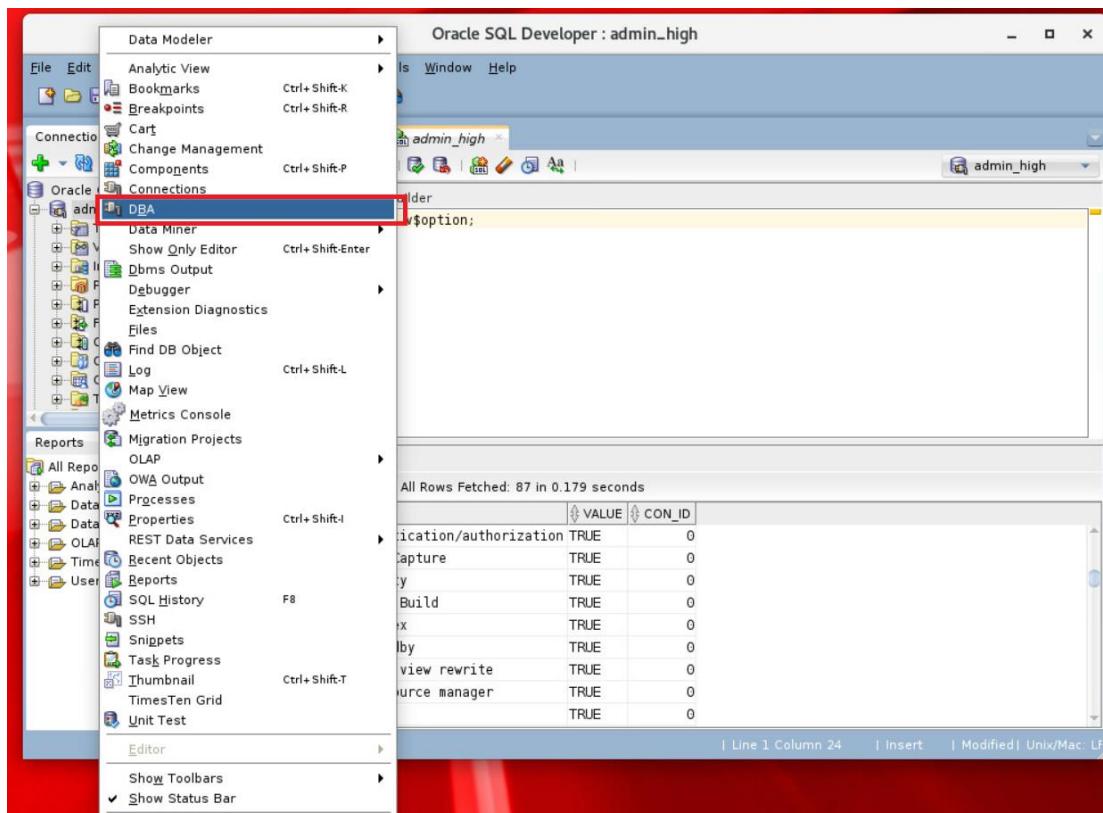
Start SQL Developer. Connect to your **admin_high** connection.

In your SQL worksheet run the following command to get a list of options in your ATP instance.

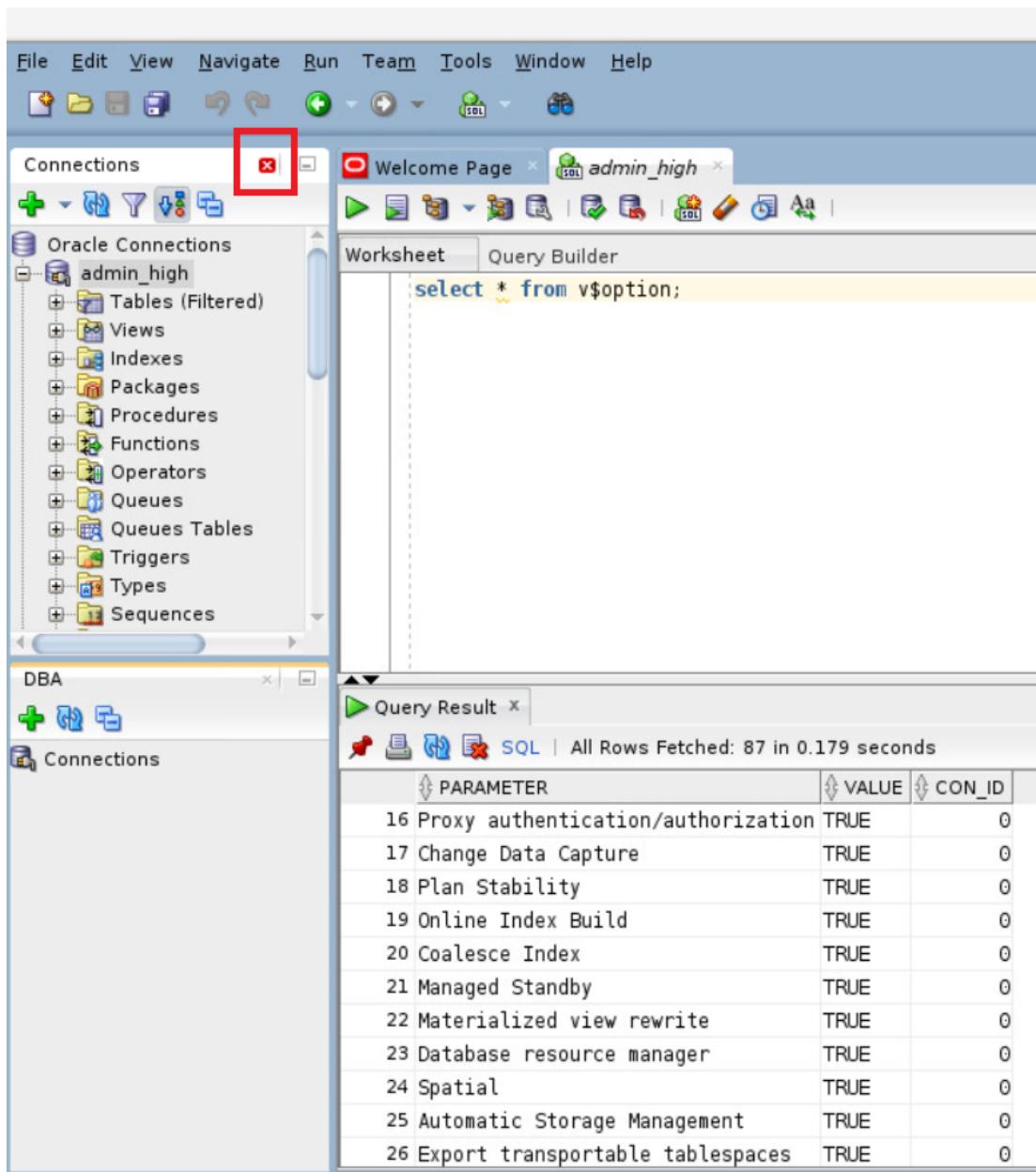
```
select * from v$option;
```



Next you will examine the DBA view in SQL Developer. In the main screen select **View->DBA**. If you cannot see the DBA option under View, go to “Window” and select “Reset Window to Factory Setting”, this should let you see the DBA option under View.



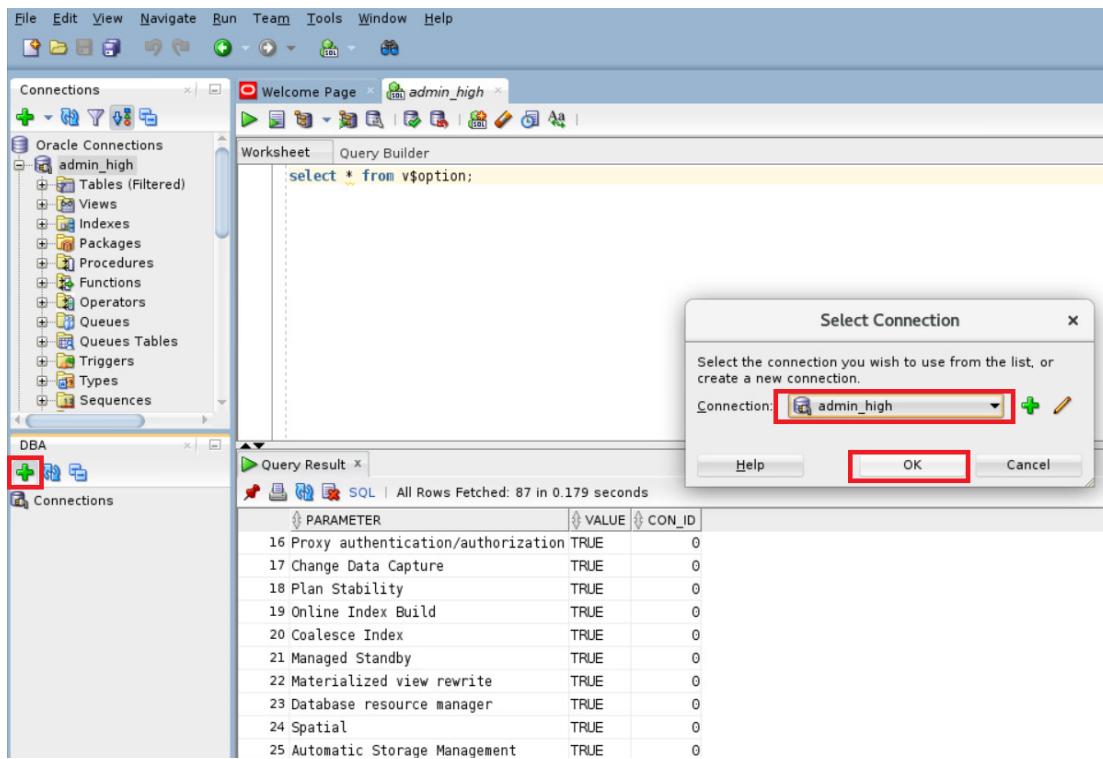
The DBA view appears in the bottom left pane. If you need more space to examine the DBA view you can close the Connections window or any other un-needed windows by clicking the ‘X’ on their menu bar.



Click the connections green plus sign in the DBA view menu.

On the connection menu pop up, select your **admin_high** connection and select **OK**.

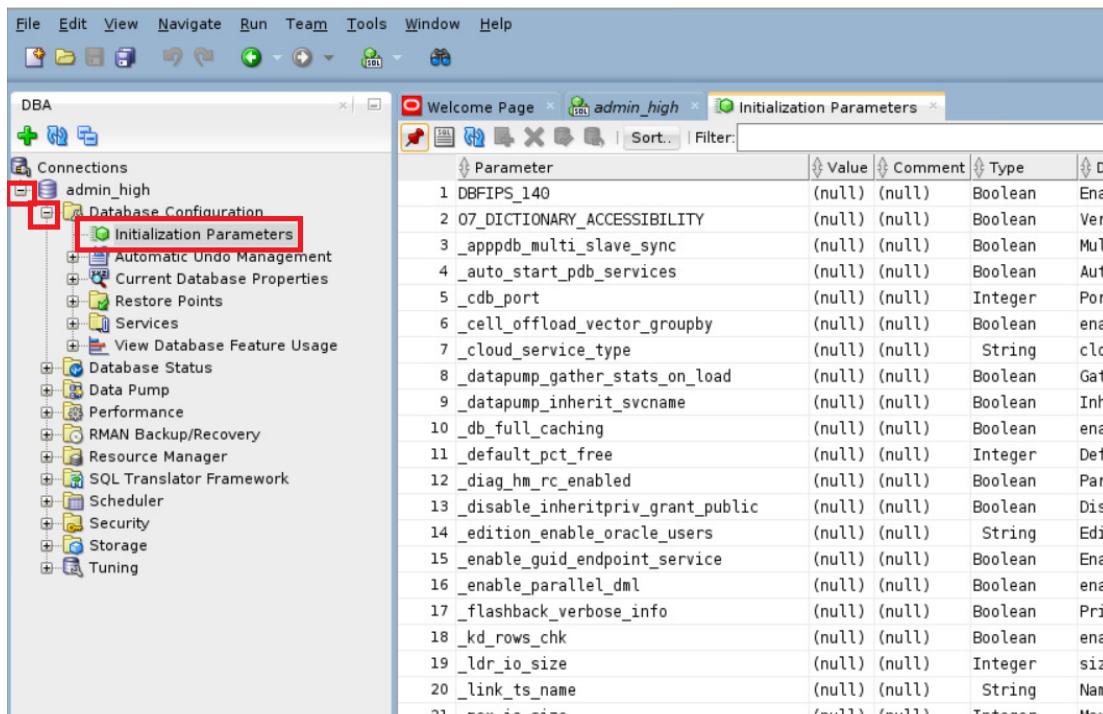
Oracle Autonomous Transaction Processing Hands-on Lab Guide



Expand your **admin_high** connection by clicking on the ‘+’ symbol.

Expand the **Database Configuration** by clicking on the ‘+’ symbol.

View the Initialization parameters by clicking on **Initialization Parameters**.



Expand **Services** by clicking on the ‘+’

Oracle Autonomous Transaction Processing Hands-on Lab Guide

You can see the services that are configured for your ATP instance.

The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : SERVICE ADMIN.BUMK28FBVPYBRGK_MELATPTRAIN01_high.atp.oraclecloud.com@admin_high". The left sidebar (DBA) has a tree view under "Connections" for "admin_high". The "Services" node is expanded, showing several service entries, with one entry highlighted by a red box. The right panel displays a table titled "Actions..." with various service parameters listed. The "NAME" column includes entries like "BUMK28FBVPYBRGK_MELATPTRAIN01_high.atp.oraclecloud.com" and "BUMK28FBVPYBRGK_MELATPTRAIN01_low.atp.oraclecloud.com". The "Value" column contains corresponding values such as "N" for DTP and "NONE" for GOAL.

In **View Database Features Usage** you can see the features that have been marked as in use. This view is not real-time and so you may not have any features marked as in use in your ATP instance.

The screenshot shows the Oracle SQL Developer interface with the title bar "Wed 17:02". The left sidebar (DBA) has a tree view under "Connections" for "admin_high". The "View Database Feature Usage" node is selected and highlighted by a red box. The main pane displays a table titled "Usage | High Water Marks" with columns: NAME, CURRENTLY_USED, DETECTED_USAGES, TOTAL_SAMPLES, FIRST_USAGE_DATE, and LAST_USAGE_DATE. The table lists numerous database features, many of which are marked as currently used (TRUE). For example, ACFS, ACFS Encryption, and ACFS Snapshot are listed with TRUE in the CURRENTLY_USED column.

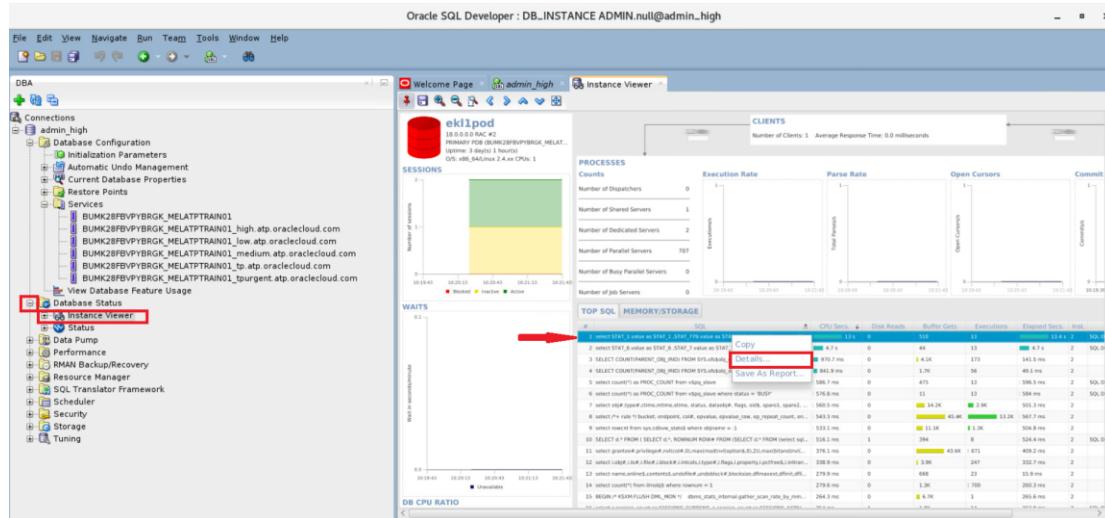
Expand **Database Status** by clicking on '+' and double click on **Instance Viewer**.

This displays a lot of information about the instance such as usage, configuration as well as the TOP SQL running on the instance.

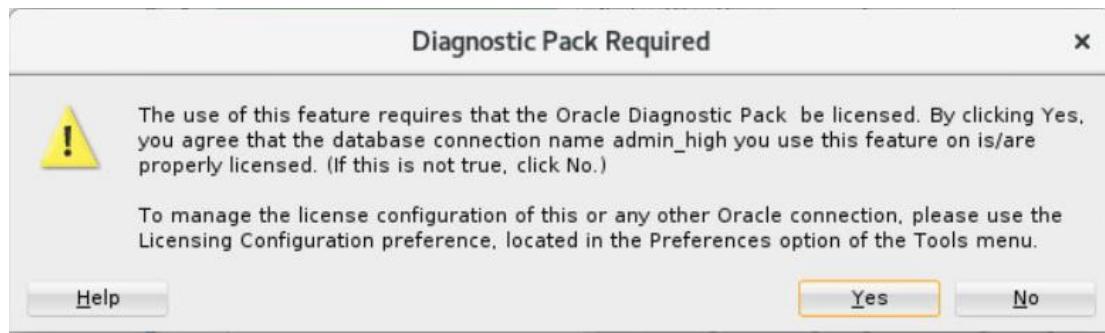
You can select a **SELECT STATEMENT** from **TOP SQL**, right click and select **Details**.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Since your instance was started it has not run many user SQL statements. This means that you may not have any statements that have interesting Explain Plan or parallelism.



You may receive a warning about Diagnostic Pack and/or Tuning Pack Click Yes on these popups to continue the lab.



This will create a new tab in the main window called SQL Detail. You can see examine the explain plan for your SQL here.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Database Control interface. On the left, the DBA navigation pane is open, showing various database management options like Connections, Database Configuration, and Services. The main pane is titled "Welcome Page" and "admin_high". The "SQL Details" tab is selected, displaying an SQL statement and its execution plan. The "Explain Plan" section contains the following table:

Query Plan	Rows	Rowsource Time(s)
1 SELECT STATEMENT Cost = 3	1	0
2 NESTED LOOPS	1	0
3 NESTED LOOPS	1	0
4 NESTED LOOPS	1	0
5 NESTED LOOPS	1	0
6 NESTED LOOPS	1	0
7 NESTED LOOPS	1	0
8 NESTED LOOPS	1	0
9 NESTED LOOPS	1	0
10 NESTED LOOPS	1	0
11 NESTED LOOPS	1	0
12 VIEW	1	0
13 SORT AGGREGATE	1	0
14 FIXED TABLE FULL X\$KSUSGOSTA	1	1
15 VIEW	1	0
16 SORT AGGREGATE	1	0
17 FIXED TABLE FULL X\$KSUSGOSTA	1	0
18 VIEW	1	0
19 SORT AGGREGATE	1	0
20 FIXED TABLE FULL X\$KSUSGOSTA	1	0
21 VIEW	1	0

The **Resource Manager** menu contains information about different consumer groups and plans defined, and the current plan in effect. Explore the entries here, most of them will have contextual activities if you select and right click on them.

Verify which plan is active by selecting **Plans**.

The screenshot shows the Oracle Database Control interface. The DBA navigation pane is open, showing various database management options. The main pane is titled "Welcome Page" and "admin_high". The "Plans" link under the Resource Manager section is selected, displaying a list of resource plans:

Plan	Status	Description
1 DEFAULT_MAINTENANCE_PLAN	Default	Default plan for maintenance windows that prioritizes SYS_GROUP operations, leaving 5% for automated
2 DEFAULT_PLAN	Default	Default, basic, pre-defined plan that prioritizes SYS_GROUP operations and allocates minimal resource
3 DSS_PLAN	Example	Example plan for DSS workloads that prioritizes DSS queries over ETL.
4 ETL_CRITICAL_PLAN	Example	Example plan for DSS workloads that prioritizes ETL and critical DSS queries.
5 MIXED_WORKLOAD_PLAN	Example	Example plan for a mixed workload that prioritizes interactive operations over batch operations
6 OLTP_PLAN	ACTIVE	(null)

This information is also available under the **Settings** link under **Resource Manager**.

The screenshot shows the Oracle Database Control interface. The DBA navigation pane is open, showing various database management options. The main pane is titled "Welcome Page" and "admin_high". The "Settings" link under the Resource Manager section is selected, displaying the active resource plan:

Active Resource Plan: 1 OLTP_PLAN

Select **Resource Manager -> Plans-> OLTP_PLAN**

This will populate the main pane with an overview of the **OLTP_PLAN**

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : RESOURCEMANAGERPLAN ADMIN.OLTP_PLAN@admin_high". The left sidebar is titled "DBA" and contains nodes like "Connections", "Database Configuration", "Database Status", "Data Pump", "Performance", "RMAN Backup/Recovery", "Resource Manager", "Consumer Group Mappings", "Consumer Groups", and "Plans". The "Plans" node is expanded, showing sub-plans: "DEFAULT_MAINTENANCE_PLAN", "DEFAULT_PLAN", "DSS_PLAN", "ETL_CRITICAL_PLAN", "MIXED_WORKLOAD_PLAN", and "OLTP_PLAN". The "OLTP_PLAN" node is also highlighted with a red box. On the right, there's a "Details" tab showing a table with columns "Name" and "Value". The table rows are: 1 Plan (Value: OLTP_PLAN), 2 Status (Value: ACTIVE), 3 Is Subplan (Value: NO), 4 Management Method (Value: EMPHASIS), and 5 Description (Value: (null)).

Select the **Resource Allocation** tab to see how the resources are shared between the groups.

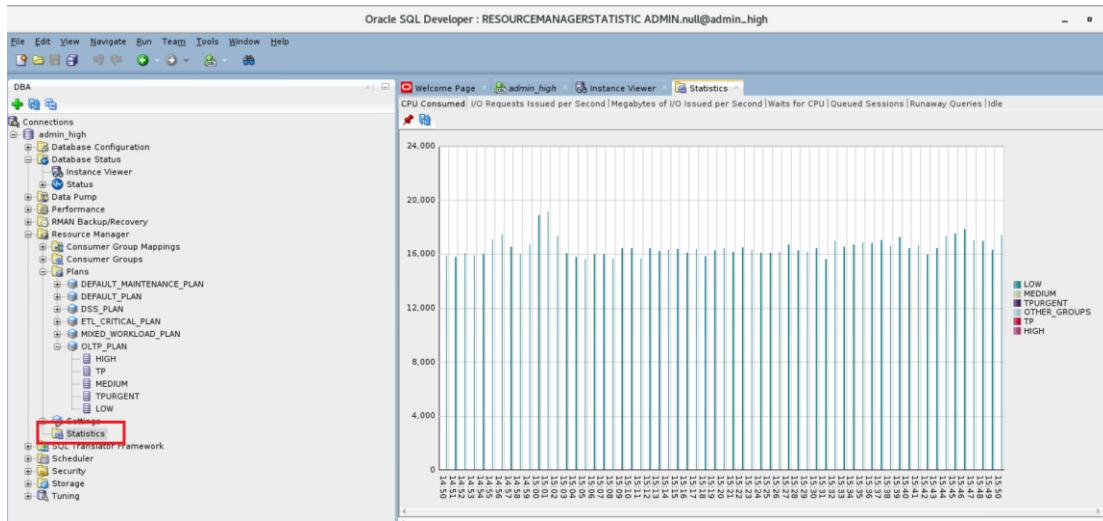
The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : RESOURCEMANAGERPLAN ADMIN.OLTP_PLAN@admin_high". The top navigation bar has tabs: "Welcome Page", "admin_high", "Instance Viewer", and "OLTP_PLAN". The "Resource Allocation" tab is selected and highlighted with a red box. Below the tab, there's a table with columns "Group/Subplan" and "Shares". The table rows are: 1 TPURGENT (Shares: 12), 2 HIGH (Shares: 4), 3 MEDIUM (Shares: 2), 4 LOW (Shares: 1), and 5 TP (Shares: 8).

Select the **Directive Values** tab to see the Maximum degree of parallelism associated with each group.

The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : RESOURCEMANAGERPLAN ADMIN.OLTP_PLAN@admin_high". The top navigation bar has tabs: "Welcome Page", "admin_high", "Instance Viewer", and "OLTP_PLAN". The "Directive Values" tab is selected and highlighted with a red box. Below the tab, there's a table with columns "Group" and "Max Degree of Parallelism". The table rows are: 1 TPURGENT (Max Degree of Parallelism: 1), 2 HIGH (Max Degree of Parallelism: 1), 3 MEDIUM (Max Degree of Parallelism: 4), 4 LOW (Max Degree of Parallelism: 1), and 5 TP (Max Degree of Parallelism: 1).

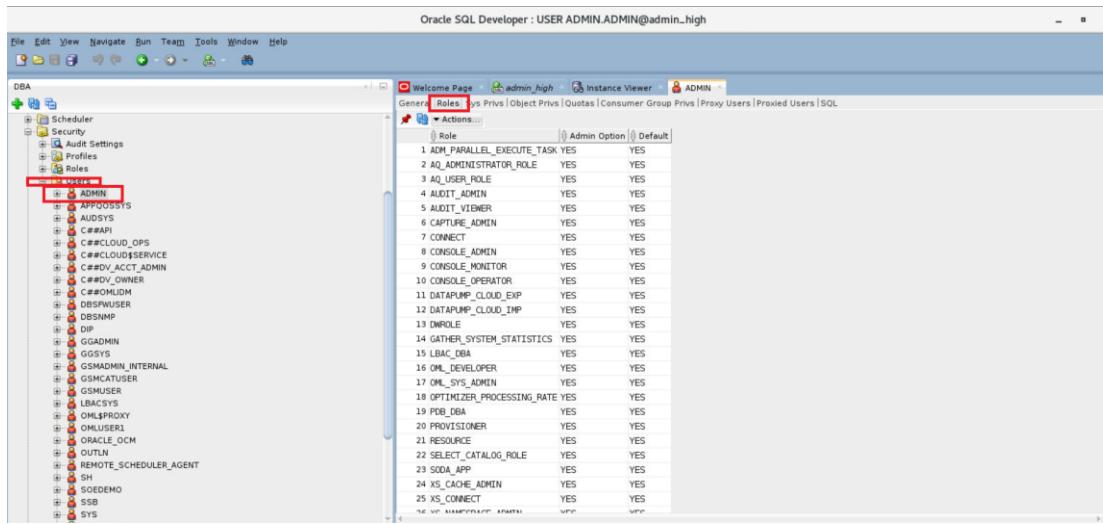
Resource Manager->Statistics provides graphical views into system usage by plan group. Double click on Statistics and select the tabs to explore the screens.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

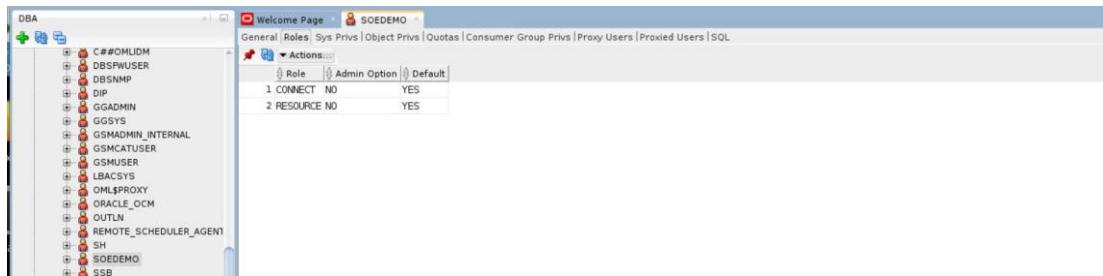


In the Security Section you can examine the users in the database and their roles and privileges.

Select **Security -> Users -> Admin** and list the roles associated with this user.



Compare this list with that of the SOEDEMO user if you created it in a previous lab.



You could also view the System privileges assigned to these users.

The **Storage** section can be used to display all the storage characteristics of the ATP database. Most of these characteristics cannot be changed, as the Autonomous Database automatically manages and optimizes storage for the service. You can still examine how the tablespace is configured.

Select Storage -> Tablespaces -> Data

This view may take a long time to refresh. Any data in a database will be stored in the DATA Tablespace. Below is a screenshot of the parameters of the DATA Tablespace . Examine the different entries in this area.

The screenshot shows the Oracle SQL Developer interface with the title bar "Oracle SQL Developer : TABLESPACE ADMIN.DATA@admin_high". The left sidebar is titled "DBA" and contains a tree view of database components. The "Storage" node under "Connections" is expanded, and its "Tablespaces" node is also expanded, with the "DATA" tablespace selected. A red box highlights the "DATA" node. The main pane displays a table titled "Actions..." showing various parameters for the DATA tablespace. Another red box highlights this table. The table has two columns: "Name" and "Value". The data is as follows:

Name	Value
1 TABLESPACE_NAME	DATA
2 BLOCK_SIZE	8192
3 INITIAL_EXTENT	65536
4 NEXT_EXTENT	(null)
5 MIN_EXTENTS	1
6 MAX_EXTENTS	2147483645
7 MAX_SIZE	2147483645
8 PCT_INCREASE	(null)
9 MIN_EXTLEN	65536
10 STATUS	ONLINE
11 CONTENTS	PERMANENT
12 LOGGING	LOGGING
13 FORCE_LOGGING	NO
14 EXTENT_MANAGEMENT	LOCAL
15 ALLOCATION_TYPE	SYSTEM
16 PLUGGED_IN	NO
17 SEGMENT_SPACE_MANAGEMENT	AUTO
18 DEF_TAB_COMPRESSION	DISABLED
19 RETENTION	NOT APPLY
20 BIGFILE	YES
21 PREDICATE_EVALUATION	STORAGE
22 ENCRYPTED	YES
23 COMPRESS_FOR	(null)
24 DEF_INMEMORY	DISABLED
25 DEF_INMEMORY_PRIORITY	(null)
26 DEF_INMEMORY_DISTRIBUTE	(null)

This concludes the SQL Developer Lab. SQL Developer is a very comprehensive powerful tool and you should continue to explore the different benefits it offers.



Lab 7- Exploring Auto-Scaling using Swingbench



Lab 7: (Optional) Exploring Auto-Scaling using Swingbench

Objectives:

- Learn how to enable the auto-scaling feature on your Autonomous Database
- Use a CPU intensive Swingbench load to trigger the scaling events.

Swingbench is a suite of utilities developed by Dominic Giles that creates Oracle Database objects and data, creating workloads that can then be monitored or used for testing in OLTP and Datawarehouse environments. It is written in Java and so can be run on many platforms. It is a free tool and provided without licenses and support. Among the many uses for Swingbench is the ability to provide a comprehensive demo of the database and its associated tools and functionality. At the end of this lab you can find links to much more information on Swingbench and its use.

The sample shell scripts for this lab are available in your VM under the directory `/home/oracle/labScripts/lab7`.

Prepare the workload

Generate the SHDEMO schema

You are going to use the Swingbench 'shwizard' utility to generate the SHDEMO schema for this lab. As the command line for this is long, there is a wrapper script `/home/oracle/labScripts/lab7/lab7_generate_schema.sh`.

The script requires 3 parameters, your connect string, the full path to your wallet zip file and the admin password for your database. Ensure that you are connecting to the **_high** service for your database.

```
cd /home/oracle/labScripts/lab7
./lab7_generate_schema.sh <connection string> <full path to wallet>
<admin password>
```

The schema generation should take under 15 minutes to complete. Debug output is enabled to give confidence that the generation is progressing.

You may observe some error messages similar to " ORA-01031: insufficient privileges". This is because the benchmark generation script tries to grant privileges that are not available in autonomous databases.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

```
Data Generated (MB) per sec          0.3
Actual Rows Generated           1,784,845
Commits Completed                304
Batch Updates Completed          9,133

2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.Wizard statusChanged() Validator Class = com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator <init>() New instance of OrderEntryValidator created
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() Validator object created, beginning validate()
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() Number of tables to validate = 8
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() Number of indexes to validate = 10
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() Number of views to validate = 0
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() Number of code to validate = 0
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.Wizard statusChanged() SalesHistoryValidator validate() Number of views to validate = 0
2:41:56 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.Wizard statusChanged() Connecting to : jdbc:oracle:thin:@MELATPTRAIN01_high
2:41:57 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.Wizard statusChanged() Connected
2:41:57 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator analyzeObjects() select table_name, status from user_tables where table_name
in ('COUNTRIES', 'CHANNELS', 'CUSTOMERS', 'PROMOTIONS', 'PRODUCTS', 'SUPPLEMENTARY_DEMOGRAPHICS', 'SALES', 'TIMES')
2:41:57 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator analyzeObjects() select index_name, status from user_indexes where index_name
in ('CUSTOMERS_MARITAL_BIX', 'CUSTOMERS_YOB_BIX', 'CUSTOMERS_GENDER_BIX', 'PRODUCTS_PROD_STATUS_BIX', 'PRODUCTS_PROD_SUBCAT_IX', 'PRODUCTS_PROD_CAT_IX', 'SALES_PROD_BIX',
'SALES_CUST_BIX', 'SALES_CHANNEL_BIX', 'SALES_PROMO_BIX')

Post Creation Validation Report
=====
The schema appears to have been created successfully.

Valid Objects
=====
Valid Tables : 'COUNTRIES', 'CHANNELS', 'CUSTOMERS', 'PROMOTIONS', 'PRODUCTS', 'SUPPLEMENTARY_DEMOGRAPHICS', 'SALES', 'TIMES'
Valid Views :
Valid Sequences :
Valid Code :
Schema Created
2:41:57 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.shdg.SalesHistoryValidator validate() OrderEntryValidator() completed validate()
2:41:57 PM FINE      [93] com.dom.benchmarking.swingbench.wizards.Wizard statusChanged() Exiting Interactive Session
[oracle@atpvm01 lab7]$
```

Validate the schema

Swingbench provides a utility script 'sbutil' to carry out many operations. As the command line for this is long, there is a wrapper script

/home/oracle/labScripts/lab7/lab7_sh_sbutil.sh.

The script requires 3 parameters, your connect string, the full path to your wallet zip file and a 'V' to indicate that you wish to validate the schema.

```
$ cd /home/oracle/labScripts/lab7
$ ./lab7_sh_sbutil.sh <your connect string> <full path to wallet zip
file> V
```

```
[oracle@atpvm01 lab7]$ ./lab7_sh_sbutil.sh MELATPTRAIN01_low /home/oracle/wallets/Wal
let_MELATPTRAIN01.zip V
  Running the sbutil command
sbutil -sh -cf /home/oracle/wallets/Wallet_MELATPTRAIN01.zip -cs MELATPTRAIN01_low -
u shdemo -p ATPwelcome1234 -val
The Sales History Schema appears to be valid.

-----|Object Type   |    Valid|  Invalid|  Missing|
-----|Table        |    8|     0|     0|
|Index        |   10|     0|     0|
|Sequence     |    0|     0|     0|
|View         |    0|     0|     0|
|Code         |    0|     0|     0|
```

Generate statistics

We will use the **/home/oracle/labScripts/lab7/lab7_sh_sbutil.sh** again, but with a different flag to call the sbutil script with appropriate parameters to generate statistics.

The script requires 3 parameters, your connect string, the full path to your wallet zip file and a 'S' to indicate that you wish to generate statistics

```
$ cd /home/oracle/labScripts/lab7
$ ./lab7_sh_sbutil.sh <your connect string> <full path to wallet zip
file> S
```

```
[oracle@atpvm01 lab7]$ ./lab7_sh_sbutil.sh MELATPTRAIN01_low /home/oracle/wallets/Wal
let_MELATPTRAIN01.zip S
Running the sbutil command
sbutil -sh -cf /home/oracle/wallets/Wallet_MELATPTRAIN01.zip -cs MELATPTRAIN01_low -
u shdemo -p ATPwelcome1234 -stats
Collecting statistics for the schema
Collected statistics in : 0:02:15.696
```

Run the workload with Auto Scaling Disabled

At the start of this exercise your Autonomous Database must have auto-scaling disabled and be configured to use 1 x OCPU. If not, use the Scale Up/Down button and adjust the CPU and disable auto-scaling.

[Autonomous Database](#) » Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2
OCID: ...ago6ma [Show](#) [Copy](#)
Created: Tue, 18 Jun 2019 07:32:10 GMT
CPU Core Count: 1 
Storage (TB): 1
License Type: Bring Your Own Licence
Database Version: 18c
Auto Scaling: Disabled 
Lifecycle State: Available

In your VNC connection to your Lab VM, open a terminal window and navigate to the \$HOME/labScripts/lab7 directory

Generate load on your database by running the charbench utility. Do this by running the wrapper script **/home/oracle/labScripts/lab7/lab7_scale_runsh.sh**.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The script requires 2 parameters, your connect string and the full path to your wallet zip file. Ensure that you are connecting to the `_low` service for your database.

```
$ cd /home/oracle/labScripts/lab7  
$ ./lab7_scale_runsh.sh <connect string> <full path to wallet>
```

```
[oracle@atpvm01 lab7]$ ./lab7_scale_runsh.sh MELATPTRAIN01_low /home/oracle/wallets/Wallet_MELATPTRAIN01.zip  
Calling charbench with the parameters charbench -c /home/oracle/labScripts/lab7/lab7_sh_config.xml -cf /home/oracle/wa  
llets/Wallet_MELATPTRAIN01.zip -cs MELATPTRAIN01_low -u shdemo -p ATPwelcome1234 -rt 0:10 -uc 32 -r /home/oracle/labScr  
ipts/lab7/scale142144.xml  
Author : Dominic Giles  
Version : 2.6.0.1082
```

Results will be written to /home/oracle/labScripts/lab7/scale142144.xml.
Hit Return to Terminate Run...

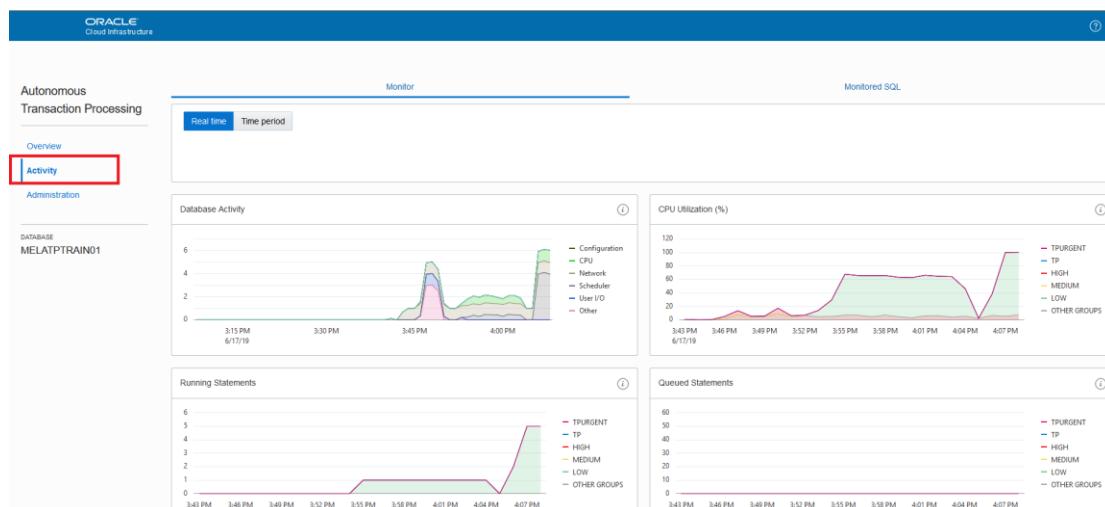
Time	Users	TPM	TPS
3:24:56 PM	32	939	13

We are overriding the following load related parameters for the benchmark.

- rt** specifies the total run time of the benchmark in hours and minutes. This has been set to 10 minutes to create a sustained high load on the system.
- uc** specifies the number of users who will connect to the system.
- r** specifies the output file for the summary results.

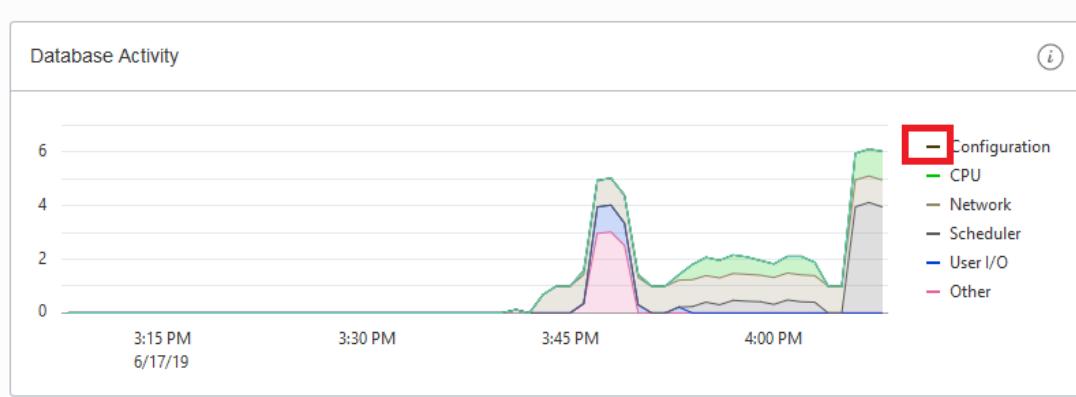
The benchmark will run for 10 minutes before completing. While this is running open the Activity page in the Autonomous Database Service Console to view real time information on your database.

The current activity is at the right hand side of the graphs. You may be able to see other recent activity in your database such as the generation of your schema. Do not worry if your graphs are not identical to the examples shown below.

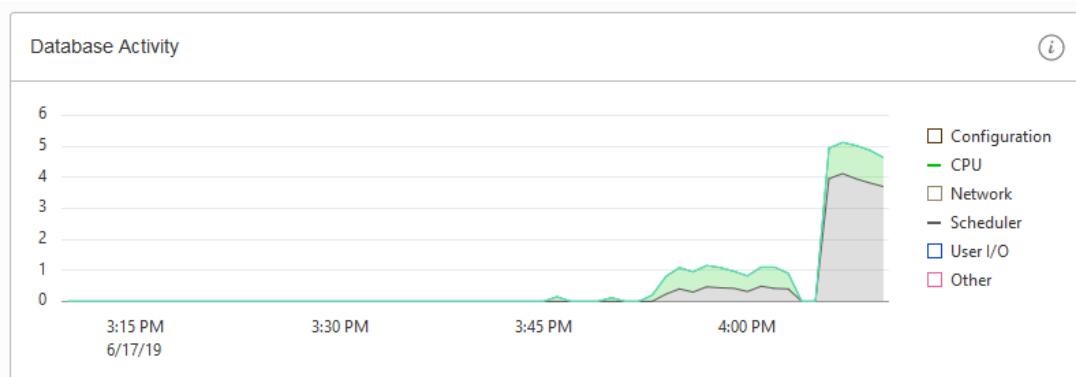


In the Activity page you can customize which items are displayed on the graphs by clicking the '-' next to a metric to remove it from display.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

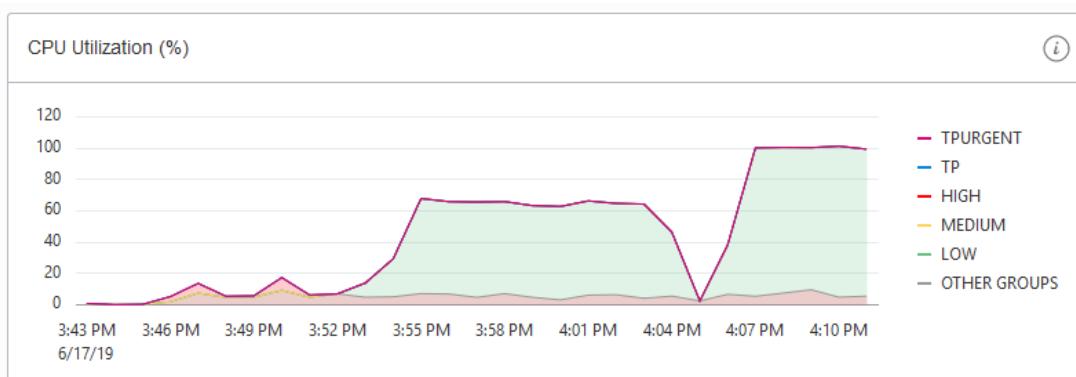


Using this method, de-select the other metrics until only CPU and Scheduler are displayed.



This example shows of the active sessions, 1 was running on the CPU, whilst the remaining sessions were in the 'Scheduler' wait class, which implies waiting to get onto a CPU to execute.

The CPU utilization graphic shows that the system was at almost 100% CPU utilization in my test run. Your values may be different.



Wait for the benchmark run to complete. When the script completes it will display the first 12 lines of the results file.

Results will be written to /home/oracle/labScripts/lab7/scale142144.xml.
Hit Return to Terminate Run...

Time	Users	TPM	TPS
2:31:50 PM	0	987	11

Completed Run.

```
#####
Your output was written to /home/oracle/labScripts/lab7/scale142144.xml
Showing the first 12 lines
#####
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Results xmlns="http://www.dominicgiles.com/swingbench">
  <Overview>
    <BenchmarkName>Sales History</BenchmarkName>
    <Comment>Configuration file for the Sales History Benchmark</Comment>
    <TimeOfRun>Jun 18, 2019 2:21:48 PM</TimeOfRun>
    <TotalRunTime>0:10:00</TotalRunTime>
    <TotalLogonTime>0:00:08</TotalLogonTime>
    <TotalCompletedTransactions>4455</TotalCompletedTransactions>
    <TotalFailedTransactions>0</TotalFailedTransactions>
    <AverageTransactionsPerSecond>7.42</AverageTransactionsPerSecond>
    <MaximumTransactionRate>480</MaximumTransactionRate>
  #####
  
```

Note the following summary results from your file

TotalCompletedTransactions	
AverageTransactionsPerSecond	
MaximumTransactionRate	

Enable Auto Scaling

In your web browser return to the Autonomous Database Details page for your Autonomous Database and select the "**Scale Up/Down**" button.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Autonomous Database » Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01

Workload Type: Transaction Processing

Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2

OCID: ...ago6ma [Show](#) [Copy](#)

Created: Tue, 18 Jun 2019 07:32:10 GMT

CPU Core Count: 1

Storage (TB): 1

License Type: Bring Your Own Licence

Database Version: 18c

Auto Scaling: Disabled [i](#)

Lifecycle State: Available

Tick the check box to enable auto-scaling and select "Update"

Scale Up/Down [help](#) [cancel](#)

CPU CORE COUNT STORAGE (TB)

The number of CPU cores to enable. Available cores are subject to your tenancy's service limits.

The amount of storage to allocate.

AUTO SCALING
Auto scaling allows the system to automatically scale the CPU core count up to three times the base core count to meet load demand.

Update

The lifecycle status for your database will change to "Scaling in progress". Like other scaling operations, this does not interrupt user sessions or require a database shutdown.

[Autonomous Database](#) » Autonomous Data



SCALING IN PROGRESS...

Wait until your lifecycle status changes to "Available" and auto scaling is enabled before proceeding.

[Autonomous Database](#) » Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2
OCID: ...ago6ma [Show](#) [Copy](#)
Created: Tue, 18 Jun 2019 07:32:10 GMT
CPU Core Count: 1
Storage (TB): 1
License Type: Bring Your Own Licence
Database Version: 18c
Auto Scaling: Enabled [*i*](#)
Lifecycle State: Available

Run the workload with Auto Scaling enabled

Generate the same load on your database by running the charbench utility, to generate load from the command line. Do this by running the same wrapper script **/home/oracle/labScripts/lab7/lab7_scale_runsh.sh**.

The script requires 2 parameters, your connect string and the full path to your wallet zip file. Ensure that you are connecting to the **_low** service for your database.

```
$ cd /home/oracle/labScripts/lab7
$ ./lab7_scale_runsh.sh <connect string> <full path to wallet>
```

```
[oracle@atpvm01 lab7]$ ./lab7_scale_runsh.sh MELATPTRAIN01_low /home/oracle/wallets/Wallet_MELATPTRAIN01.zip
Calling charbench with the parameters charbench -c /home/oracle/labScripts/lab7/lab7_sh_config.xml -cf /home/oracle/wallets/Wallet_MELATPTRAIN01.zip -cs MELATPTRAIN01_low -u shdemo -p ATPwelcome1234 -rt 0:10 -uc 32 -r /home/oracle/labScripts/lab7/scale140018.xml
Author : Dominic Giles
Version : 2.6.0.1082

Results will be written to /home/oracle/labScripts/lab7/scale140018.xml.
Hit Return to Terminate Run...

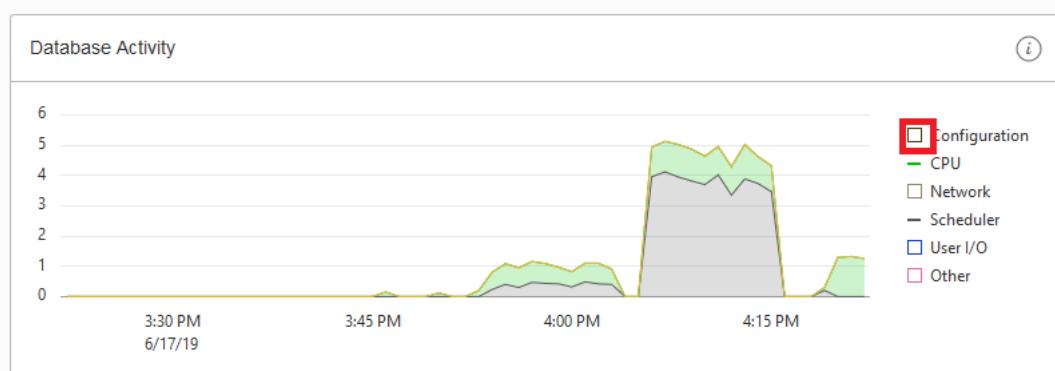
Time      Users    TPM     TPS
2:10:23 PM    0     1153     0
```

We are overriding the following load related parameters for the benchmark.

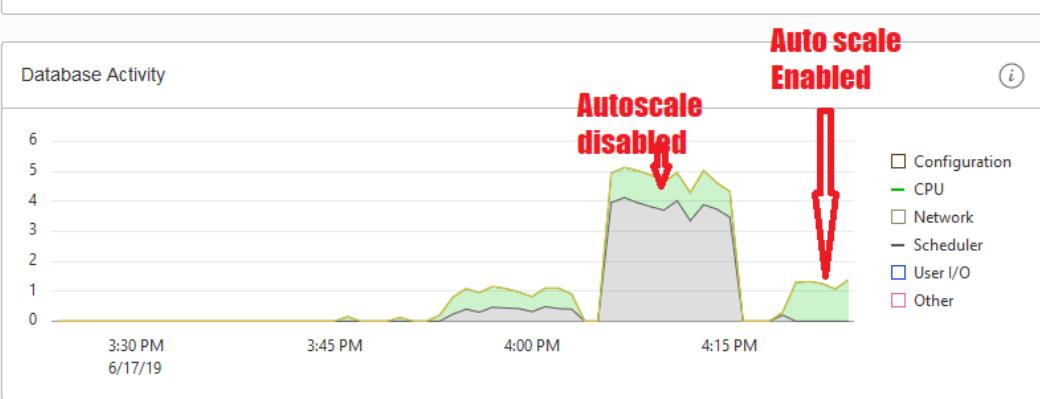
- rt** specifies the total run time of the benchmark in hours and minutes. This has been set to 10 minutes to create a sustained high load on the system.
- uc** specifies the number of users who will connect to the system.
- r** specifies the output file for the summary results.

The benchmark will run for 10 minutes before completing. Allow the benchmark to run for a few minutes and review the Activity Page in the Autonomous Database Service Console.

Customize the display of the "Database Activity" graph to only show the "CPU" and "Scheduler" classes of activity by clicking the '-' next to each of the other activity types.

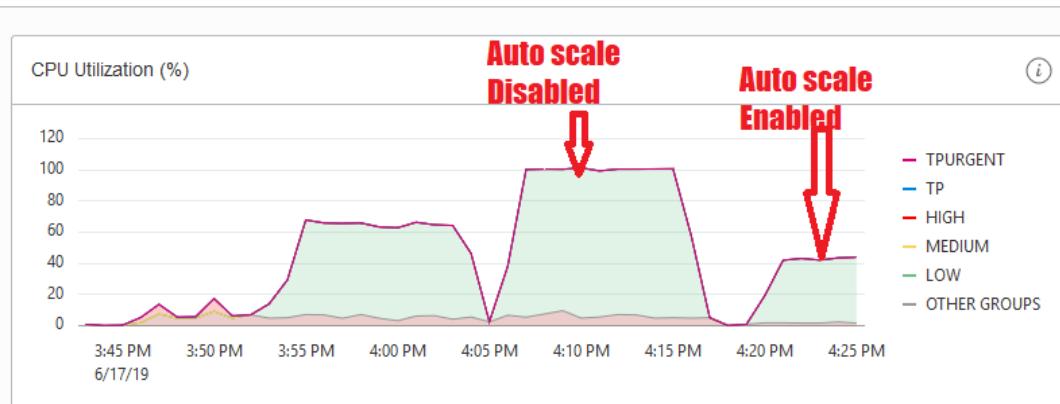


Now analyse the Database Activity chart.



After auto scaling was enabled, you can see that we have no sessions waiting in the "scheduler" activity, and more sessions using CPU.

The CPU Utilization chart should show that the overall % CPU Utilization is reduced.



Wait for the benchmark run to complete. When the script completes it will display the first 12 lines of the results file.

Completed Run.

```
#####
Your output was written to /home/oracle/labScripts/lab7/scale140018.xml
Showing the first 12 lines
#####
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Results xmlns="http://www.dominicgiles.com/swingbench">
  <Overview>
    <BenchmarkName>Sales History</BenchmarkName>
    <Comment>Configuration file for the Sales History Benchmark</Comment>
    <TimeOfRun>Jun 18, 2019 2:00:22 PM</TimeOfRun>
    <TotalRunTime>0:10:00</TotalRunTime>
    <TotalLogonTime>0:00:08</TotalLogonTime>
    <TotalCompletedTransactions>11619</TotalCompletedTransactions>
    <TotalFailedTransactions>0</TotalFailedTransactions>
    <AverageTransactionsPerSecond>19.36</AverageTransactionsPerSecond>
    <MaximumTransactionRate>1196</MaximumTransactionRate>
#####
oracle@atnvm01 lab7$
```

Note the following summary results from your file :

TotalCompletedTransactions	
AverageTransactionsPerSecond	
MaximumTransactionRate	

The results show an increase in the number of TotalCompletedTransactions, AverageTransactionsPerSecond and MaximumTransactionRate.

Your database has automatically, transparently and dynamically increased the number of OCPUs available during this test.

The Service Console Overview page only displays usage in 1 hour increments, so "The number of OCPUs allocated" graph may not reflect your auto scale operation for some time.

Disable Auto Scaling in your Database

In your web browser return to the Autonomous Database Details page for your Autonomous Database and select the "**Scale Up/Down**" button.

Autonomous Database » Autonomous Database Details

MELATPTRAIN01

ATP

AVAILABLE

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2
OCID: ...ago6ma [Show](#) [Copy](#)

Un-tick the check box to disable auto-scaling and select "Update"

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Scale Up/Down

[help](#) [cancel](#)

CPU CORE COUNT	STORAGE (TB)
1	1

The number of CPU cores to enable. Available cores are subject to your tenancy's service limits.

The amount of storage to allocate.

AUTO SCALING
Auto scaling allows the system to automatically scale the CPU core count up to three times the base core count to meet load demand.

Update

The lifecycle status for your database will change to "Scaling in progress". Like other scaling operations, this does not interrupt user sessions or require a database shutdown.



After a short wait your lifecycle status changes to "Available" and auto scaling is disabled.

Autonomous Database » Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console Scale Up/Down Stop Actions ▾

Autonomous Database Information Tags

General Information

Database Name: MELATPTRAIN01
Workload Type: Transaction Processing
Compartment: oractdemeabdmautodb (root)/ATP_Workshop/ATP_HOL2
OCID: ...ago6ma [Show](#) [Copy](#)
Created: Tue, 18 Jun 2019 07:32:10 GMT
CPU Core Count: 1
Storage (TB): 1
License Type: Bring Your Own Licence
Database Version: 18c
Auto Scaling: Disabled ⓘ
Lifecycle State: Available

Want to learn more about Swingbench?

Visit <http://www.dominicgiles.com/swingbench.html> to see more usage instructions.
See his blog at <http://www.dominicgiles.com/blog/blog.html> for more tips, tricks and innovations.

Section 3- Focus on Developers



Microservices on Kubernetes and Autonomous Database

Microservices on Kubernetes and Autonomous Database

This lab will walk you through the steps to set up a CI/CD environment for developing Microservices, based on the automation possible in Developer Cloud and deploying all components on Oracle's Managed Container platform and the ATP database.

The lab is hosted from GitHub, please use the below link to access the hands on guide for this lab. You will continue to use your VM and ATP created for the previous sections.

<https://github.com/oracle/cloudtestdrive/blob/Instructor-led/AppDev/ATP-OKE/README.md>

Want to Learn More?

Recommended Reading

- [Autonomous Transaction Processing Made Easy](#) (eBook)

Additional Resources

- Additional Resources: https://cloud.oracle.com/en_US/atp/additional-resources
- Demos and Videos: https://cloud.oracle.com/en_US/atp/videos
- [Oracle Autonomous Transaction Processing Documentation](#)
- Join the Autonomous Transaction Processing [forum](#).
- [Getting Started with Oracle Cloud](#)



Appendix

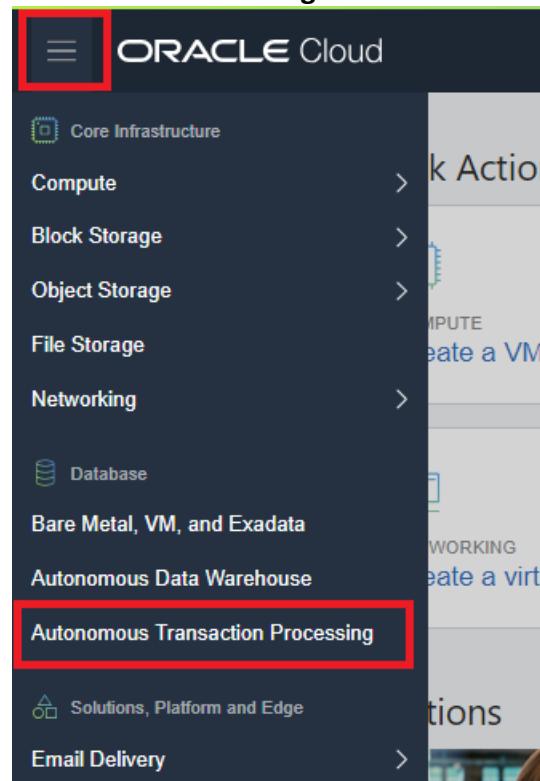
Appendix A – Provisioning Process Walkthrough

This is a walkthrough of the process used to create an ATP instance. This was demonstrated live during your lab session.

Creating your Autonomous Transaction Processing Database

Click on the **MENU** link at the top left of the page.

This will produce a drop-down menu, where you should select “**Autonomous Transaction Processing**”



This will take you to the management console page for ATP in the root compartment of the tenancy. The red warning icon "**Forbidden**" indicates that the Policies within the tenancy does not allow your user to create ATP Instances in the root compartment.

The screenshot shows the Oracle Autonomous Database management interface. On the left, there's a sidebar with 'List Scope' and 'Filters' sections. Under 'List Scope', the 'COMPARTMENT' dropdown is set to 'oractdemeabdmautodb (root)'. Below it is a search bar with placeholder text 'Don't see what you're looking for?'. Under 'Filters', there are dropdowns for 'STATE' (set to 'Any state') and 'WORKLOAD TYPE' (set to 'ATP'). At the bottom of the sidebar, there's a 'Tag Filters' section with a 'No tag filters applied' message. The main area is titled 'Autonomous Databases in oractdemeabdmautodb (root) Compartment'. It contains a table with columns: Name, State, Database Name, CPU Core Count, Storage (TB), Workload Type, and Created. The table is currently empty. A red exclamation mark icon is displayed above the table, and the word 'Forbidden' is written below it. At the bottom right of the main area, there are links for 'No Autonomous Databases' and 'Page 1'.

To begin the process of creating your ATP instance in this tenancy you need to select a compartment.

Click on the pulldown menu marked **Compartment**. Expand the menu under ATP_Workshop by clicking the ‘+’. Select the compartment that is given by the instructor. It may differ from ATP_Delegate shown below.

The screenshot shows the 'Autonomous Transaction Processing' interface. On the left, there's a 'List Scope' section. Under 'COMPARTMENT', the dropdown is set to 'oractdemeabdmautodb (root)', which is highlighted with a red box. Below it is a search bar with placeholder text 'Search compartments'. A list of compartments follows, with 'oractdemeabdmautodb (root)' at the top, followed by '+ ATP_Workshop' (with a red box around the '+' sign), 'ATP_Delegate' (which is also highlighted with a red box), and 'ManagedCompartmentForPaaS'. At the bottom of the dropdown, there's a 'Any state' dropdown. To the right of the dropdown, there's a vertical sidebar with a 'Create' button at the top, followed by a 'Name' input field and other empty fields.

Note – Your list of compartments may be different to the one shown above.

The main page will now change to show the list of ATP instances within your compartment, as shown below:

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Note: this lab uses the same tenancy and compartment for all lab attendees, therefore, it is possible that you may see ATP instances listed on this page which have already been created by other users attending this lab.

To create a new instance, click the blue "Create Autonomous Database" button.

Autonomous Databases in ATP_Delegate Compartment								
		Create Autonomous Database						
List Scope	Compartment	Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
ATP_Delegate	workteamdelegation (root) / ATP_Workshop / ATP_Delegate	dd11ATP	Available	dd11ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
		dd20ATP	Available	dd20ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:09:51 GMT
		dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
		dd22ATP	Available	dd22ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
		dd23ATP	Available	dd23ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
		dd24ATP	Available	dd24ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
		dd25ATP	Available	dd25ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
		dd26ATP	Available	dd26ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT
		dd27ATP	Available	dd27ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:41:50 GMT
		dd28ATP	Available	dd28ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:37:49 GMT
		dd29ATP	Available	dd29ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:33:48 GMT
		dd30ATP	Available	dd30ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:29:47 GMT

Enter the required information and click the "Create Autonomous Database" button at the bottom of the form. For the purposes of this lab, use the information below:

Workload Type: Autonomous Transaction Processing

Compartment: Verify that the correct compartment is selected. You will not be able to create instances outside this compartment due to Identity Management policies.

Display Name: Enter the display name for your ATP Instance. (Hint- use your username so you can identify your instance more easily)

Database Name: Enter any database name you choose that fits the requirements for ATP. The database name must consist of letters and numbers only, starting with a letter. The maximum length is 14 characters. (Hint- use your username so you can identify your instance more easily)

CPU Count: 1

Storage Capacity (TB): 1

Administrator Password: Enter any password you wish to use noting the specific requirements imposed by ATP. A suggested password for this lab is ATPwelcome-1234

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Create Autonomous Database help cancel

Workload Type

AUTONOMOUS DATA WAREHOUSE

AUTONOMOUS TRANSACTION PROCESSING
Configures the database for a transactional workload, with a bias towards high volumes of random data access.

Database Information

COMPARTMENT: ATP_Delegate

DISPLAY NAME: MELATPTRAIN01

DATABASE NAME: MELATPTRAIN01
The name must contain only letters and numbers, starting with a letter. 14 characters max.

CPU CORE COUNT: 1 STORAGE (TB): 1
The number of CPU cores to enable. Available cores are subject to your tenancy's service limits.
The amount of storage to allocate.

Administrator Credentials

Set the password for your Autonomous Transaction Processing database ADMIN user here.

USERNAME: READ-ONLY
ADMIN

PASSWORD:

CONFIRM PASSWORD:

License Type

MY ORGANIZATION ALREADY OWNS ORACLE DATABASE SOFTWARE LICENSES
Bring my existing database software licenses to the database cloud service ([details](#)).
 SUBSCRIBE TO NEW DATABASE SOFTWARE LICENSES AND THE DATABASE CLOUD SERVICE

TAGS

Tagging is a metadata system that allows you to organize and track resources within your tenancy. Tags are composed of keys and values that can be attached to resources.

[Learn more about tagging](#)

TAG NAMESPACE	TAG KEY	VALUE
None (apply a free-form tag)	<input type="text"/>	<input type="text"/>

+ Additional Tag

Create Autonomous Database

When you enter the administrator password, note the specific requirements imposed by ATP:

Administrator Credentials

Set the password for your Autonomous Transaction Processing database ADMIN user here.

USERNAME: READ-ONLY
ADMIN

PASSWORD:
•

>Password must be 12 to 30 characters and contain at least one uppercase letter, one lowercase letter, and one number. The password cannot contain the double quote ("") character or the username "admin".

CONFIRM PASSWORD:

Oracle Autonomous Transaction Processing Hands-on Lab Guide

When you have completed the required fields, scroll down and click on the blue **Create Autonomous Database** button at the bottom of the form:

The screenshot shows the 'Create Autonomous Database' form. It includes sections for 'TAG NAMESPACE' (None (apply a free-form tag)), 'TAG KEY' (empty), 'TAG VALUE' (empty), and a '+ Additional Tag' button. At the bottom is a large blue 'Create Autonomous Database' button.

The console page will display the message “**Provisioning**” under the “State” column.

Autonomous Databases in ATP_Delegate Compartment						
List Scope		Name	State	Database Name	CPU Core Count	Storage (TB)
compartments	MELATPTRAIN01		Provisioning	MELATPTRAIN01	1	1
ATP_Delegate	DELEGATE		Available	ds2CATP	1	1
compartments (except ATP_Delegate)	DELEGATE		Available	ds2CATP	1	1
Designs	DELEGATE		Available	ds2CATP	1	1
Don't see what you're looking for?						
Filters		name	Any State			
Workload Type		All				
Tag Filters		Any tag filters applied				

Click on the name of your ATP instance, as shown below

Autonomous Databases in ATP_Delegate Compartment						
List Scope		Name	State	Database Name	CPU Core Count	Storage (TB)
compartments	MELATPTRAIN01		Provisioning	MELATPTRAIN01	1	1
ATP_Delegate	DELEGATE		Available	ds2CATP	1	1
compartments (except ATP_Delegate)	DELEGATE		Available	ds2CATP	1	1
Designs	DELEGATE		Available	ds2CATP	1	1
Don't see what you're looking for?						
Filters		name	Any State			
Workload Type		All				
Tag Filters		Any tag filters applied				

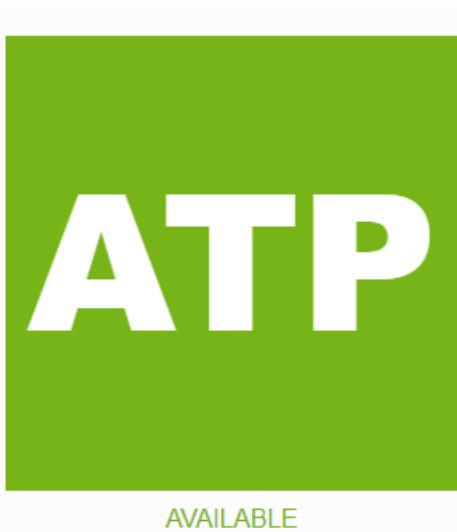
This will display more information about your instance and you should notice the various menu buttons that help you manage your new instance – because the instance is currently being provisioned all the management buttons are grayed out.

Autonomous Database - Autonomous Database Details											
MELATPTRAIN01	ATP										
DB Connect	Service Console										
Scale Up/Down	Start										
Actions ▾											
Autonomous Database Information											
Workload Type: Transaction Processing	Created: Fri, 08 Mar 2019 08:38:38 GMT										
Display Name: MELATPTRAIN01	Compressed: No										
Database Name: MELATPTRAIN01	OCDB: No (79% Used)										
CPU Core Count: 1	License Type: Bring Your Own License										
Storage (TB): 1	Lifecycle Status: Provisioning										
Resources											
Backups											
Backups are automatically created daily.	Create Manual Backup										
	<table border="1"><thead><tr><th>Name</th><th>State</th><th>Type</th><th>Started</th><th>Ended</th></tr></thead><tbody><tr><td colspan="5">No items found.</td></tr></tbody></table>	Name	State	Type	Started	Ended	No items found.				
Name	State	Type	Started	Ended							
No items found.											
	Showing 0 item(s) < Page 1										

A summary of your instance status is shown in the large box on the left. In this example, the color is amber and the status is **Provisioning**



After a short while the status will change to **Available** and the “ATP” box will change color to green:



Once the Lifecycle Status is **Available**, additional summary information about your instance is populated, including workload type. You can also see the Lifecycle Status reported in this region.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

MELATPTRAIN01

Autonomous Database Information Tags

Workload Type: Transaction Processing
Display Name: MELATPTRAIN01
Database Name: MELATPTRAIN01
CPU Core Count: 1
Storage (TB): 1
Lifecycle State: Available

Resources Backups

Backups are automatically created daily.

Create Manual Backup

Name	State	Type	Started	Ended
No items found.				

Showing 0 item(s) < Page 1 >

Congratulations you have created your first ATP instance!

Return to the main page which list all your ATP instances by clicking on the Autonomous Database link at the top of the page:

Autonomous Database » Autonomous Database Details

MELATPTRAIN01

DB Connection Service Console Scale Up/Down

Autonomous Database Information Tags

Workload Type: Transaction Processing
Display Name: MELATPTRAIN01
Database Name: MELATPTRAIN01
CPU Core Count: 1
Storage (TB): 1

Autonomous Database

Autonomous Databases in ATP_Delegate Compartment

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Created
MELATPTRAIN01	Available	MELATPTRAIN01	1	1	Transaction Processing	Fri, 08 Mar 2019 08:38:38 GMT
dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:13:58 GMT
dd24ATP	Available	dd24ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:09:57 GMT
dd23ATP	Available	dd23ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:05:56 GMT
dd22ATP	Available	dd22ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 19:01:55 GMT
dd21ATP	Available	dd21ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:57:54 GMT
dd20ATP	Available	dd20ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:53:53 GMT
dd19ATP	Available	dd19ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:49:52 GMT
dd18ATP	Available	dd18ATP	1	1	Transaction Processing	Thu, 07 Mar 2019 18:45:51 GMT

You are now ready to start Lab 2.

Appendix B – Creating and preparing a user to access Object Storage

Information about how the environment was prepared

To load data from the Oracle Cloud Infrastructure Object Storage you will need a Cloud user with the appropriate privileges to read data from the Object Store. The communication between the database and the object store relies on the Swift protocol and a username/password authentication token.

This is an outline of the steps that your instructor carried out – your lab user does not have permission to do these operations, or to see all of the referenced objects.

Select/Create your object storage bucket.

The screenshot shows a list of buckets in the ATP_Workshop Compartment. There is one bucket named "DEMO_DATA". The bucket has a blue circular icon with a white letter "B" in the center. To the right of the bucket name, the text "Created: Fri, 21 Dec 2018 11" is displayed. At the top of the page, there is a "Create Bucket" button.

Upload the files to the object storage bucket.

The screenshot shows the details for the "DEMO_DATA" bucket. At the top, there are buttons for "Change Compartment", "Update Visibility", "Delete", and "Apply Tag(s)". Below these, there are tabs for "Bucket Information" and "Tags", with "Bucket Information" being the active tab. Under "Bucket Information", the namespace is listed as "oscemea001", the storage tier is "Standard", the ETag is "984a1594-6465-4243-8c8a-3e4dde44353c", and the encryption key is "None". A note at the bottom says "Developer tools are available for advanced object operations.".

The screenshot shows the "Objects" page for the "DEMO_DATA" bucket. At the top, there are buttons for "Upload Object" and "Restore Object". Below these, a list of objects is shown: "ATP-HOL-Long-v0.6.pdf", "channels.csv", "channels_special.csv", and "countries.csv".

Construct the URL that points to the location of the file staged in the OCI Object Storage. The URL is structured as follows

https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<tenant_name>/<bucket_name>/<file_name>

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Create a user, in this example, 'atp_oss_access' and associate the user with a group.

atp_oss_access

Description: ATP Workshop shared OSS access account

Create/Reset Password Edit User Capabilities Unblock Delete Apply Tag(s)

User Information Tags

OCID: aps47a Show Copy

Created: Fri, 21 Dec 2018 12:47:52 GMT

Capabilities

Local password: Yes
API keys: Yes
Auth tokens: Yes

Groups

Add User to Group

G ATP_Group OCID: aps47a Show Copy

Create a policy statement to allow the group to read the object storage bucket.

Allow group Atp_group to read buckets in compartment Atp_workshop

Create an Authentication Token (Auth Token) for this user. Make a note of it as you are only told this information once.

atp_oss_access

Description: ATP Workshop shared OSS access account

Create/Reset Password Edit User Capabilities Unblock Delete Apply Tag(s)

User Information Tags

OCID: aps47a Show Copy Created: Fri, 21 Dec 2018 12:47:52 GMT Status: Active Federated: No

Capabilities

Local password: Yes API keys: Yes Auth tokens: Yes SMTP credentials: Yes Customer secret keys: Yes

Auth Tokens

Generate Token

AT OCID: xxyyyq Show Copy Description: Token created for ATP workshops

To access data in the Object Store you must enable your database user to authenticate itself with the Object Store using your object store account and Auth Token.

You do this by creating a private CREDENTIAL object for your user that stores this information encrypted in your ATP instance using the DBMS_CLOUD package. This encrypted connection information is only usable by your user schema.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

```
set define off
begin
  DBMS_CLOUD.create_credential(
    credential_name => 'OBJ_STORE_CRED',
    username => 'atp_oss_access',
    password => 's<i[GK}R1:SD+uRA42MG'
  );
end;
/
```

Appendix C – Connecting to the database using Oracle Machine Learning Notebook

You can use the included Oracle Machine Learning OML Notebook based environment to connect to your ATP environment. OML is a browser-based environment, and so provides an easy to connect and fast environment to work with ATP. It can be used to run SQL queries and scripts, which can then be grouped together within a notebook. Notebooks can be used to build single reports, collections of reports and dashboards, and notebooks can be shared with other users.

Creating an OML user

The OML Notebook requires you to create a user, as the ATP instance admin user is prohibited from creating notebooks and jobs.

Return to your Firefox window and open your ATP instance service console.

Select Administration and Manage Oracle ML Users

The screenshot shows the Oracle Cloud Infrastructure (OCI) Autonomous Transaction Processing (ATP) service console. The top navigation bar has 'ORACLE Cloud Infrastructure' and a search icon. Below it, the main menu has 'Autonomous Transaction Processing' selected, with 'Overview' and 'Activity' also visible. A red box highlights the 'Administration' tab, which is currently active. On the left, under 'DATABASE', it shows 'MELATPTTRAIN01'. The central content area has several cards: 'Download Client Credentials (Wallet)', 'Set Resource Management Rules', 'Set Administrator Password', 'Download Oracle Instant Client', and 'Send Feedback to Oracle'. The 'Manage Oracle ML Users' card is highlighted with a red box. This card describes how to create new Oracle Machine Learning user accounts and manage existing ones. It includes a link to 'Create' a new user.

You will now see the User management page. Select **Create** to create your new user.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Machine Learning User Administration interface. At the top, it says 'ORACLE Machine Learning User Administration'. Below that is a header with 'Users' and buttons for '+ Create', 'Delete', and 'Show All Users'. There is also a search bar and a status indicator. The main area is a table with columns: User Name, Full Name, Role, Email, Created On, and Status. One row is shown: 'ADMIN' with 'System Administrator' role, 'melanie.ashworth-march@oracle.com' email, '11/25/18 4:42 PM' created on, and 'Open' status.

Enter the required information and click **Create**. For the purposes of this lab, use the information below:

Username: *omluser1*

Email Address: Enter a valid email address

Uncheck the generate password button. You can optionally have a secure password emailed to the email address and then reset it on first login. As access to your email is not guaranteed, to speed up this lab we are going to manually enter a password.

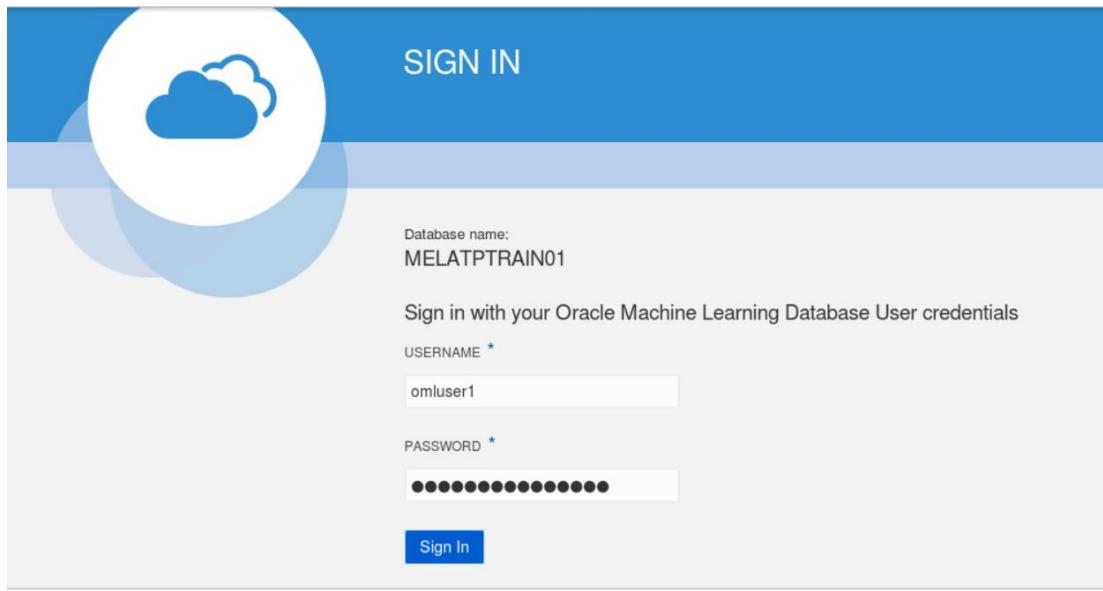
Password: Enter any password you wish to use noting the specific requirements imposed by ATP. A suggested password for this lab is ATPwelcome-1234

The screenshot shows the 'Create User' form. It has fields for Username ('omluser1'), First Name, Last Name, and Email Address ('melanie.ashworth-march@oracle.com'). There is a checkbox for generating a password and account details, which is unchecked. Below that are fields for Password and Confirm Password, both containing 'ATPwelcome-1234'. At the bottom right are 'Create' and 'Cancel' buttons, with 'Create' highlighted by a red box.

Connect as your new OML user by selecting the Home Icon (shaped like a house) on the Users screen

The screenshot shows the 'Users' screen again. A message at the top says 'User Created'. The table now includes a new row for 'OMLUSER1' with 'Developer' role, 'melanie.ashworth-march@oracle.com' email, '1/3/19 4:37 PM' created on, and 'Open' status. In the top right corner, there is a small navigation bar with icons, one of which is highlighted by a red box.

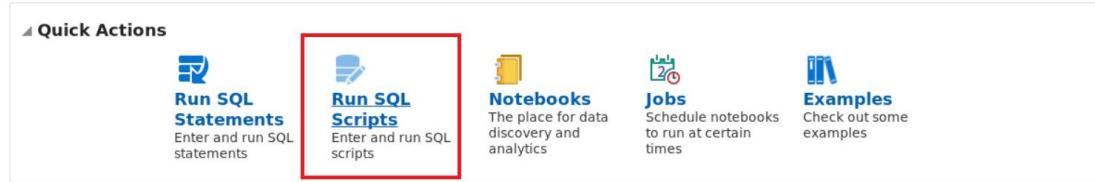
This will open a new browser tab. Sign in as **omluser1**



You are now connected as an OML Notebook user.

Now we will use the OML SQL Scripts function to run the same simple query we ran in SQL Developer.

Select **Run SQL Scripts** to open the SQL Script Scratchpad.



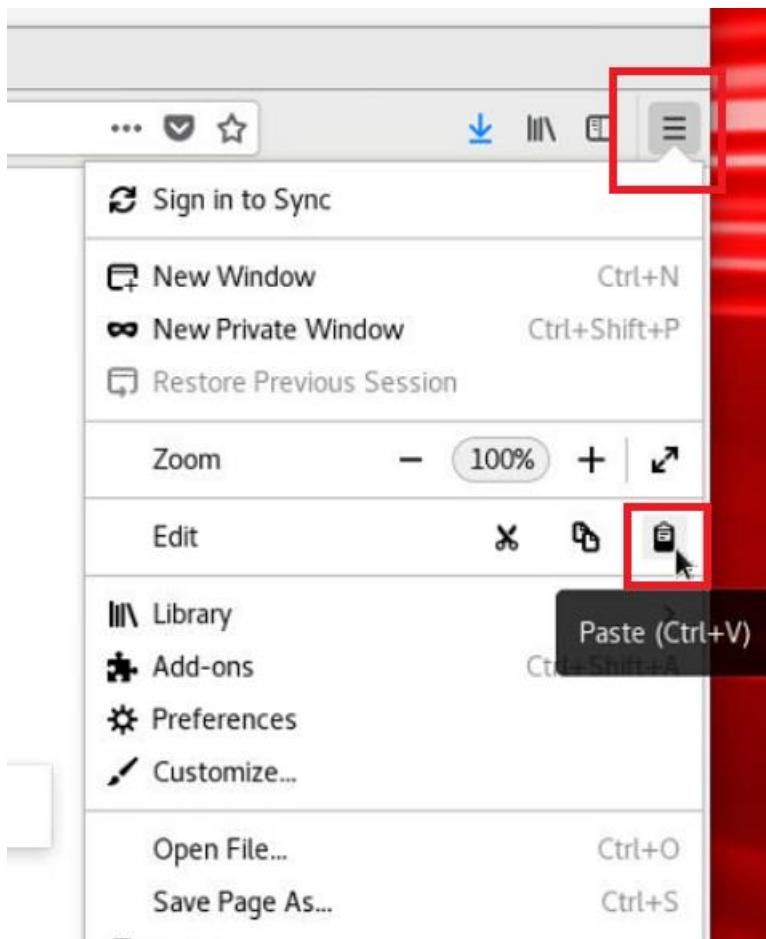
Note: As this is the first time the Notebook Server has been started in this ATP instance, sometimes it will fail to start first time with the message 'Notebook server not available, please try again at a later time' Just try the Run SQL Scripts button again and it should start successfully.

Notebook server not available, please try again at a later time

Copy this SQL statement and paste it in the gray SQL script box.

```
select country_region, count(country_name) from sh.countries group by country_region;
```

If you are having a problem pasting data into Firefox there is a 'Paste' menu item in the Firefox Menu.



Select the **Run all Paragraphs** icon to execute the script.



Select **ok** on the **Run all paragraphs** confirmation screen. The results of the query will be displayed.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Machine Learning interface. At the top, it says "ORACLE Machine Learning" and "OMLUSER1 Project [OMLUSER1 Work...]" with a "Connected" status. Below that is the "SQL Script Scratchpad" tab. In the main area, there is a code editor with the following SQL script:

```
%script
select country_region, count(country_name) from sh.countries group by country_region;
```

Below the code, the output is displayed:

COUNTRY_REGION	COUNT(COUNTRY_NAME)
Asia	5
Middle East	1
Europe	10
Americas	4
Oceania	2
Africa	1

Text at the bottom of the output area: "6 rows selected." and "Total 4 sec. Last updated by OMLUSER1 at January 04 2019, 1:56:07 PM."

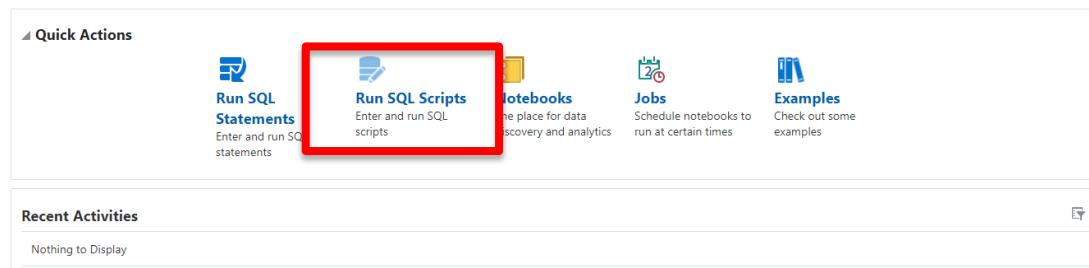
You have successfully connected and run an operation against ATP with Oracle OML.

Selecting the Service in OML

OML makes it easier to do this because Notebooks automatically “see” all services available in the instance. Notebooks also allow re-running the same query multiple times without having to make any changes or re-posting the query.

Your OML session should still be open from the previous section, if not connect to OML as your omluser1 session.

If not already in **Run SQL Scripts**, select it from Quick Actions:



Before running a query, explore the different services that can be used to run the query by selecting the **Interpreter binding** button (next to the **default** drop down button) which is shown below:

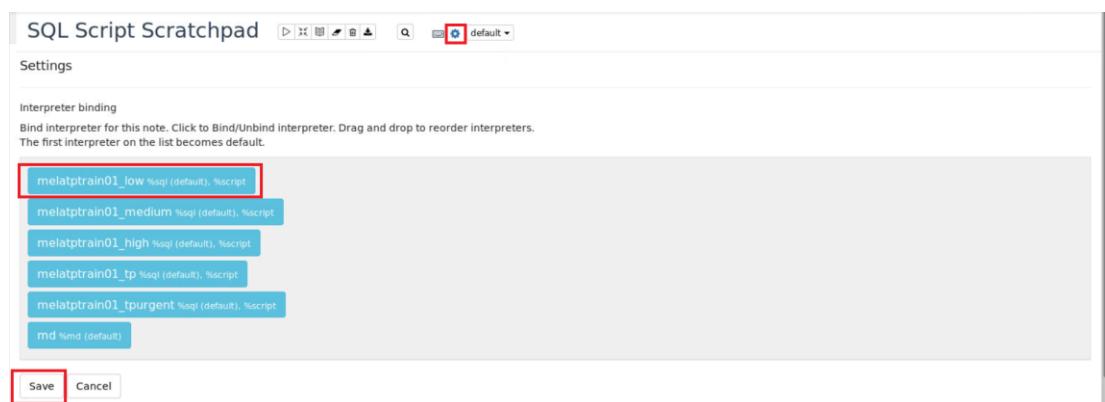


A new pane opens listing all the services available to run queries. You will see the five services already discussed previously and a new one specific to OML (md). The services are selected in list order, with the service at the top of the list being tried first. OML will try to connect to each selected service in order, and will then execute the SQL against the first successfully connected service.

The first time you run a query against a service, OML will establish a database connection. This will be reflected in the total execution time, which could include several additional seconds for the new connection. With a short query this will substantially add to the total execution time.

A blue color indicates the service is selected and will be tried, if you click on any of the services it will turn white, which de-selects it. You can drag and drop any service up or down the list to specify the order in which they will be attempted.

For the first test we will use the **_LOW** service. If it is not at the top of the list, drag and drop it to the top of the list. Select **Save** when done.



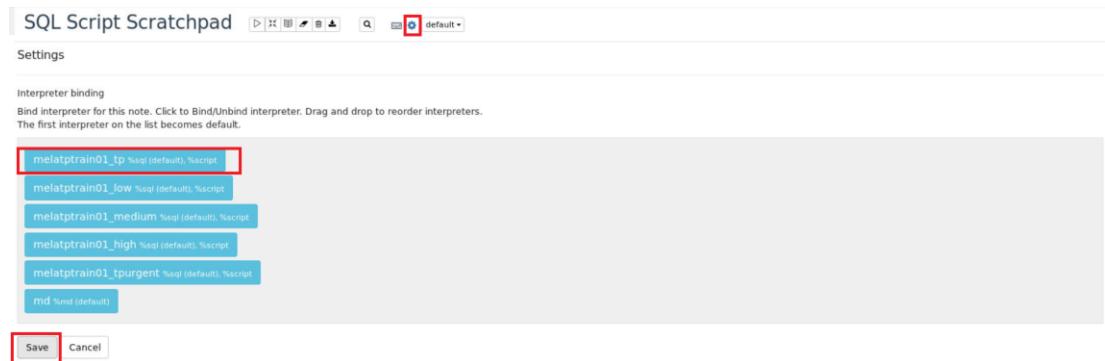
Paste this query into OML. The sql can be found in the file
/home/oracle/labScripts/lab2/lab2_oml_countries.sql

```
Select /*+ no_result_cache */ count(*), co.country_name,
co.country_region
From sh.countries co, sh.customers cu
where co.country_id=cu.country_id
group by co.country_region, co.country_name
order by co.country_region;
```

Now run the query by selecting the **Run This Paragraph** button



Now run the query in the **_tp** service. Follow the steps above for changing the service by selecting the **Interpreter binding** button and when the list of services appears drag the **_tp** service to the top and **Save**:



Now run the exact same query as before by selecting the **Run This Paragraph** button. As this is the first connection to the **_tp** service the elapsed run time for this SQL will be slower, and it will include the time taken to make the initial connection.



The number of CPUs assigned to your instance, the other workloads running and the amount of parallelization that your query requires will affect how much impact selecting the different built in services has on your query run time.

Appendix D – Configuring Oracle Cloud Infrastructure for REST

In lab 3 we demonstrate how to use REST and the Oracle provided oci-curl script to control an Autonomous Transaction Processing cloud. To make this lab faster, the environment was pre-configured. This appendix will detail the steps taken to get the environment ready for REST in preparation for Lab 3 in this guide.

For full information on the REST APIs please see the Oracle Infrastructure Cloud Documentation:

<https://docs.cloud.oracle.com/iaas/Content/API/Concepts/usingapi.htm>

Create the keys

To allow REST scripts to be able to communicate with the Oracle cloud, we need to authenticate them to ensure the security of the cloud environment. A REST script uses an already created Oracle cloud user. The REST script can only perform actions that the selected cloud user can perform, these permissions are part of the policies which apply to that user. If a REST script tries to perform an action that the cloud user is not authorized to perform the REST script will fail.

To ensure authentication between the REST script and the cloud user we create an API signing key that the REST script will use to authenticate itself when running. The API signing key uses a RSA key pair in PEM format (minimum 2048 bits).

First create the private key (we are generating it in `~/.oci`):

```
$ openssl genrsa -out ~/.oci/oci_api_key.pem 2048
```

- We can then create the public key from it:

```
$ openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out  
~/.oci/oci_api_key_public.pem
```

Uploading the Key to Oracle Cloud Infrastructure

Once the key is generated we need to upload the public key to the Oracle cloud user that we will use. In this example we will use the user `atpworkshop01`.

Oracle Autonomous Transaction Processing Hands-on Lab Guide

Autonomous Database

List Scope

COMPARTMENT

ATP_Delegate

oscemea001 (root)/ATP_Workshop/ATP_Delegate

Don't see what you're looking for? (i)

Filters

STATE

Any state

Create Autonomous Database

Name	State	Database Name	CPU Core Count	Storage (TB)	Workload Type	Last Modified
MELATPTRAIN1	Available	MELATPTRAIN1	1	1	Transaction Processing	Wed, 29 May 2019 09:04:35 GMT
RESTADB1	Available	RESTADB1	1	1	Transaction Processing	Fri, 24 May 2019 19:42:33 GMT

Displaying 2 Autonomous Databases < Page 1 >

Profile atpworkshop01
Tenancy: oscemea001
Change Password
User Settings
Sign Out

Terms of Use and Privacy | Cookie Preferences Copyright © 2019, Oracle and/or its affiliates. All rights reserved.

Select the API keys from the left-hand menu, and then “Add Public Key”

Identity > Users > User Details

ATPworkshop01

Description: ATP Workshop user account - ATPworkshop01

Create/Reset Password | Enable Multi-Factor Authentication | Edit User Capabilities | Unblock | Delete | Apply Tag(s)

User Information | Tags

OCID: ...lomita Show Copy

Created: Tue, 28 May 2019 17:14:51 GMT

Multi-factor authentication: Disabled

Email:

Capabilities

Local password: Yes | SMTP credentials: Yes

API keys: Yes | Customer secret keys: Yes

Auth tokens: Yes

Resources

API Keys (0) | Auth Tokens (1) | SMTP Credentials (0) | Customer Secret Keys (0) | Groups

Add Public Key

There are no API Keys for this User.

Add Public Key

Add Public Key

Note: Public Keys must be in the PEM format.

PUBLIC KEY

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAQ8AMTIBBgkqhkiG9w0BAQEwDQYJKoZIhvcNAQEBBQADggEwC0RhP+8L15Fd9B
U2JzqfXWvHgRzKdVnOjPj7p0sL7z17wpqB7j7Q1L5p0jDwDy101200X
C1+Uit+vS08cYcrv51eng7KdUEUhBeRUDjhHulg77Okd4EwNm0THSe8TzLYLvv
9He1CF7zG2FP/(QmNv49p01Xgd1Tc+pZtLuk+9KNTYXwGLRSFcep/U2LJHna1Toj
9BXmeRoDZE+DULQ3jyxjh4m4jp0hgogkxYLh1fdVCCBxEuFh2u2nDQVHYloqKm
1NG6+Ly9chahGc4oxom9MFTovQ2NW1YZXFnBy32PTGB1/oPjoTkbk0iL2nCFE715
+IDAQAB
-----END PUBLIC KEY-----
```

help cancel

User Info

OCID: ...lomita

Create/Reset Password | Unblock | Delete

Multi-Factor Authentication

Email:

Capabilities

In the pop-up window, copy and paste the contents of the public key file you generated. Press “Add” in the window when completed.

The key is added to the user and the fingerprint is displayed in the console. Each user can have up to three API keys.

The screenshot shows the Oracle Cloud Infrastructure (OCI) console interface. On the left, there's a sidebar with 'Resources' and a list of items: 'API Keys (1)', 'Auth Tokens (1)', 'SMTP Credentials (0)', 'Customer Secret Keys (0)', and 'Groups'. The main area is titled 'API Keys' and shows a single entry. There's a blue button labeled 'Add Public Key'. Below it, a green circle contains the letters 'PK'. To the right of the circle is a red-bordered box containing the text 'Fingerprint: 61:85:ae:a6:03:3b:8e:da:27:76:6e:24:03:1fa5:f9'. Further to the right is the text 'Time Created: Thu, 30 May 2019 13:06:31 GMT'. At the top right of the main area, it says 'Displaying 1 API Keys' and there are three dots (...).

Download and customize the script

This lab uses the oci-curl script provided in the Oracle documentation.

<https://docs.cloud.oracle.com/iaas/Content/API/Concepts/signingrequests.htm>

Examples are also given in:

- PowerShell
- C#
- Java
- NodeJS
- Perl
- PHP
- Python
- Ruby
- Go

These scripts make it easier to start working with REST. Before you can use the downloaded script, you must edit it to add information about your environment.

At the time of writing the oci-curl script is on version 1.0.2. Towards the top of the file you will see a comment #TODO

```
# TODO: update these values to your own
Local tenancyId=<Your tenancy OCID>;
Local authUserId=<Your user OCID> ;
local keyFingerprint=<API Key fingerprint>;
local privateKeyPath=<location of private key file>;
```

You need to populate these four variables with details from your environment.

To populate the 'tenancyId' you need to locate your Tenancy OCID. This can be found by selecting the region menu and selecting "**Manage Regions**"

Oracle Autonomous Transaction Processing Hands-on Lab Guide

The screenshot shows the Oracle Cloud Identity interface. In the top right corner, there is a search bar with 'eu-frankfurt-1' and a dropdown arrow, which is highlighted with a red box. Below the search bar is a 'Regions' section with a list of regions: eu-frankfurt-1, us-ashburn-1, ap-seoul-1, uk-london-1, ap-tokyo-1, us-phoenix-1, and ca-toronto-1. A 'Manage Regions' button is also present, highlighted with a red box. The main content area displays a user profile for 'ATPworkshop01'. The user has a green circular icon with a white 'A' and is marked as 'ACTIVE'. The 'User Information' tab is selected. The 'Description' field contains 'ATP Workshop user account - ATPworkshop01'. Below the description are several buttons: 'Create/Reset Password', 'Enable Multi-Factor Authentication', 'Edit User Capabilities', 'Unblock', 'Delete' (highlighted with a red box), and 'Apply Tag(s)'. The 'Tags' tab is also visible. At the bottom, the 'OCID' is listed as '...lomtq' with options to 'Show' or 'Copy', and the 'Status' is shown as 'Active'.

Which takes you to the information page about your tenancy. Copy the Tenancy OCID from this page.

The screenshot shows the Oracle Cloud Identity interface. The top right corner features a search bar with 'eu-frankfurt-1' and a dropdown arrow, highlighted with a red box. The main content area displays a tenancy profile for 'oscemea001'. It has a green circular icon with a white 'T' and is marked as 'ACTIVE'. The 'Tenancy Information' tab is selected. The 'Tenancy Information' section includes the 'OCID' (...u5mb5q) with options to 'Show' or 'Copy', which is highlighted with a red box. Other details shown include 'Name: oscemea001', 'Home Region: eu-frankfurt-1', and audit retention information. Below this is the 'Object Storage Settings' section, which notes that Amazon S3 Compatibility API Designated Compartment and SWIFT API Designated Compartment are not available at this time. The 'Object Storage Namespace' is listed as 'oscemea001'.

To populate the 'authUserId' and 'keyFingerprint', return to the '**User Details**' page for the user where you configured the public key earlier in this appendix. Copy both the user OCID and fingerprint.

The screenshot shows the Oracle Cloud Identity interface. The top right corner features a search bar with 'eu-frankfurt-1' and a dropdown arrow, highlighted with a red box. The main content area displays a user profile for 'ATPworkshop01'. The user has a green circular icon with a white 'A' and is marked as 'ACTIVE'. The 'User Information' tab is selected. The 'OCID' (...lomtq) is highlighted with a red box. Other details shown include 'Created: Tue, 28 May 2019 17:14:51 GMT', 'Multi-factor authentication: Disabled', and an 'Email' field. The 'Status' is 'Active' and 'Federated' is 'No'. Below this is the 'Capabilities' section, which lists 'Local password: Yes', 'API keys: Yes', 'Auth tokens: Yes', 'SMTP credentials: Yes', and 'Customer secret keys: Yes'. The 'Resources' section is expanded, showing the 'API Keys' section. Under 'API Keys', there is a table with one row. The first column is a green circular icon with a white 'PK'. The second column is the 'Fingerprint' value: '61:85:ae:a6:03:3b:8e:da:27:76:6e:24:03:1fa:5:f9', which is highlighted with a red box. The third column is the 'Time Created': 'Thu, 30 May 2019 13:06:31 GMT'. There is also a 'Displaying 1 API Keys' message and a 'Add Public Key' button.

The final piece of information is the path and name of the private key file. Once this has been saved you are ready to use the bash oci-curl script to run REST commands.



Integrated Cloud Applications & Platform Services

ORACLE®

Oracle Corporation, World Headquarters
500 Oracle Parkway
+1.650.506.7000
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone:
Fax: +1.650.506.7200

CONNECT WITH US

-  blogs.oracle.com/oracle
-  facebook.com/oracle
-  twitter.com/oracle
-  oracle.com

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. This document is provided *for* information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0619