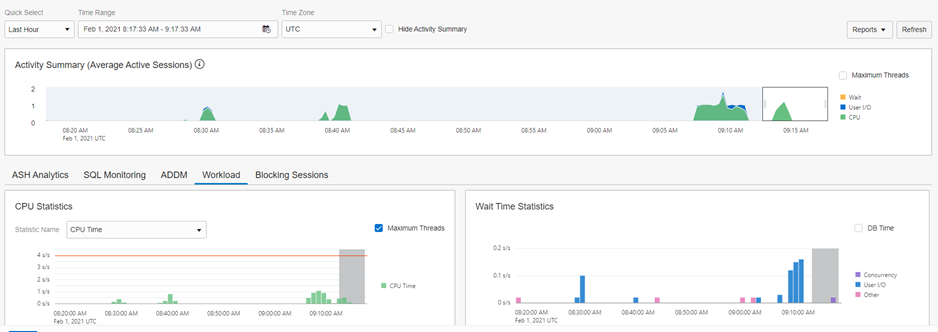
# Monitoring and Ad-hoc scaling up ADW activity for optimal OAC experience



ADW Monitor

## Introduction

In this lab, you will explore capabilities available for your Autonomous Data Warehouse (ADW) using Database Service Console, you use ad-hoc scaling up of ADW for optimal Oracle Analytics Cloud (OAC) experience and check OAC query logs looking for the Logical and Physical SQL.

Estimated Lab Time: 40 minutes

### Objectives

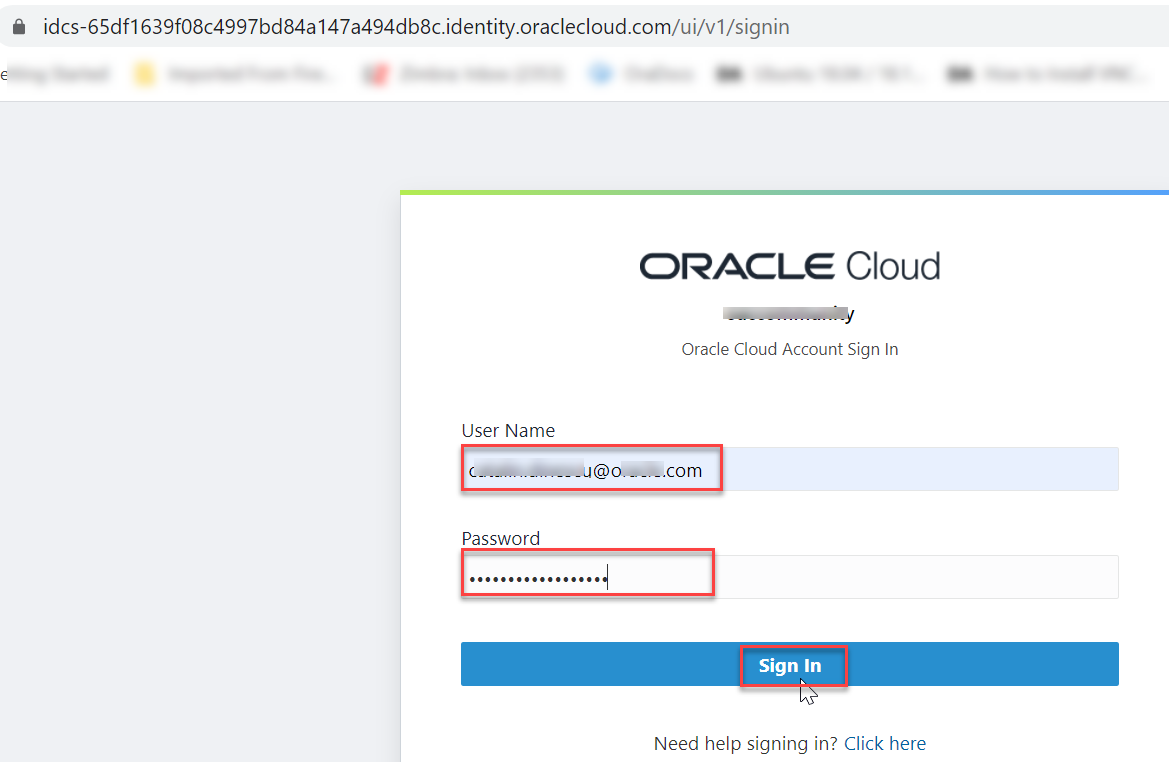
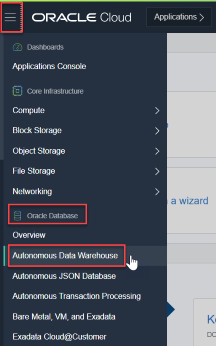
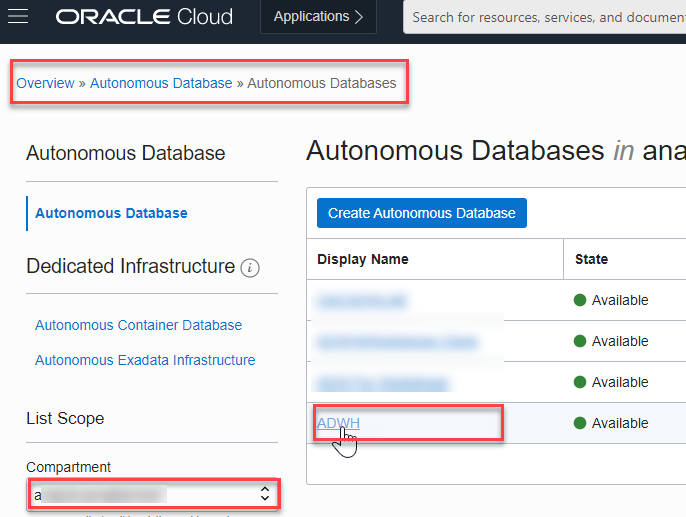
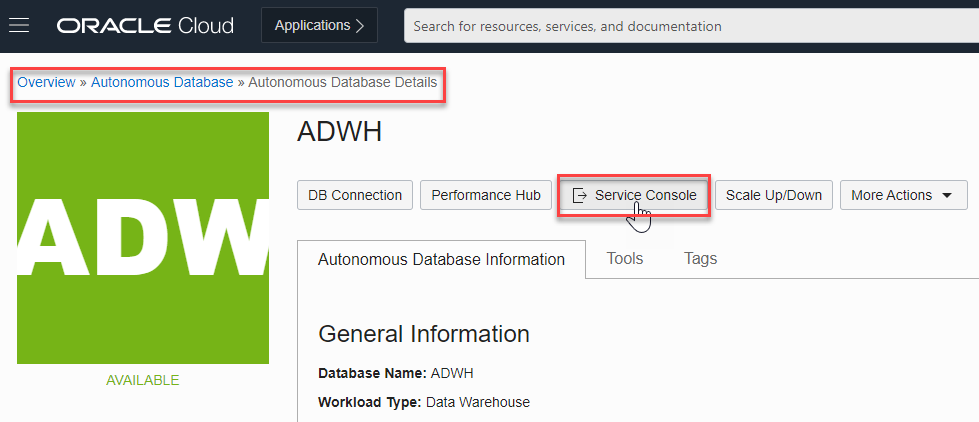
* **Monitoring ADW activity generated by OAC**
* **Ad-hoc scaling up of ADW for optimal OAC experience**
* **Checking OAC query logs**

### Prerequisites

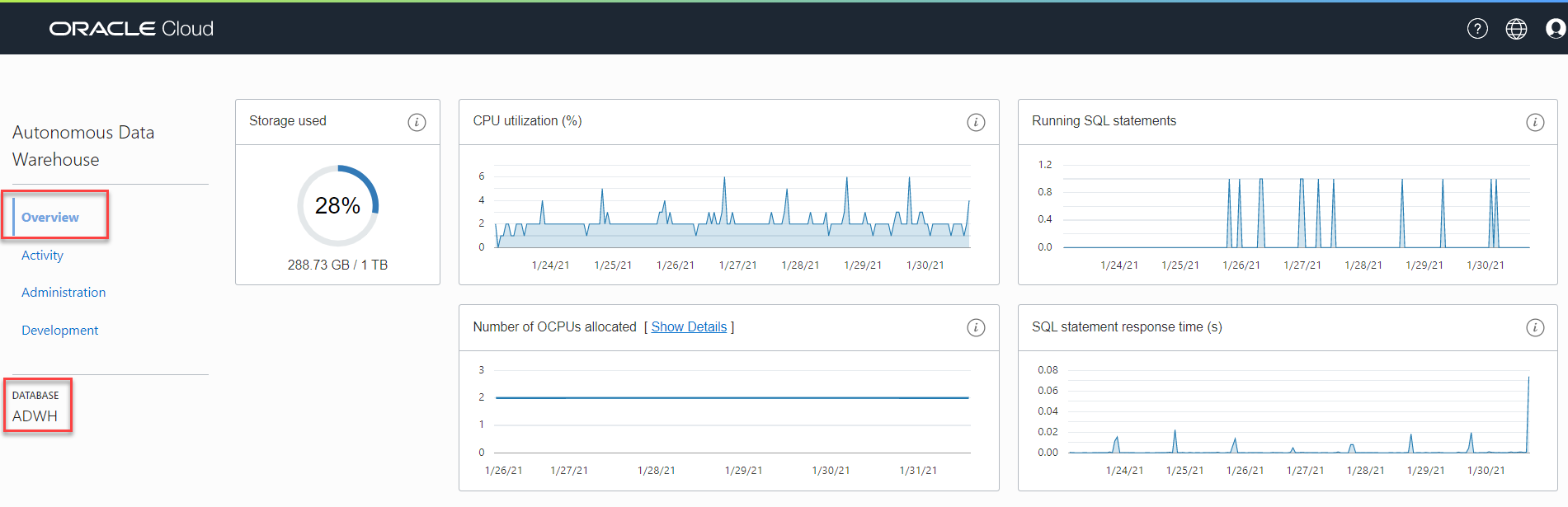
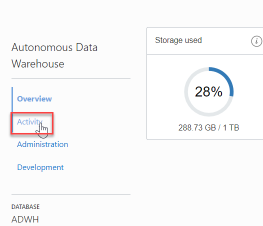
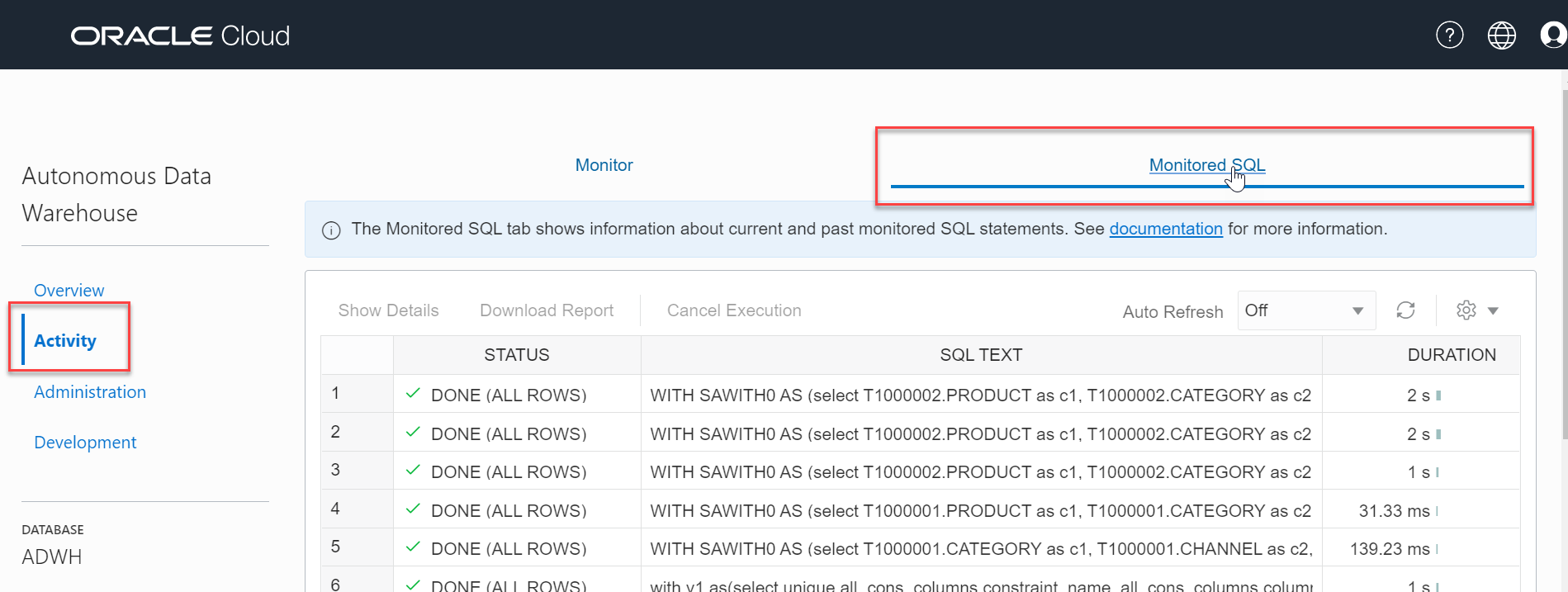
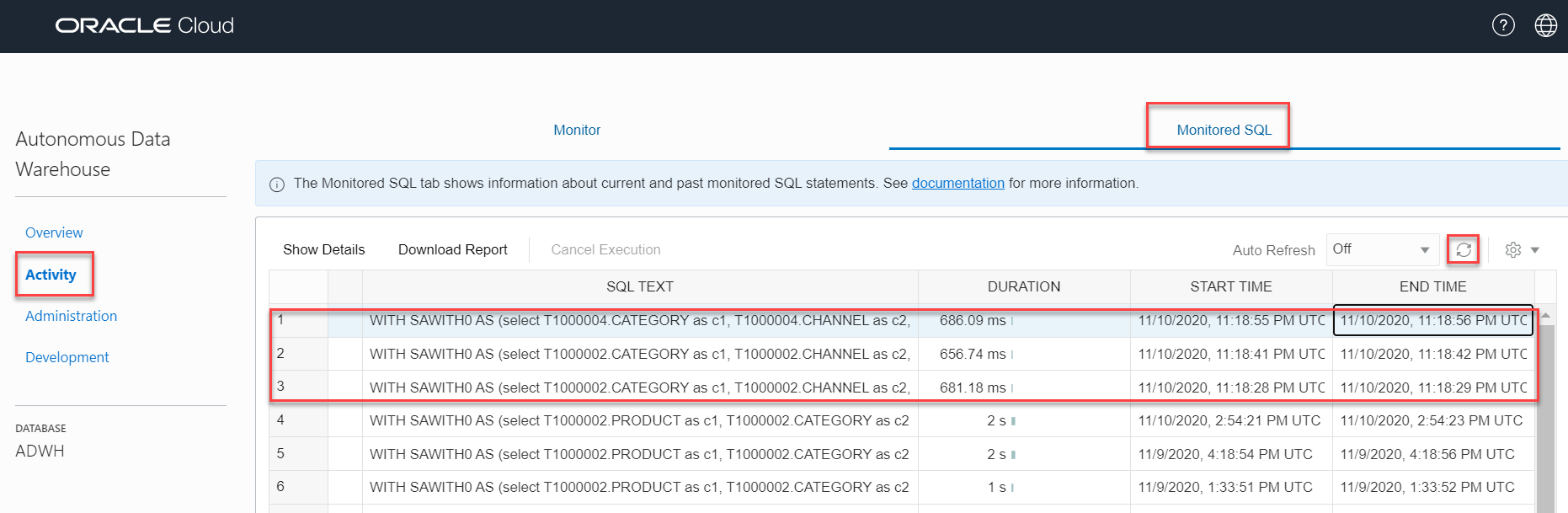
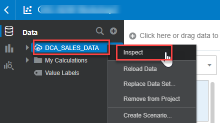
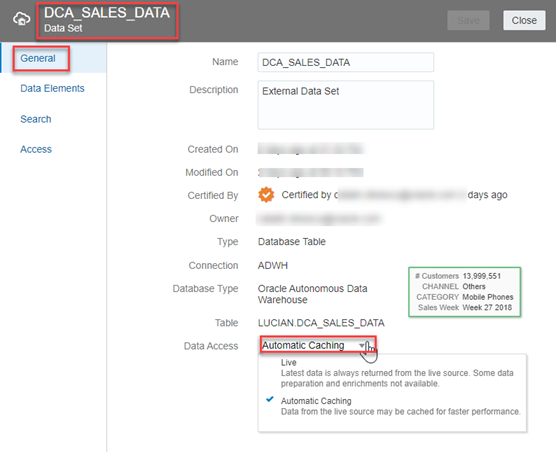
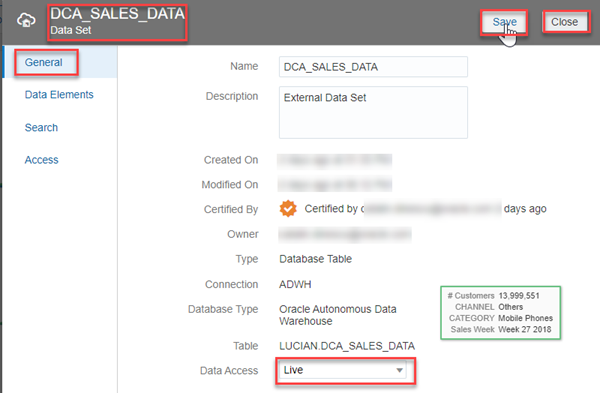
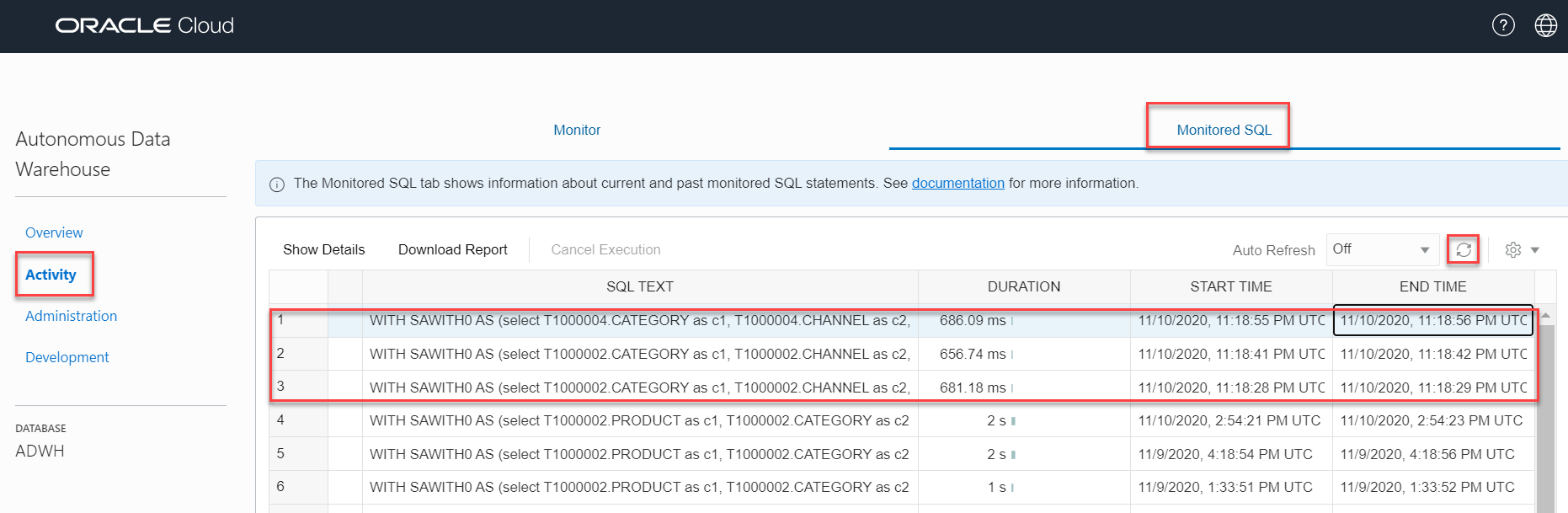
* An Oracle Cloud Free Tier or Paid
* You have completed  
  [“Lab 1: Provisioning your Autonomous Database instance”](%22../../provision-adw/provision-adw.md%22)  
  [“Lab 2: Provisioning your Oracle Analytics Cloud instance”](%22../../provision-oac/provision-oac.md%22)  
  [“Lab 3: Connecting OAC to ADW and adjusting Data Set properties”](%22../../a-connoactoadw/a-connoactoadw.md%22)  
  [Lab 5: Gaining Insights with Visual Data Dialog](%22../../d-gaininsigt/d-gaininsigt.md%22)

## **STEP 1**: Monitoring ADW activity generated by OAC

### Lunch Database Service Console

* The ADW Service Console provides dashboards to monitor the real-time and historical CPU and storage utilization, as well as database activity, like the number of running or queued statements.
* Log in to **Oracle Cloud Console**, select **Autonomous Data Warehouse** from the Hamburger Menu and navigate into your ADW instance.
* 1.1 Enter your **User Name**, **Password** and Click **Sign In.**  
  
* 1.2 Click on the hamburger **MENU** link at the upper left corner of the page.  
  This will produce a drop-down menu, where you select **Autonomous Data Warehouse.**  
  
* 1.3 Select your Compartment and Click on your ADW instance  
  
* 1.4 In your ADW **Database Details** page, click the **Service Console** button.  
  

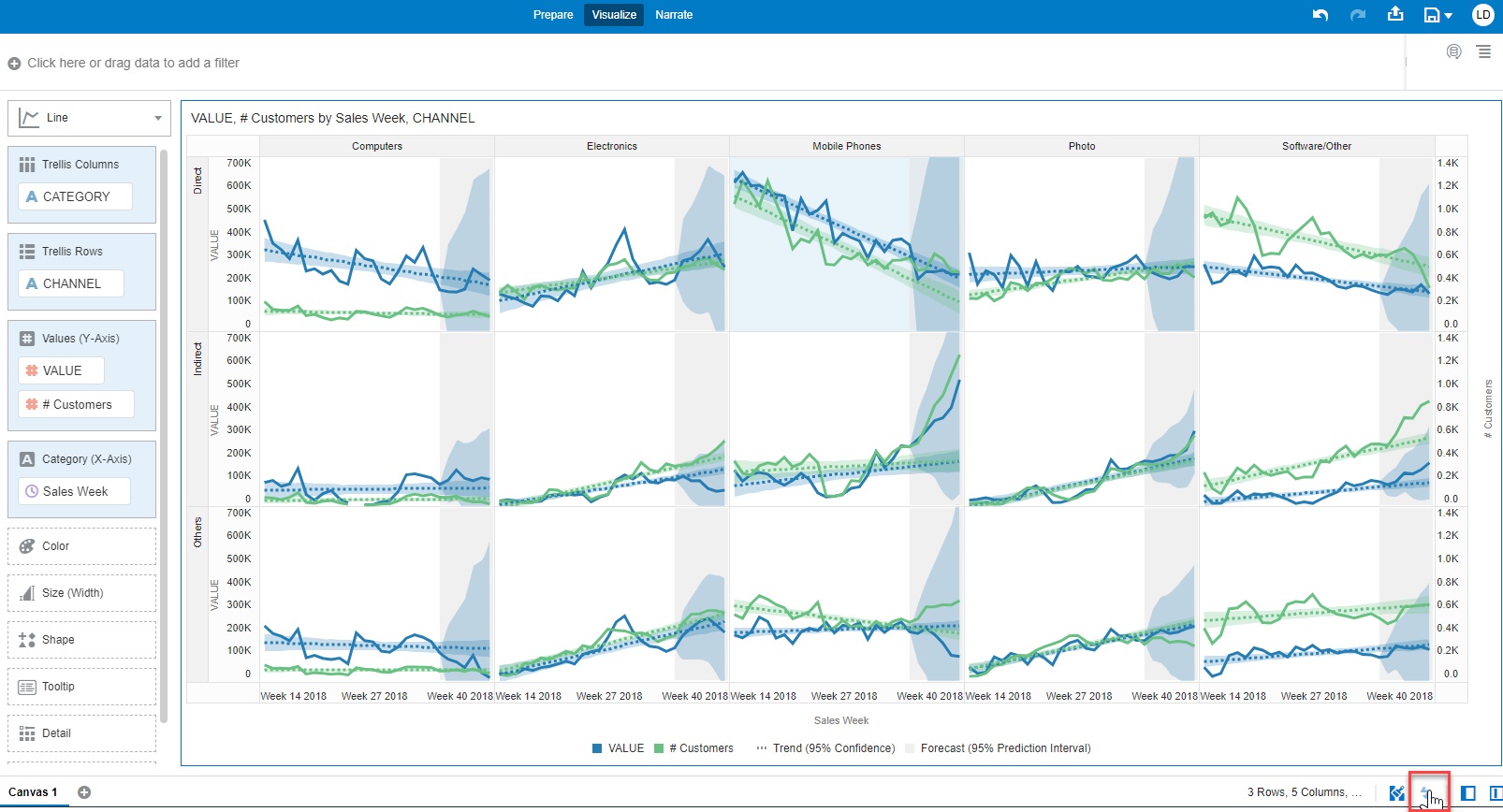
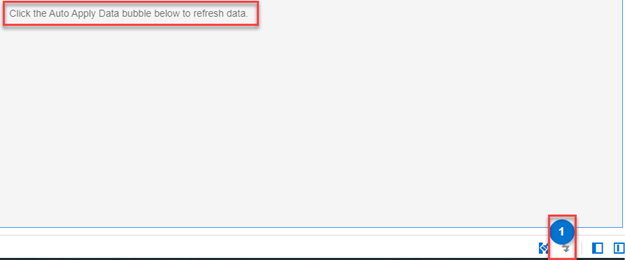
### Examine the Console Overview and Activity Page

* 2.1 A new Canvas Page is opening.  
  Examine the components of the Overview page: *Storage used, CPU utilization, Running SQL statements, Number of OCPUs allocated, SQL statement response times*  
  
* 2.2 To access detailed information about the service’s performance, click the **Activity** tab in the Service Console.  
  
* 2.3 Click the **Monitored SQL** tab to see information about current and past monitored SQL statements.  
  
* 2.4 Back in OAC canvas, lets **refresh** the data and see if new queries get generated in ADW.  
  Open the browser page with your OAC canvas (previous lab) and click **Refresh Data** button from top right.  
  
* 2.5 Going back to ADW tab you notice that no new queries are generated from OAC data refresh. What might be the reason?.  
  
* 2.6 Lets inspect the Data Set. It is set to *automatic caching* and hence, *queries fetched from cache*.  
  Back in OAC, go to the **Data Pane, Data**, Right Click **DCA\_SALES\_DATA** and Select **Inspect**.  
  
* 2.7 Go to **General** tab and notice the **Data Access** is set *Automatic Caching*.  
  
* 2.8 We need to set it to *Live* to make sure queries are fired against the database.  
  Click **Data Access** and change to *Live*, Click **Save** and then **Close**.  
  
* 2.9 Lets regenerate the report to see if fresh queries are getting fired against the database.  
  Repeat Steps 2.4 and 2.5.  
  Now… if we refresh, fresh queries can be seen getting generated.  
  

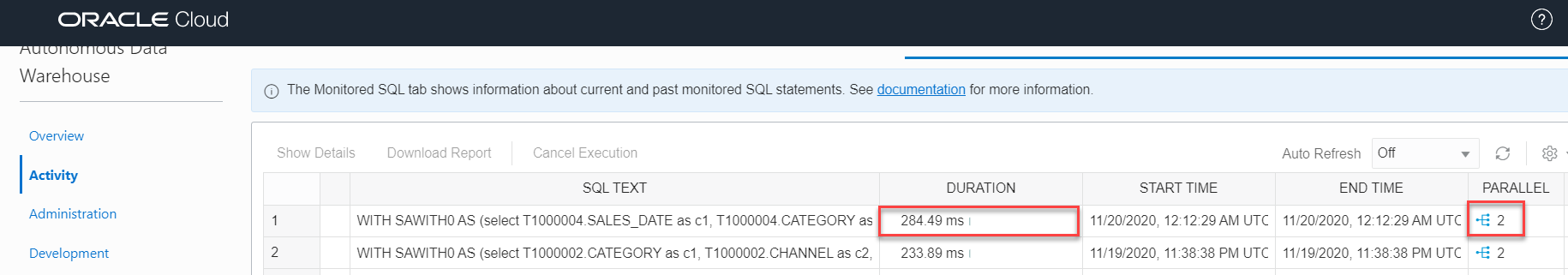
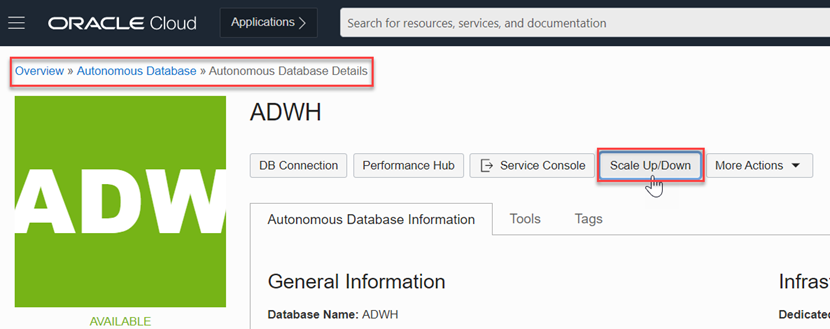
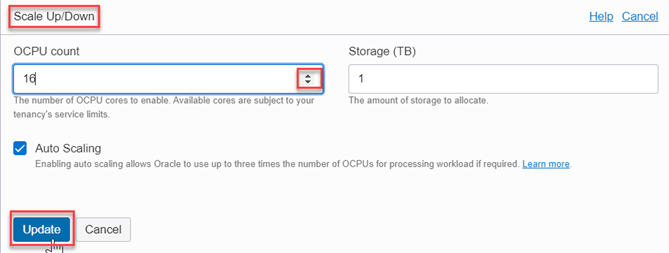
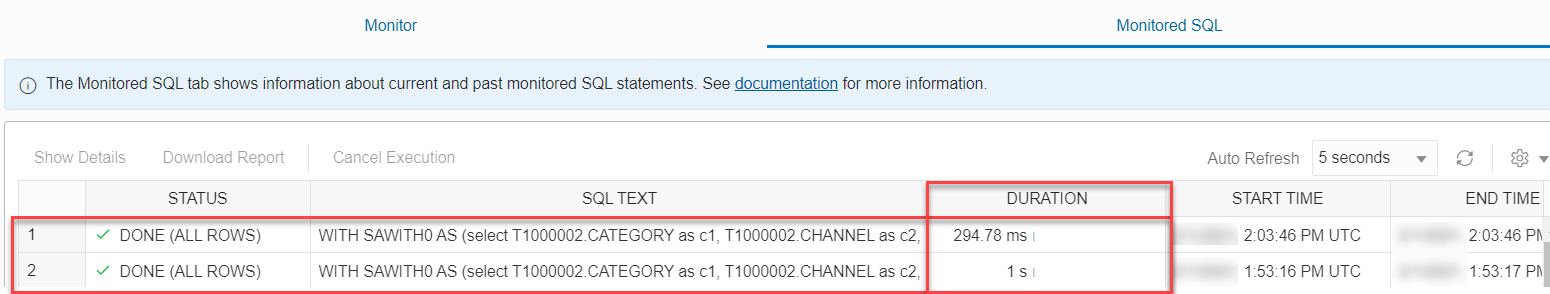
## **STEP 2**: Ad-hoc scaling up of ADW for optimal OAC experience

You are going to create a slightly larger OAC report to mimic the performance scenario by adding greater level of granularity.  
In this section, you will scale up your Oracle Autonomous Data Warehouse (ADW) service to have more CPUs and understand the performance impact over a more complex OAC generated queries.

### OAC Canvas

* Lets create a slightly larger report.
* 1.1 Back in OAC, lets *turn off* *Auto Apply Data* to queues data requests while filtering or constructing visuals (not refresh the report immediately).  
  Open the OAC browser and click **Auto Apply Data** button from bottom right.  
    
  > notice the button turned greyed out Auto Apply Data
* 1.2 Replace *Sales Week* with *Day of Year*.  
  Go to **Data Panel**, Select **Sales\_Date > Day of Year** data element, drag and drop it over **Sales Week** from **Grammar Panel** **Category (X-Axis)** section .  
  That is going to replace the existing data element (week) with a more granular one (day) and therefore increase query complexity and ADW “work”.  
  > Note: in case that the query is displaying an error, you can skip the replace step and use the visualization as is, it would be relevant enough
* Auto Apply Data
* Auto Apply Data
* 1.3 You get the message ***“Click the Auto-Apply Data below to refresh data”*** with a clear indication where to go.  
  
* 1.4 Refresh the data by clicking on **Auto Apply data** and notice that the report takes a while to run.Auto Apply Data

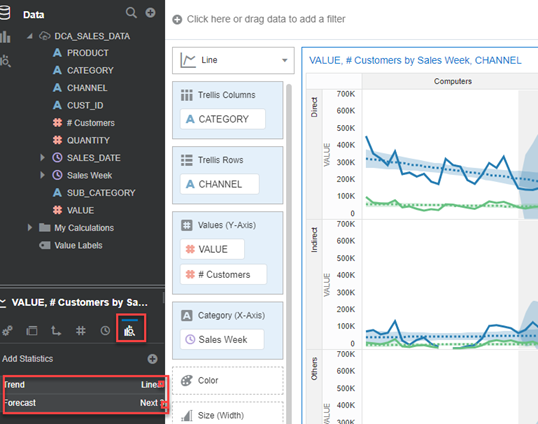
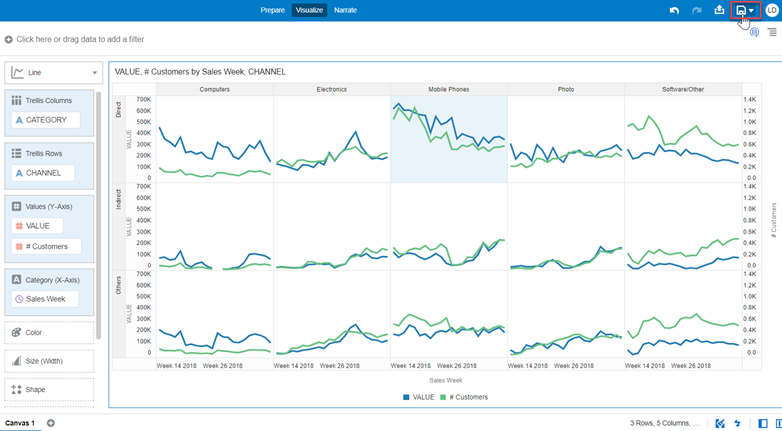
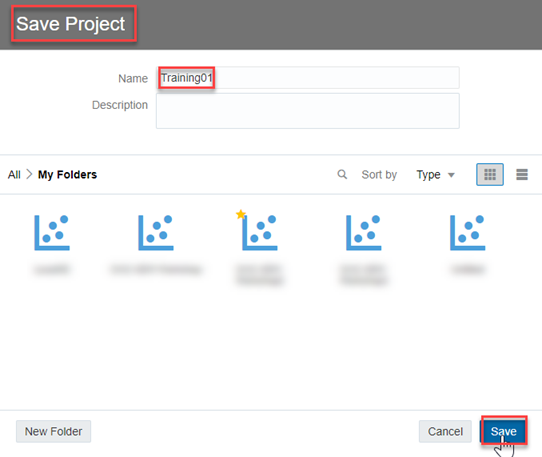
### ADW Service Console

* 2.1 Back in the ADW console, note the duration that the query ran with the current compute resources.  
  You are going to compare this duration with the duration after you scale-up the ADW compute resources.  
  
* 2.3 Lets **Scale-Up** the ADW to use the maximum number of OCPUs that you can spare (16 OCPUs is a good number, but not a mandatory one). It takes about 90 seconds for the scale up process to complete.  
  > Note: in your training environments, if you are using the *Free Tier* offer, or you have limited resources, you might not be able to actually scale-up the ADW to use 16 OCPUs; please use as reference this [video](https://www.youtube.com/watch?app=desktop&v=koOLAY48lZM&list=PL6gBNP-Fr8KVdvmZUg7eqXljXnFhidBV-&index=2) starting from minute 10:1
* Go back to the **Cloud Console** you used during the provisioning exercise and open the database instance’s **Details** page. From the **action menu**, click **Scale Up/Down** 
* 2.4 Fill in the form with the following information:
  + **Change** **OCPU Count** to 16 (or to a value that you can afford)
  + **Auto-Scaling**: Disabled  
    > if you have enough resources you can enable auto-scaling; this feature allows your database to use up to three times the current base number of CPU cores at any time -> basically if this feature was disabled, just by enabling and leave the same number of OCPUs. you’ll get extra performance when need it.
* Click **Update** 
* 2.5 Back in OAC page, lets **refresh** the data and see how fast the new queries get generated in a scaled-up ADW instance.  
  Open the browser page with your OAC canvas and click **Refresh Data** button from top right.  
  Refresh Data
* 2.6 Back in the ADW **Service Console > Activity > Monitored SQL**, note the duration that the query ran with the updated compute resources.  
  The new query generated has taken just a few millie seconds to complete with 16 OCPUs being utilized.  
  

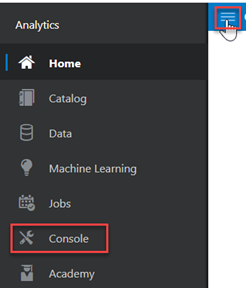
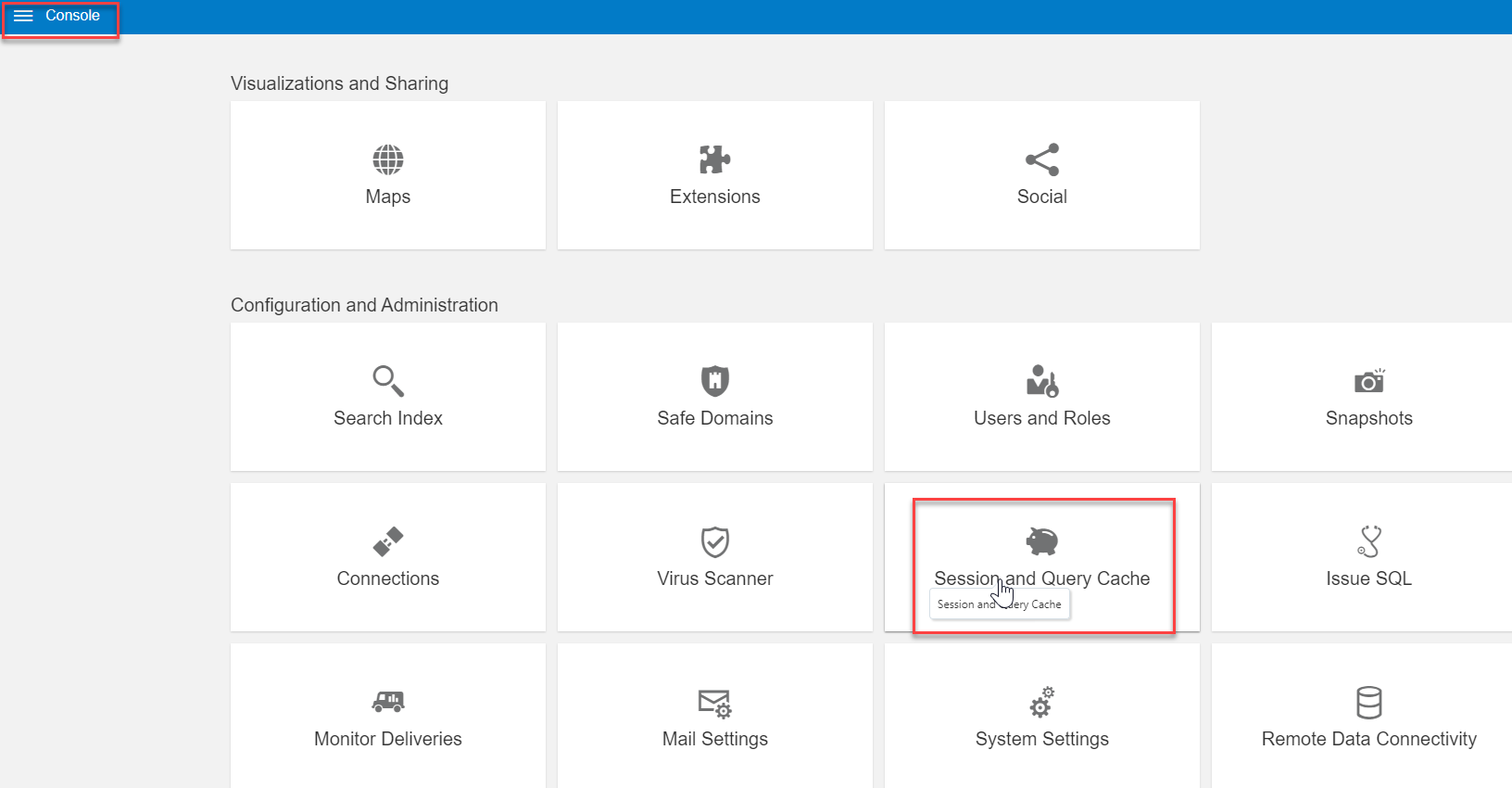
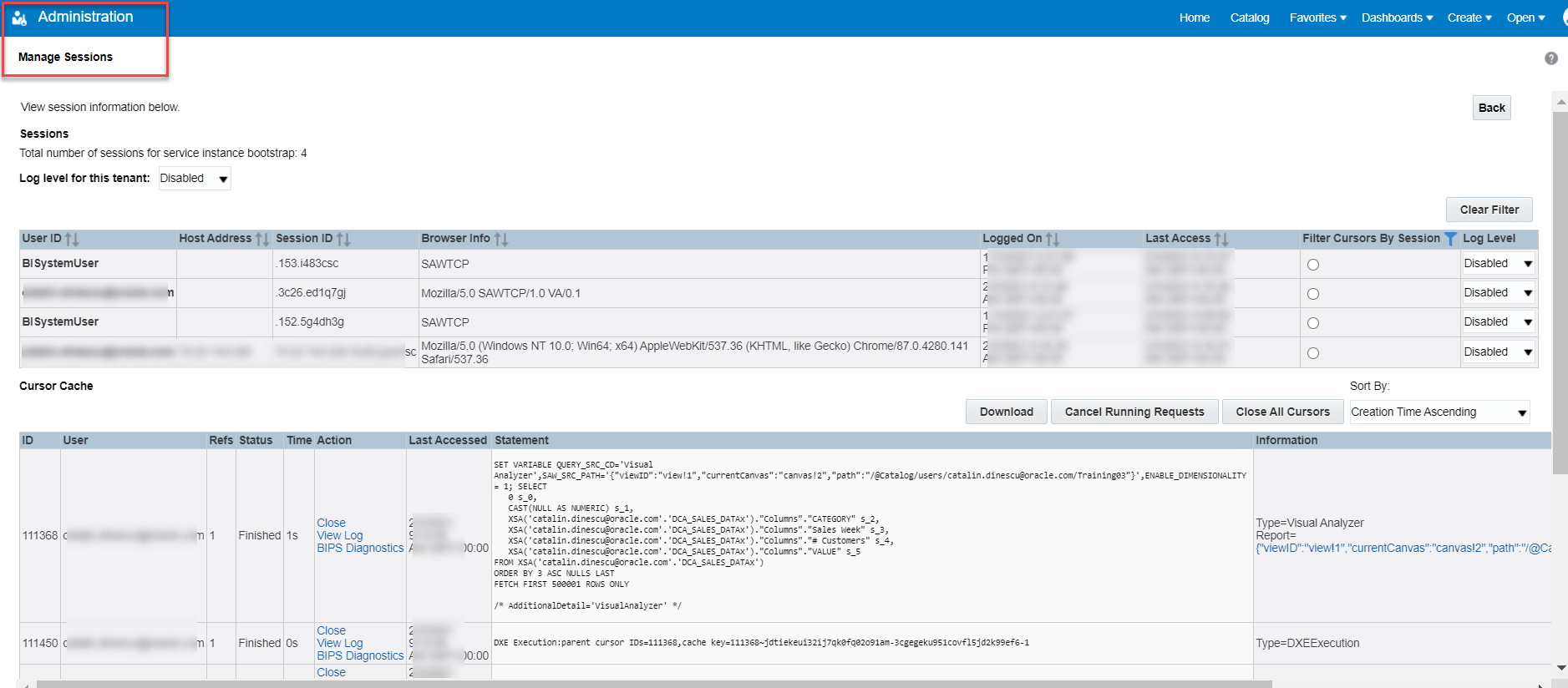
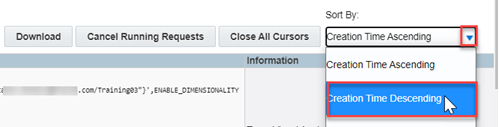
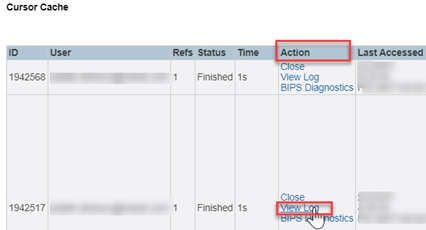
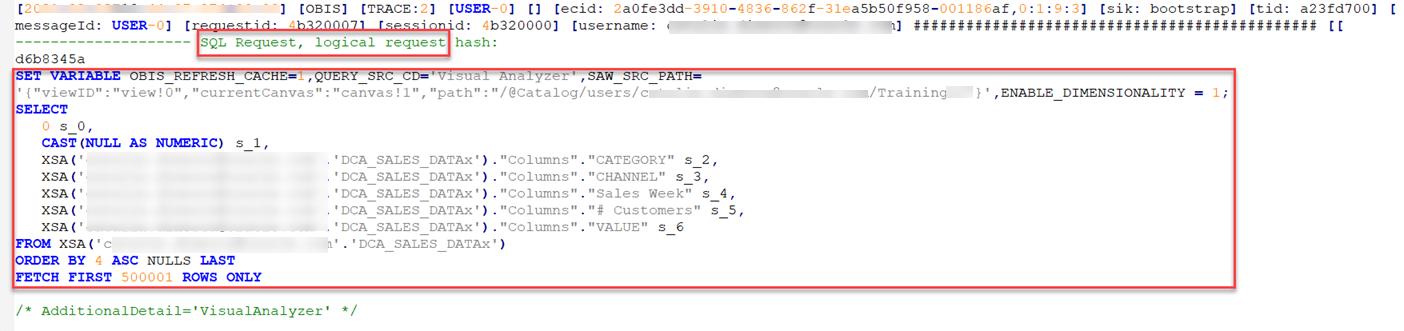
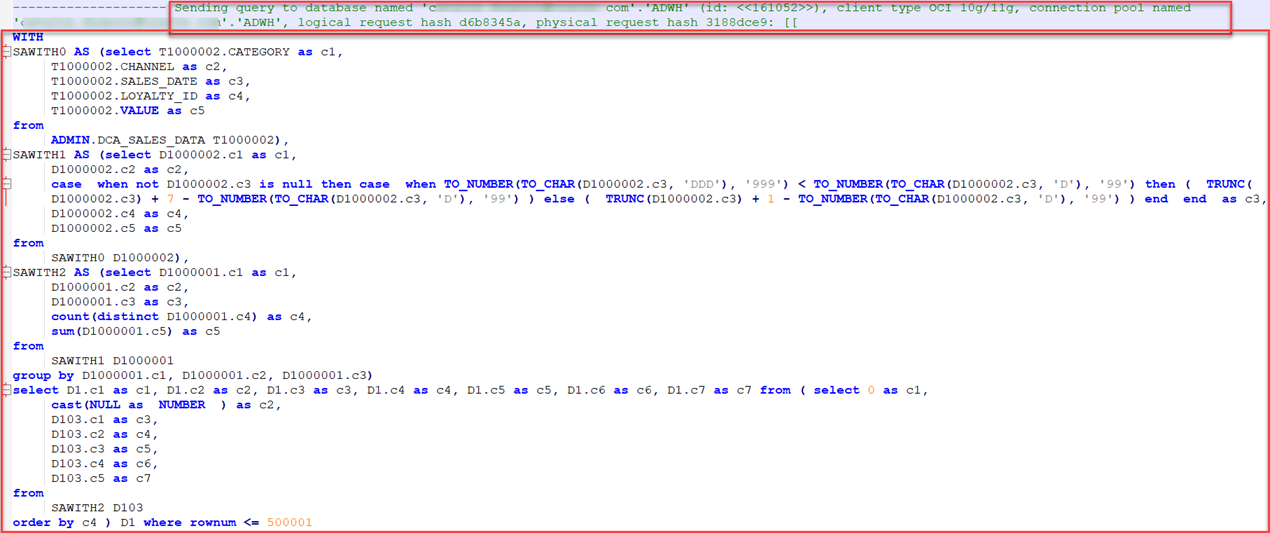
## **STEP 3**: Checking OAC Query Logs

Administrators can examine the underlying SQL query requests that are run.

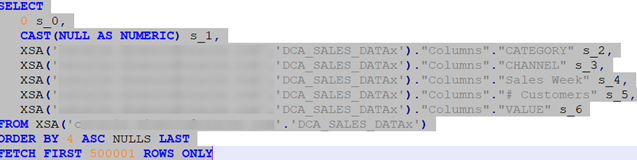
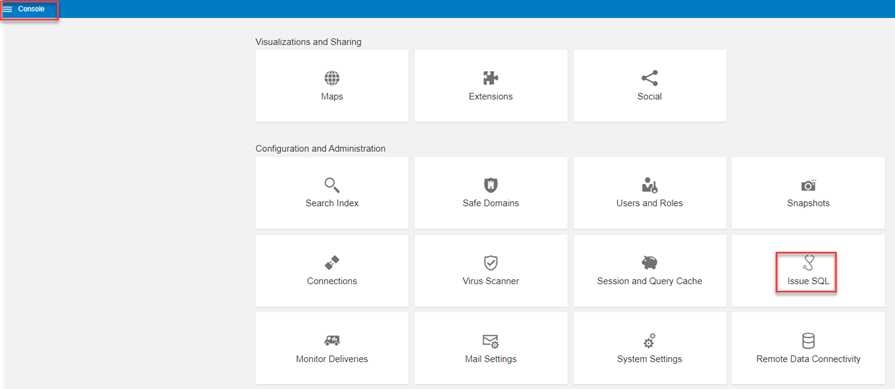
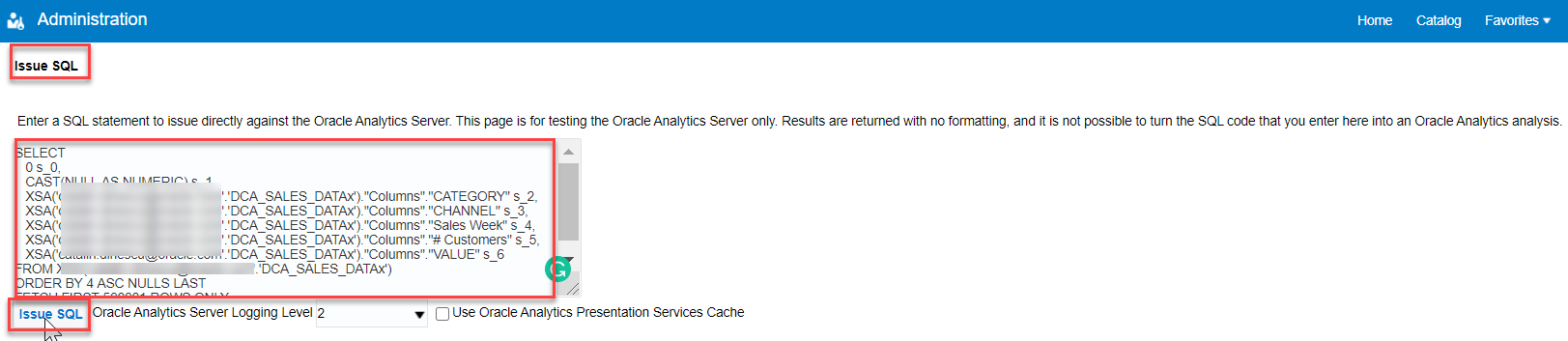
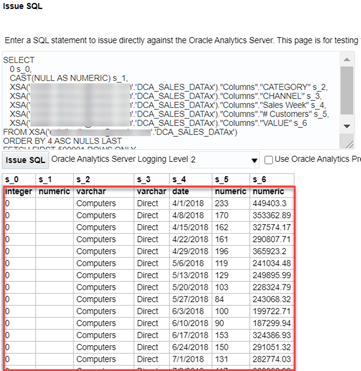
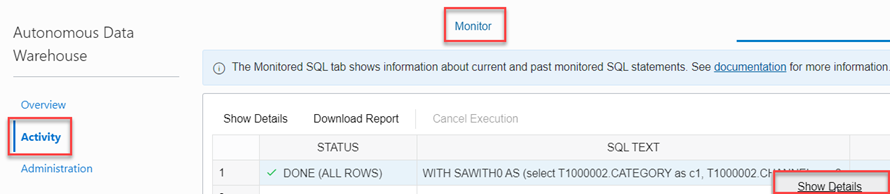
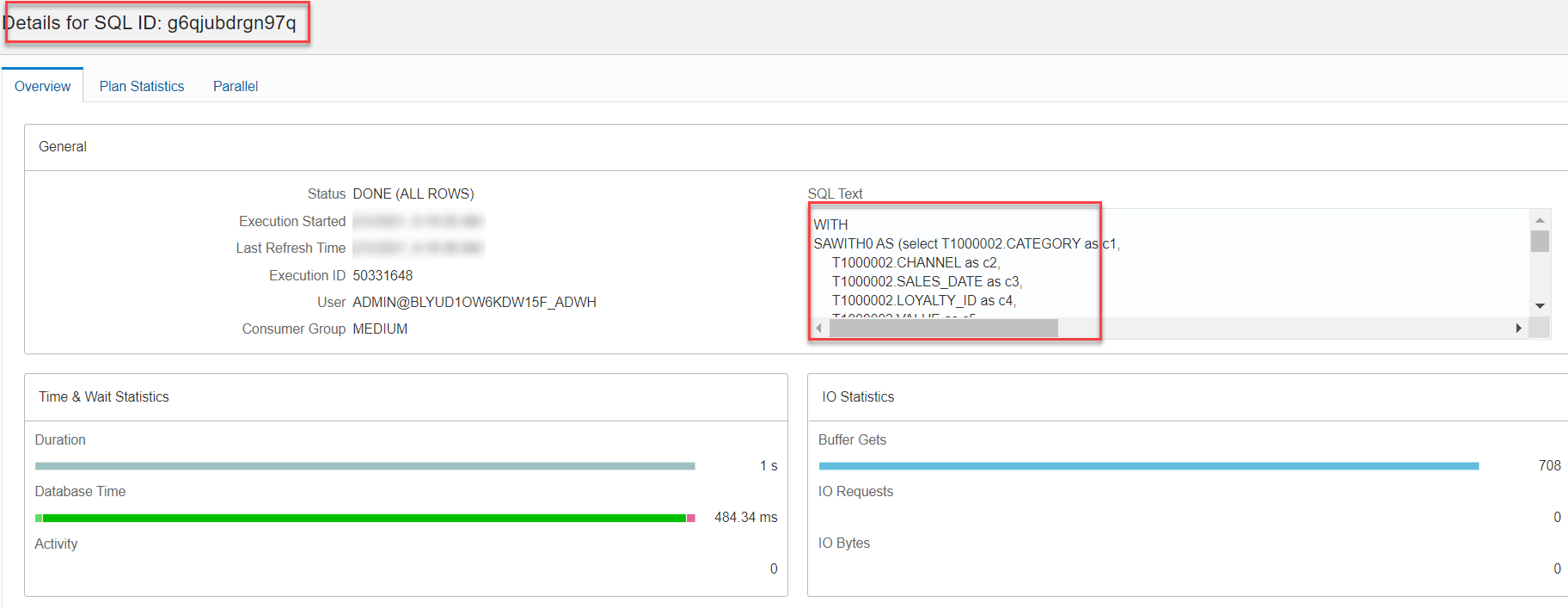
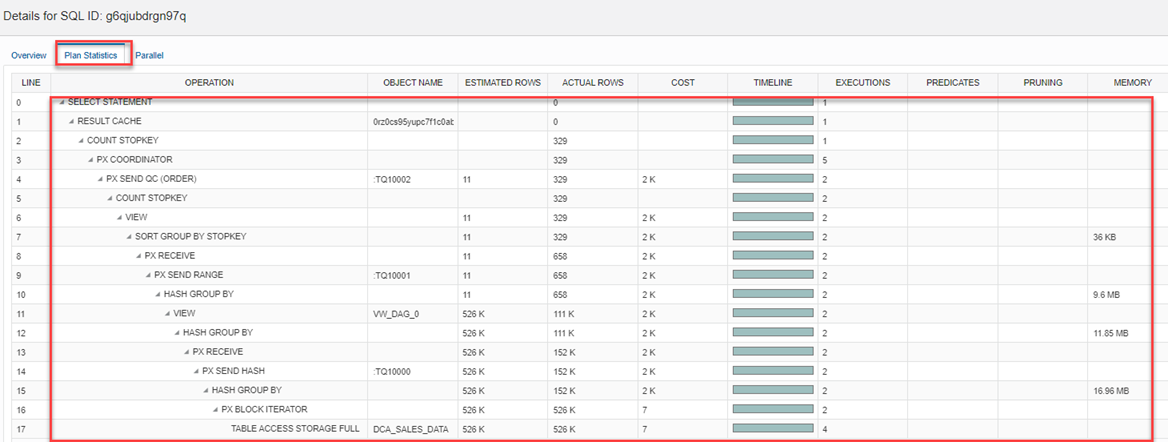
### OAC Save Project

* Going back to our visualization lets Remove Analytics.
* 1.1 Click your visualization and you get the Visualization Properties Pane on the lower left of the screen.  
  Click on **Analytics** Properties and hover the mouse on the **Trend** till you get an red x, Click on it and do the same with **Forecast**.  
  
* 1.2 Lets save our project for further analysis.  
  Click on **Save** icon from upper top right of the screen.  
  
* 1.2 Save Project as Training01 (or any name we want to).  
  Type **Name**: **Training01** an Click **Save** button.  
  

### OAC Query Log

* Lets now check the query logs generated inside of OAC.
* 1.1 Click **Go Back** left arrow from upper left of the screen to go to the Analytics Home Page. Go Back
* 1.2 Click **Navigator** and then Select **Console**.  
  
* 1.3 Click **Sessions and Query Cache** under *Configuration and Administration* section.  
  
* 1.4 The logs screen shows all the query logs and it can be sorted in ascending/descending order.  
  Locate the **Cursor Cache** section, and review the query information recorded there.  
  
* 1.5 Sort the query log in a descending order.  
  **Sort By**> Select **Order Time Descending**.  
  
* 1.6 Pick-up the most recent one and Click **View Log**.  
  
* 1.7 Logs contain logical SQL generated by the BI Server and the physical SQLs fired against the database.  
  Scroll down the log file and you’ll find on top the **Logical SQL** and further down the **Physical SQL** as well.  
    
  > Note: for the physical SQL search the log for “*Sending query to database named*” 

### Bonus

* Lets test the Logical SQL with OAC and see the Physical SQL in the ADW Console as well.
* 1.1 **Copy** the **Logical SQL** from the log file. 
* 1.2 Go back to **Console** and Click **Issue SQL** under *Configuration and Administration* section.  
  
* 1.3 Paste the *Logical SQL* and Click **Issue SQL** button.  
  
* 1.4 You can see the results in an unformatted table.  
  
* 1.5 Go to the ADW **Service Console > Activity > Monitored SQL**, select your query, right-click and select **Show Details**.  
  
* 1.6 You can see in the **Overview** tab the Physical **SQL Text** which is identical with the one from OAC query log.  
  
* 1.7 You also browse through the other tabs as **Plan Statistics** to see other relevant query information.  
  

You have just finished learning how to monitor ADW activity generated by OAC, as well as the benefits of ad-hoc scaling up ADW for optimal OAC experience. You have also learn how to check OAC query logs and find the Logical and Physical generated SQL.

**This concludes this lab. Please proceed to the next lab in the Content menu.**

## Want to Learn More?

* [Where is the Log for My Data-Flow In Oracle Analytics?](https://blogs.oracle.com/analytics/where-is-the-log-for-my-data-flow-in-oracle-analytics)
* [Analyze SQL Queries and Logs](https://docs.oracle.com/en/cloud/paas/analytics-cloud/acabi/monitor-users-and-activity-logs.html#GUID-24782185-AED4-4F6B-BBF6-51AE9C0C7962)
* [Scaling and Performance in the Autonomous Database Workshop](https://apexapps.oracle.com/pls/apex/dbpm/r/livelabs/view-workshop?wid=608&session=121182585525212)

## **Acknowledgements**

* **Author** - Lucian Dinescu, Product Strategy, Analytics
* **Contributors** -
* **Last Updated By/Date** - Lucian Dinescu, February 2021

## Need Help?

Please submit feedback or ask for help using our [LiveLabs Support Forum](https://community.oracle.com/tech/developers/categories/livelabsdiscussions). Please click the **Log In** button and login using your Oracle Account. Click the **Ask A Question** button to the left to start a *New Discussion* or *Ask a Question*. Please include your workshop name and lab name. You can also include screenshots and attach files. Engage directly with the author of the workshop.  
Please contact us at analyticsenablement\_us\_grp@oracle.com for any suggestions or challenges you might have with **Oracle Analytics** labs.

If you do not have an Oracle Account, click [here](https://profile.oracle.com/myprofile/account/create-account.jspx) to create one.