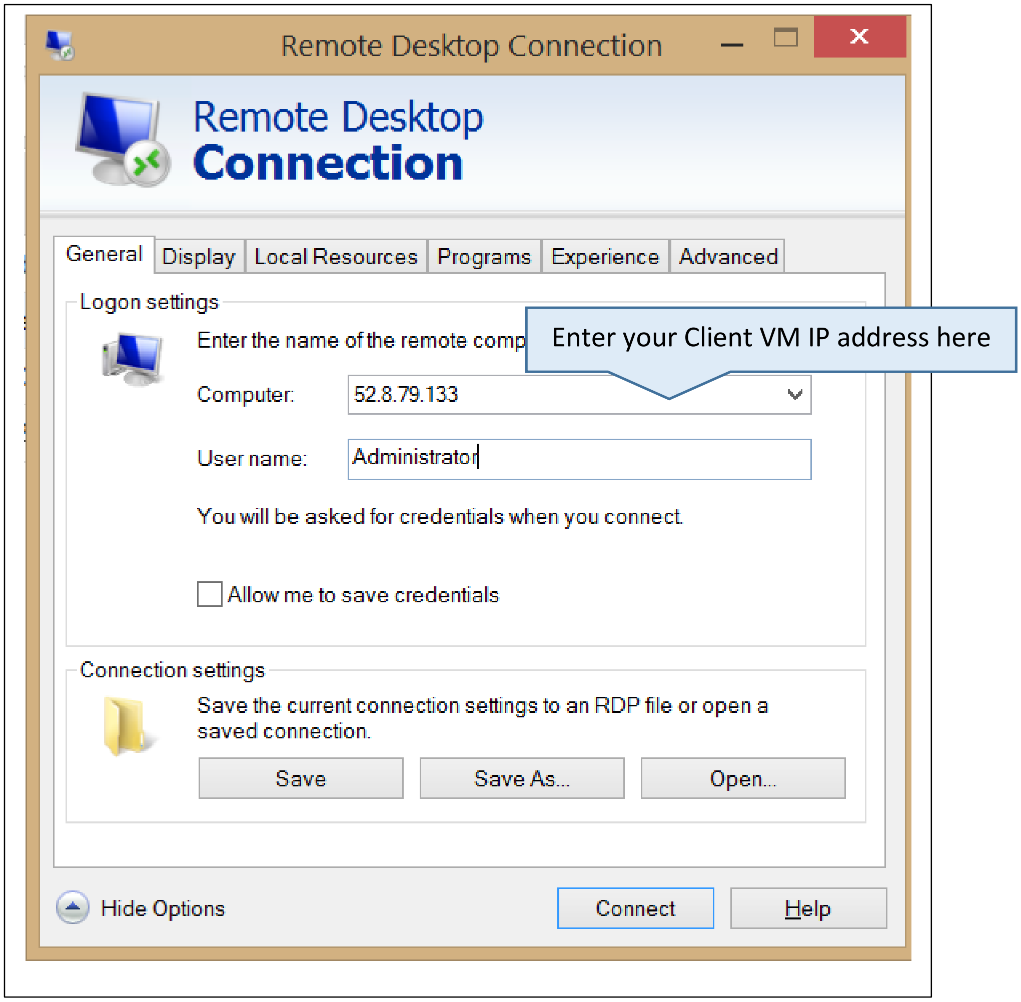
You will have two VMs assigned to you for this hands-on training lab, a windows-based client VM and a server VM. The client VM will contain the majority of the software which we will use in the lab including RStudio, SQL Developer, and OBIEE's BI Admin tool. You will open connections to your own individual server VM through those programs and through the ODBC system administration tool.

To open a connection to your client VM, use Remote Desktop and enter the IP address of your Client VM.

User name = Administrator

Password = odtug2015



Enter the name of the remote comp

Computer: 52.8.79.133

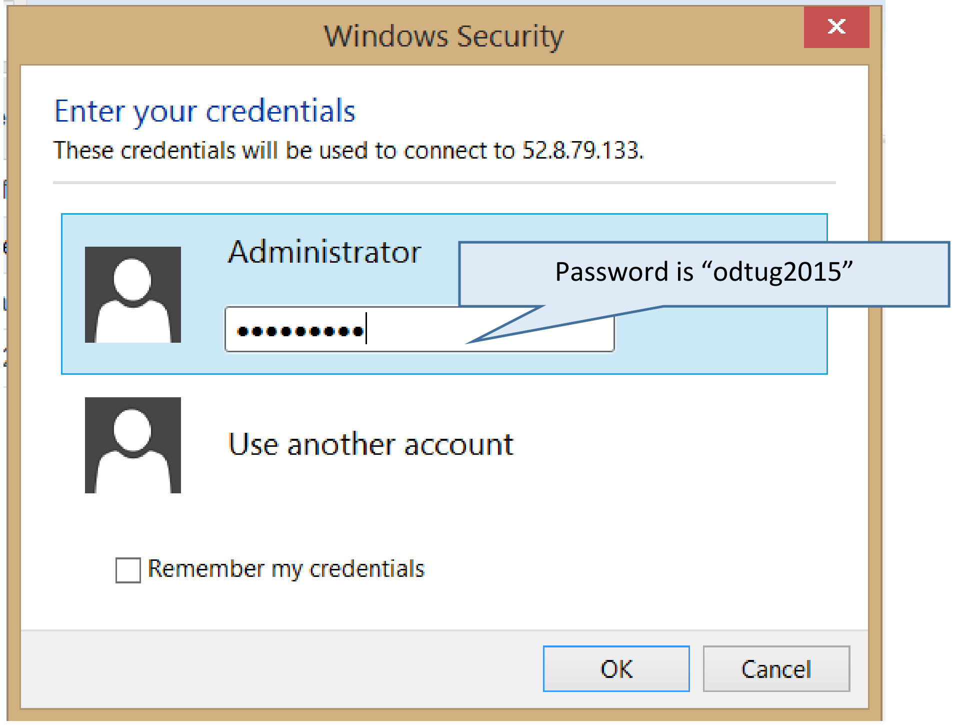
User name: Administrator

You will be asked for credentials when you connect.

□ Allow me to save credentials

Save the current connection settings to an RDP file or open a saved connection.

Enter your Client VM IP address here



Windows Security

Enter your credentials

These credentials will be used to connect to 52.8.79.133.

Administrator

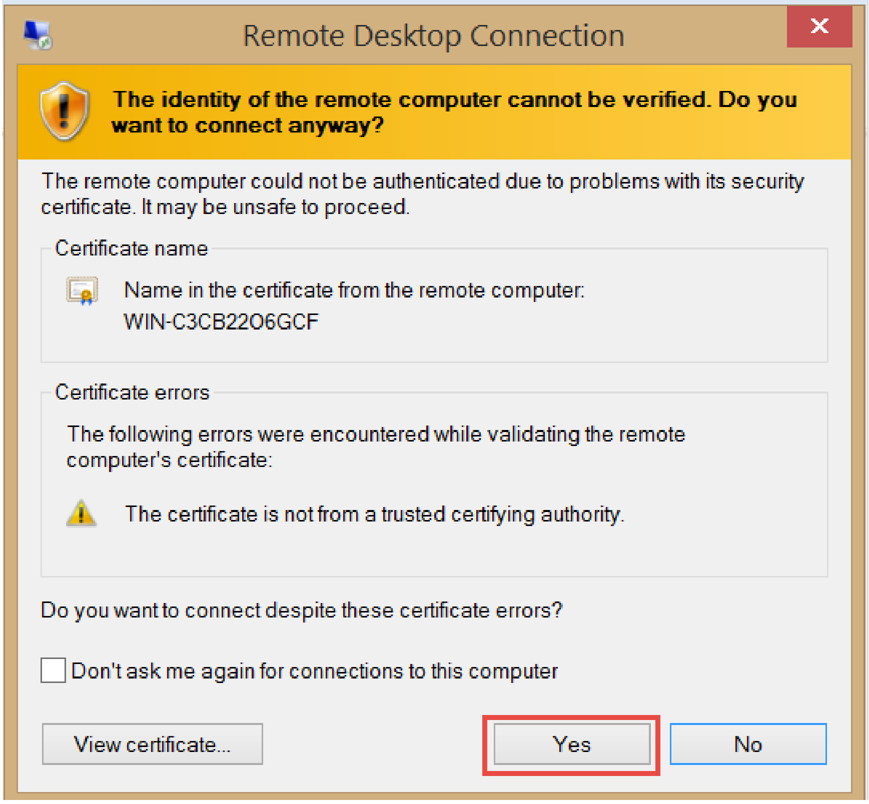
Password is “odtug2015”

Use another account

☐ Remember my credentials

Click OK.

You may see a warning. Ignore it and click Yes.



Remote Desktop Connection

The identity of the remote computer cannot be verified. Do you want to connect anyway?

The remote computer could not be authenticated due to problems with its security certificate. It may be unsafe to proceed.

Certificate name

Name in the certificate from the remote computer:
WIN-C3CB22O6GCF

Certificate errors

The following errors were encountered while validating the remote computer's certificate:

⚠ The certificate is not from a trusted certifying authority.

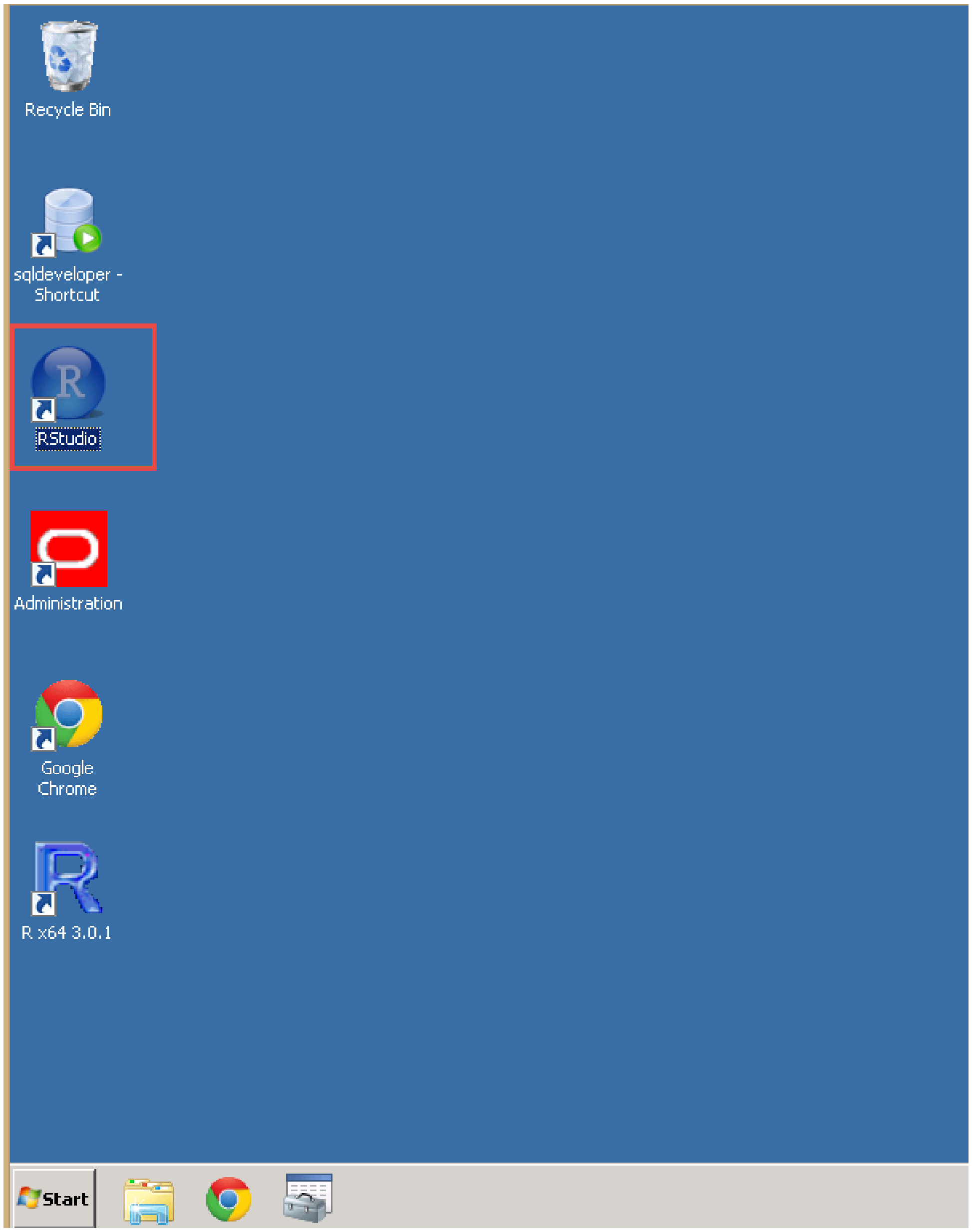
Do you want to connect despite these certificate errors?

☐ Don't ask me again for connections to this computer

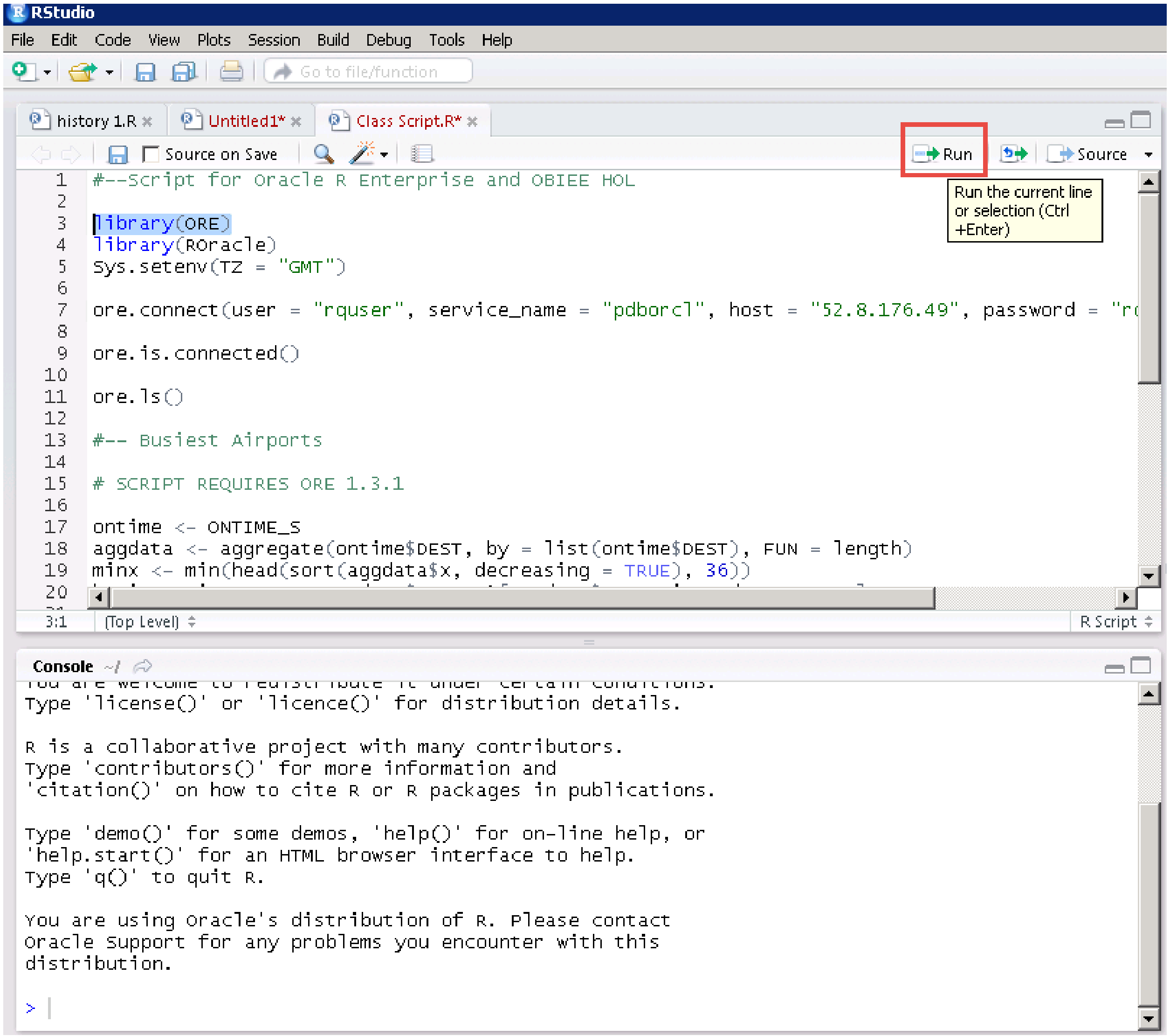
In this exercise, we will create an analysis in R including a table and a custom box plot visualization and then set up an automatic work flow in Oracle database to spin up an R instance on the database server machine and publish the table and graph on an OBIEE dashboard.

We will start with the original R script. We will use the RStudio development environment to establish a connection to our database server VM and then run our script.

Launch RStudio by double-clicking the RStudio icon on your desktop.



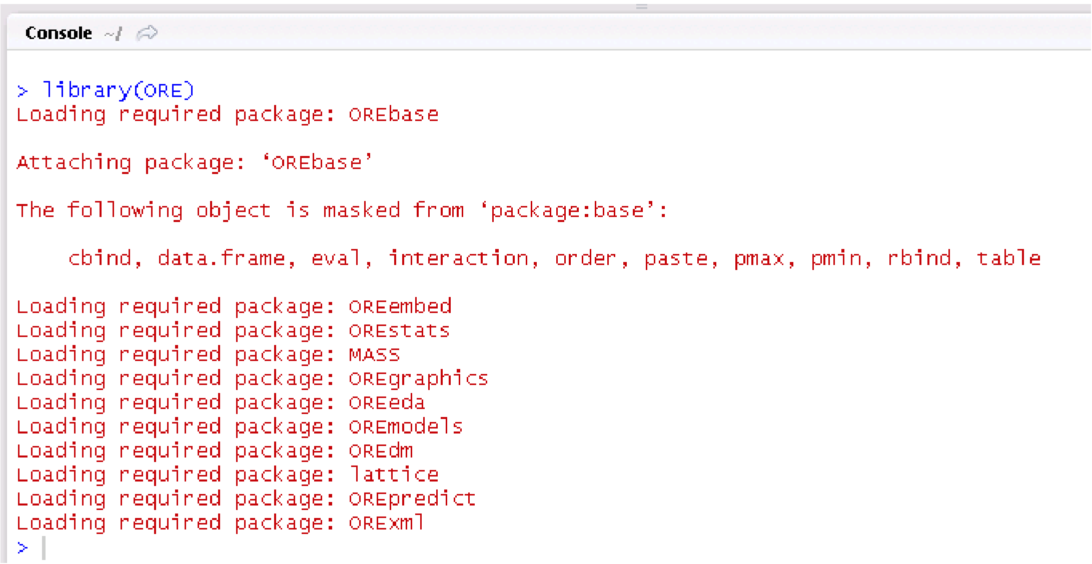
You should see an R script in the upper left hand window pane as shown in the screen shot below. Highlight the first line of the script and click the “Run”. The **library(ORE)** function will load the ORE libraries that have already been installed on your client VM. This enables you to run custom functions that translate R code into SQL code through the Oracle R Enterprise transparency layer.



```
#--Script for Oracle R Enterprise and OBIEE HOL

library(ORE)
library(ROracle)
Sys.setenv(TZ = "GMT")

ore.connect(user = "rquser", service_name = "pdborcl", host = "52.8.176.49", password = "r(

ore.is.connected()

ore.ls()

#-- Busiest Airports

# SCRIPT REQUIRES ORE 1.3.1

ontime <- ONTIME_S
aggdata <- aggregate(ontime$DEST, by = list(ontime$DEST), FUN = length)
minx <- min(head(sort(aggdata$x, decreasing = TRUE), 36))
```

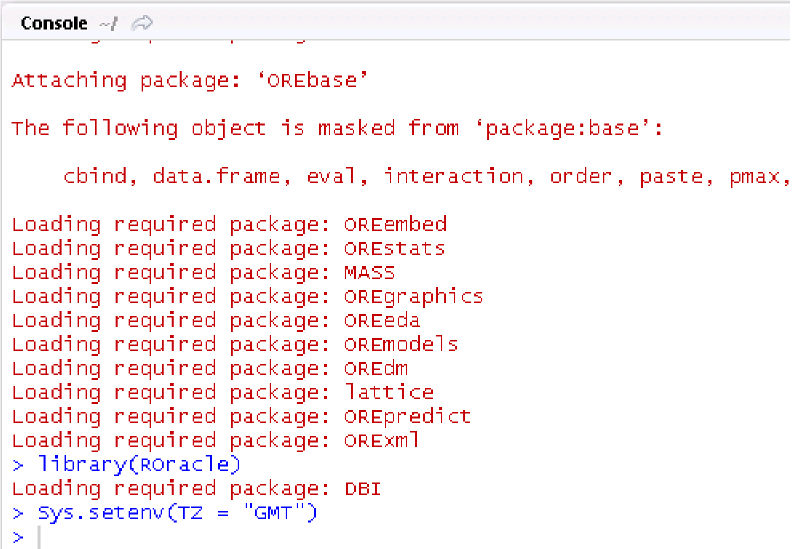
```
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

You are using Oracle's distribution of R. Please contact
Oracle Support for any problems you encounter with this
distribution.

>
```

After running library(ORE) your console panel should look like the below screen.



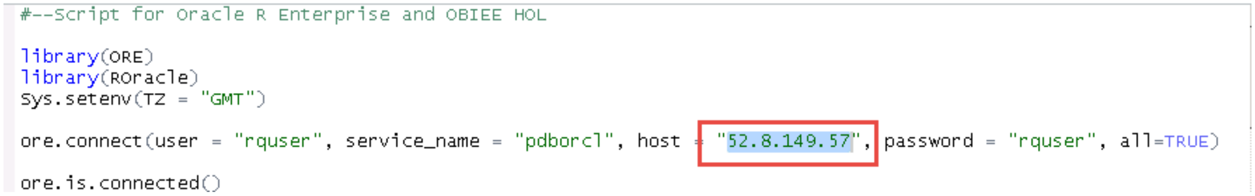
```
> library(ORE)
Loading required package: OREbase

Attaching package: ‘OREbase’

The following object is masked from ‘package:base’:

    cbind, data.frame, eval, interaction, order, paste, pmax, pmin, rbind, table

Loading required package: OREembed
Loading required package: OREstats
Loading required package: MASS
Loading required package: OREgraphics
Loading required package: OREeda
Loading required package: OREmodels
Loading required package: OREdm
Loading required package: lattice
Loading required package: OREpredict
Loading required package: OREXml
> |
```

Highlight the second line of the script **library(ROracle)** and the third line of the code **Sys.setenv(TZ = “GMT”)** and click “Run”. You can either execute R scripts one line at a time or highlight several lines and execute them together. Your console should look like the screen shot below.



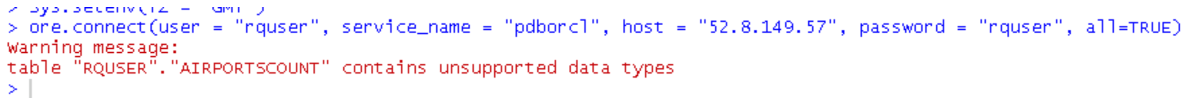
```
Attaching package: ‘OREbase’

The following object is masked from ‘package:base’:

    cbind, data.frame, eval, interaction, order, paste, pmax,

Loading required package: OREembed
Loading required package: OREstats
Loading required package: MASS
Loading required package: OREgraphics
Loading required package: OREeda
Loading required package: OREmodels
Loading required package: OREdm
Loading required package: lattice
Loading required package: OREpredict
Loading required package: OREXml
> library(ROracle)
Loading required package: DBI
> Sys.setenv(TZ = "GMT")
> |
```

Next we will connect to your database server VM. Highlight the IP address in the connection string using the ore.connect function as shown below and paste in the IP address for your server VM. Click the “Run” button to run the ore.connect line of code and connect to your server VM.



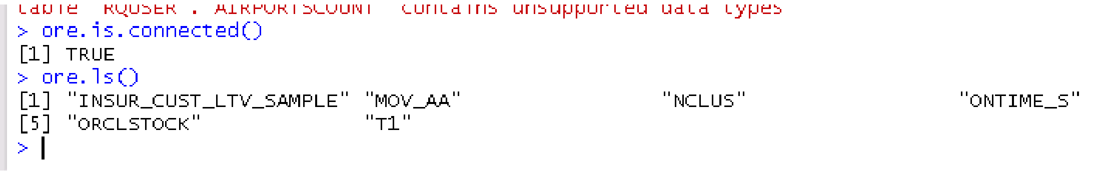
```
#--Script for Oracle R Enterprise and OBIEE HOL

library(ORE)
library(ROracle)
Sys.setenv(TZ = "GMT")

ore.connect(user = "rquser", service_name = "pdborcl", host = "52.8.149.57", password = "rquser", all=TRUE)

ore.is.connected()
```

It may take a few minutes for your client VM to connect to your server VM, so please be patient. Once the connection is established, you should see the below warning message in your console (don't worry about it.)



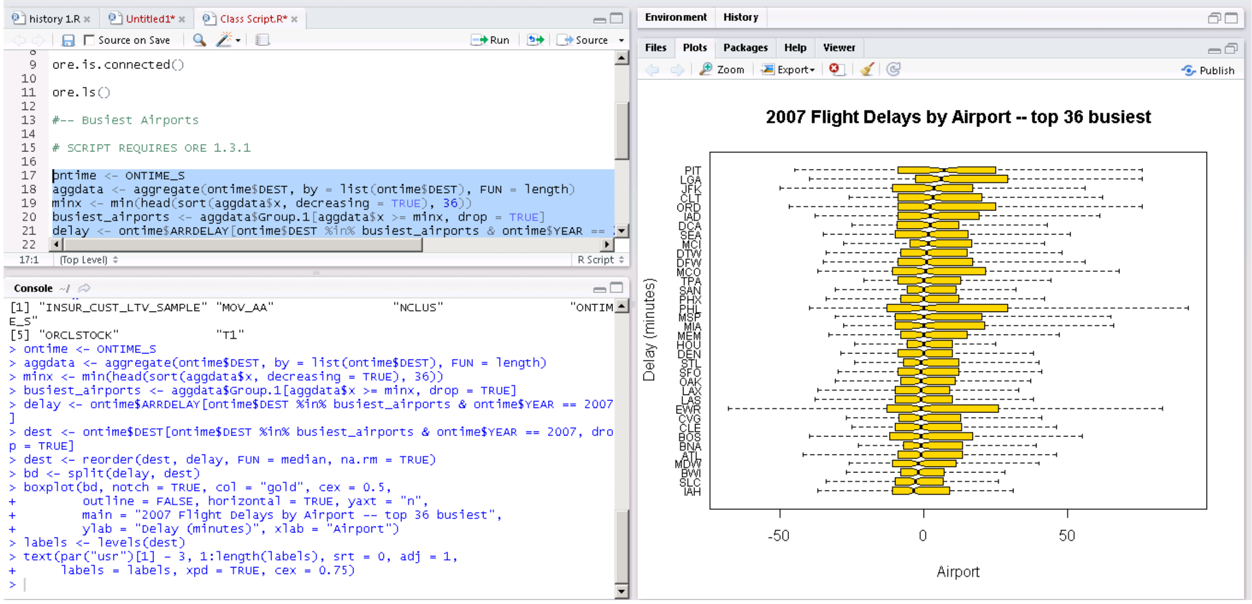
```
> Sys.setenv(TZ = "GMT")
> ore.connect(user = "rquser", service_name = "pdborcl", host = "52.8.149.57", password = "rquser", all=TRUE)
Warning message:
table "RQUSER"."AIRPORTSCOUNT" contains unsupported data types
> 
```

Run the **ore.is.connected()** code to confirm that your connection is live and then the **ore.ls()** code to see what objects exist in the rquser schema in the database.



```
table "RQUSER"."AIRPORTSCOUNT" contains unsupported data types
> ore.is.connected()
[1] TRUE
> ore.ls()
[1] "INSUR_CUST_LTV_SAMPLE" "MOV_AA"                "NCLUS"                 "ONTIME_S"
[5] "ORCLSTOCK"             "T1"
> 
```

Next we will run a script that finds the top 36 busiest airports in the country and creates a box plot graph showing their distribution of their delays for the year 2007. Take a minute to read through the script. Even though you may not be an R expert, you should be able to follow the script. Highlight the entire script and click "Run".



```
ore.is.connected()

ore.ls()

#-- Busiest Airports

# SCRIPT REQUIRES ORE 1.3.1

ontime <- ONTIME_S
aggdata <- aggregate(ontime$DEST, by = list(ontime$DEST), FUN = length)
minx <- min(head(sort(aggdata$x, decreasing = TRUE), 36))
busiest_airports <- aggdata$Group.1[aggdata$x >= minx, drop = TRUE]
delay <- ontime$ARRDELAY[ontime$DEST %in% busiest_airports & ontime$YEAR == :
```

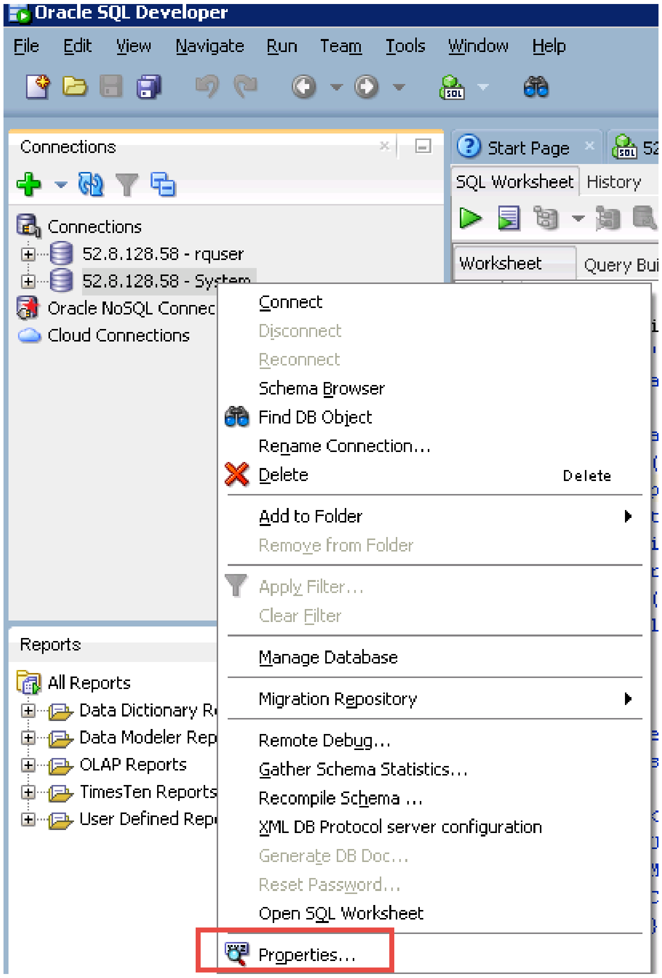
```
[1] "INSUR_CUST_LTV_SAMPLE" "MOV_AA"                "NCLUS"                 "ONTIM
E_S"
[5] "ORCLSTOCK"             "T1"
> ontime <- ONTIME_S
> aggdata <- aggregate(ontime$DEST, by = list(ontime$DEST), FUN = length)
> minx <- min(head(sort(aggdata$x, decreasing = TRUE), 36))
> busiest_airports <- aggdata$Group.1[aggdata$x >= minx, drop = TRUE]
> delay <- ontime$ARRDELAY[ontime$DEST %in% busiest_airports & ontime$YEAR == 2007
]
> dest <- ontime$DEST[ontime$DEST %in% busiest_airports & ontime$YEAR == 2007, dro
p = TRUE]
> dest <- reorder(dest, delay, FUN = median, na.rm = TRUE)
> bd <- split(delay, dest)
> boxplot(bd, notch = TRUE, col = "gold", cex = 0.5,
+         outline = FALSE, horizontal = TRUE, yaxt = "n",
+         main = "2007 Flight Delays by Airport -- top 36 busiest",
+         ylab = "Delay (minutes)", xlab = "Airport")
> labels <- levels(dest)
> text(par("usr")[1] - 3, 1:length(labels), srt = 0, adj = 1,
+      labels = labels, xpd = TRUE, cex = 0.75)
> 
```

Your screen should look something like the above screen shot. Drag the borders of the plot window to make it larger and to make the graph bigger. We will register our script in the Rscript repository in the database so that Oracle database can automatically spin up an R engine in its database server memory, pull the data from the ONTIME_S table, and create the box plot graph image.

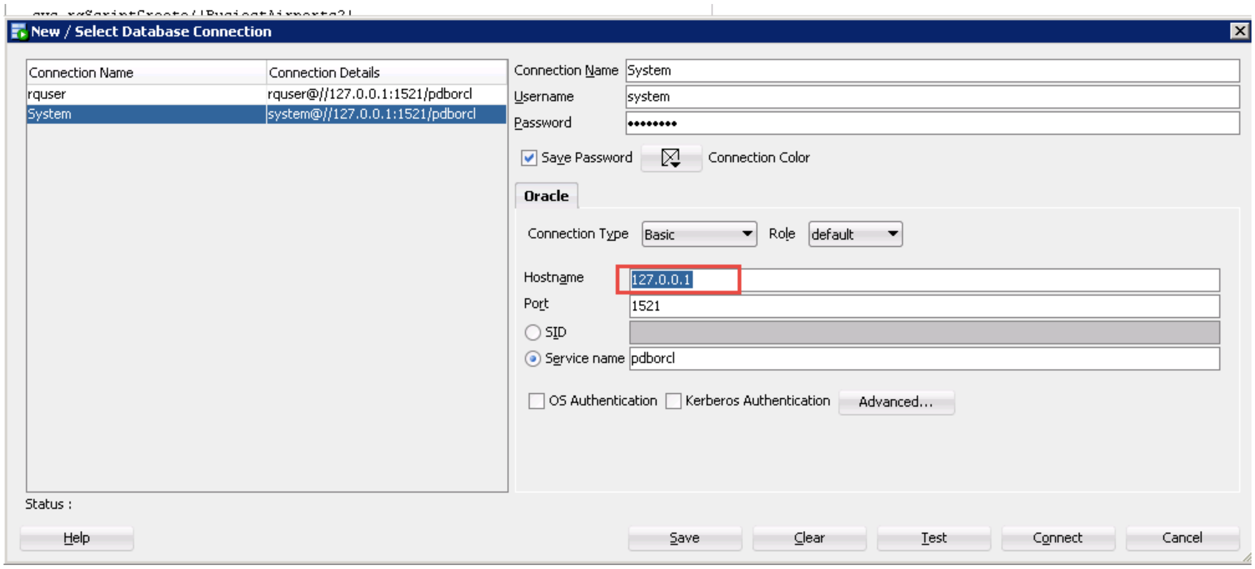
Exit RStudio. (Do NOT save your workspace. Because we have R running on our local client machine and because it's connected to a remote database server, it's generally recommended that you don't save your workspace. The database is likely to change next time you connect and you don't want to load old information from a previous session into R's memory, so it's typically best to establish a fresh connection every time.)

Next we will register our script in the database using SQL Developer. Find the SQL Developer icon

on your desktop and launch it. You should see an **rquser** and a **System** connection on your connection pane. Right-click the **System** connection and select **Properties**.



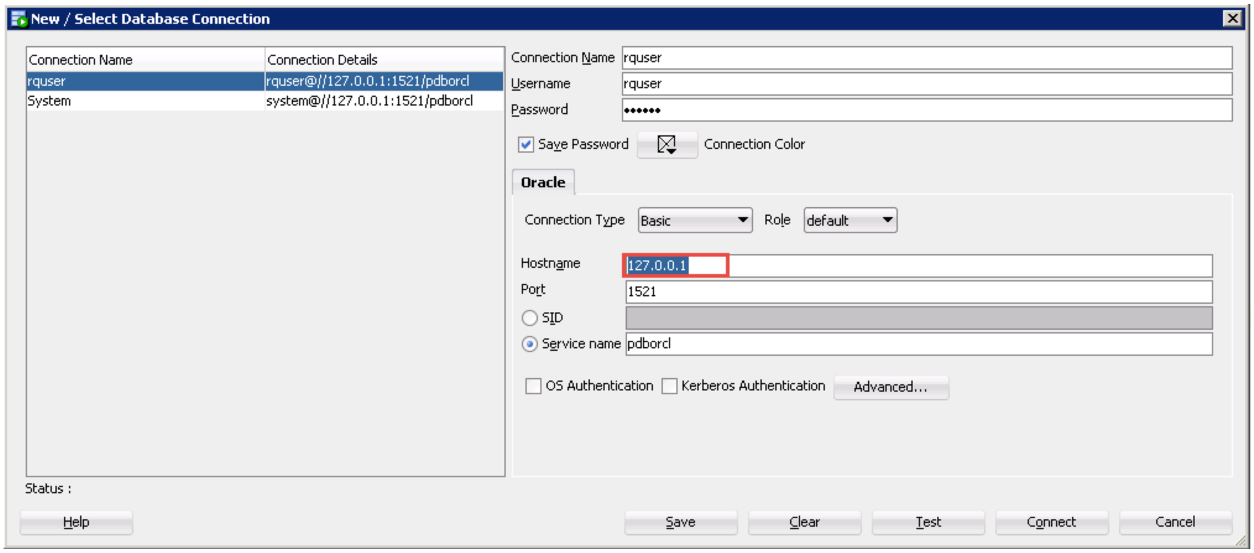
Replace the IP address in the **Hostname** field with the IP address for your Database Server VM. Click, **Test** then **Save**, then **Connect**.



Connection Name	Connection Details
rquser	rquser@//127.0.0.1:1521/pdborcl
System	system@//127.0.0.1:1521/pdborcl

Connection Name: System
Username: system
Password: ••••••••
☑ Save Password
Connection Type: Basic
Role: default
Hostname: 127.0.0.1
Port: 1521
○ SID
● Service name: pdborcl
□ OS Authentication □ Kerberos Authentication

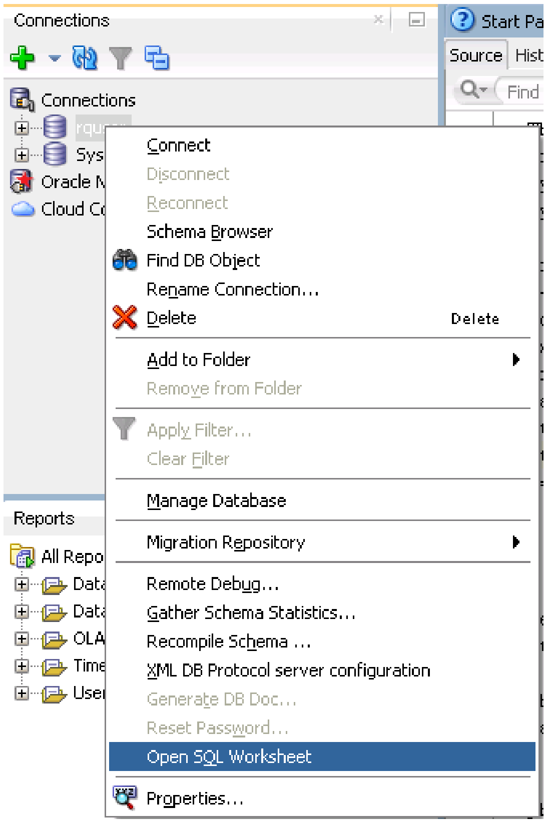
Right-click the **rquser** connection in the **Connections** pane and follow the same steps. Replace the **Hostname** field with your database server VM IP, click **Test**, **Save**, and **Connect**.



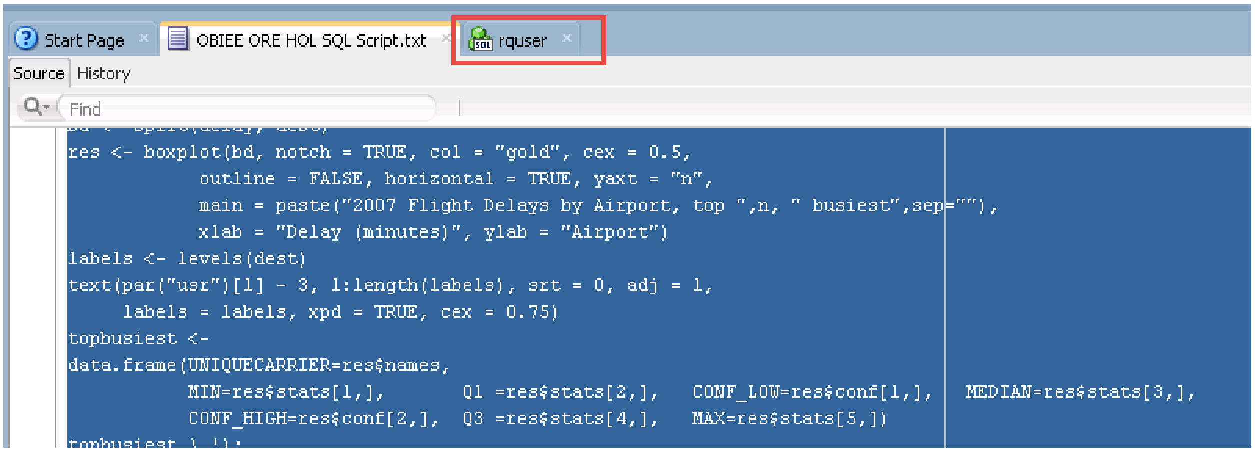
Connection Name	Connection Details
rquser	rquser@//127.0.0.1:1521/pdborcl
System	system@//127.0.0.1:1521/pdborcl

Connection Name: rquser
Username: rquser
Password: ••••••
☑ Save Password
Connection Type: Basic
Role: default
Hostname: 127.0.0.1
Port: 1521
○ SID
● Service name: pdborcl
□ OS Authentication □ Kerberos Authentication

Right-click the **rquser** connection and select **Open SQL Worksheet**.

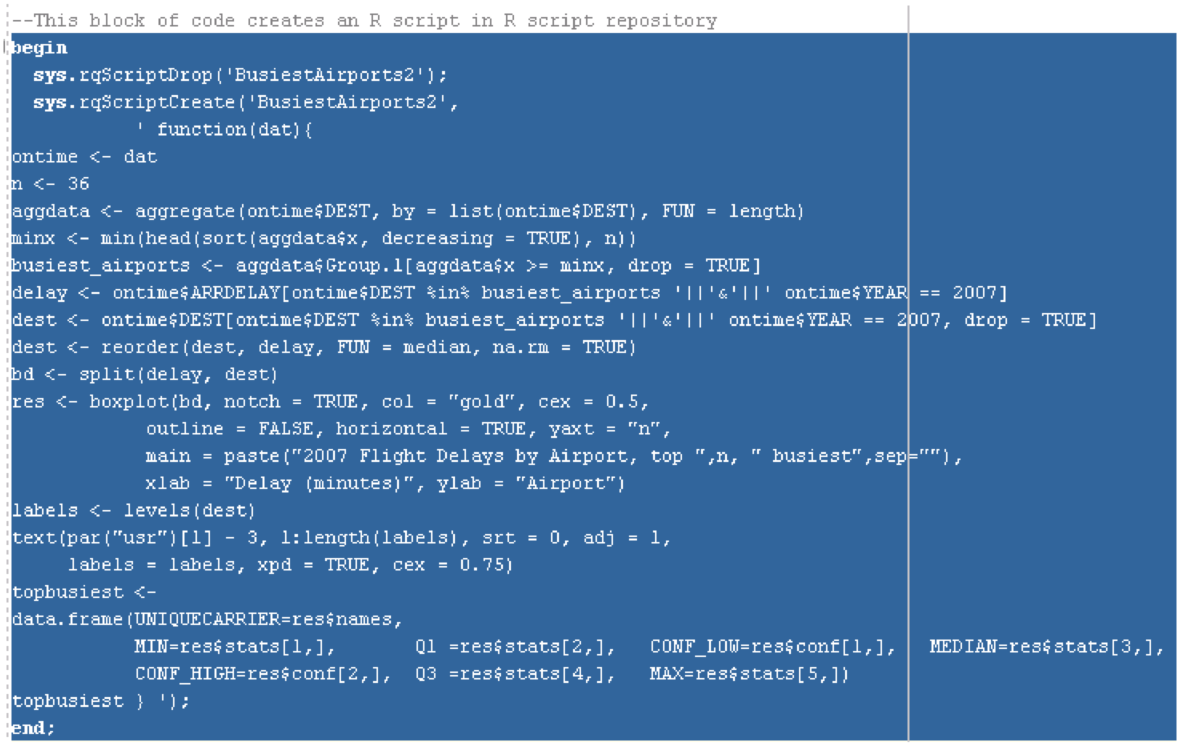


Click on the **OBIEE ORE SQL Script.txt** tab or navigate to the .txt file on the desktop. Highlight the entire script (cntrl A), copy it (cntrl C), and then paste it onto the rquser SQL Worksheet (cntrl V).



```
res <- boxplot(bd, notch = TRUE, col = "gold", cex = 0.5,
               outline = FALSE, horizontal = TRUE, yaxt = "n",
               main = paste("2007 Flight Delays by Airport, top ",n, " busiest",sep=""),
               xlab = "Delay (minutes)", ylab = "Airport")
labels <- levels(dest)
text(par("usr")[1] - 3, 1:length(labels), srt = 0, adj = 1,
     labels = labels, xpd = TRUE, cex = 0.75)
topbusiest <-
data.frame(UNIQUECARRIER=res$names,
           MIN=res$stats[1,],       Q1 =res$stats[2,],   CONF_LOW=res$conf[1,],   MEDIAN=res$stats[3,],
           CONF_HIGH=res$conf[2,],  Q3 =res$stats[4,],   MAX=res$stats[5,])
```

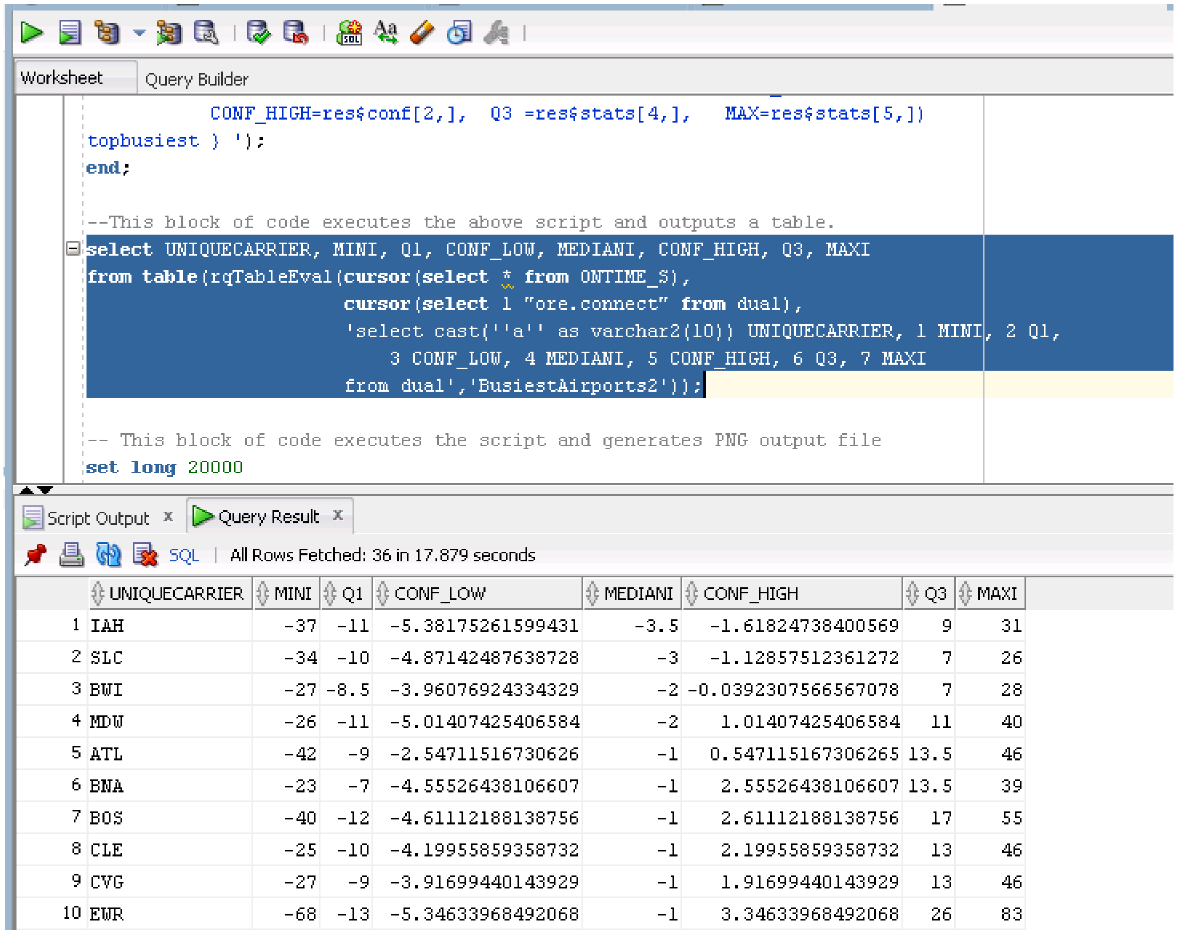
Highlight the top block of the script and run it. Take a moment to read through the script. Notice that while it is written in R, there are some minor syntax modifications to accommodate SQL's special character syntax. We now have registered our script under the name "BusiestAirports2" in the R script repository in the database. We could have done this from the RStudio interface as well, but chose to do it in SQL Developer as many DBAs like to have System schema scripts registered in PL/SQL process (the R Script Repository uses the SYSTEM schema.)



```
--This block of code creates an R script in R script repository
begin
  sys.rqScriptDrop('BusiestAirports2');
  sys.rqScriptCreate('BusiestAirports2',
           ' function(dat){
ontime <- dat
n <- 36
aggdata <- aggregate(ontime$DEST, by = list(ontime$DEST), FUN = length)
minx <- min(head(sort(aggdata$x, decreasing = TRUE), n))
busiest_airports <- aggdata$Group.1[aggdata$x >= minx, drop = TRUE]
delay <- ontime$ARRDELAY[ontime$DEST %in% busiest_airports '||'&'||' ontime$YEAR == 2007]
dest <- ontime$DEST[ontime$DEST %in% busiest_airports '||'&'||' ontime$YEAR == 2007, drop = TRUE]
dest <- reorder(dest, delay, FUN = median, na.rm = TRUE)
bd <- split(delay, dest)
res <- boxplot(bd, notch = TRUE, col = "gold", cex = 0.5,
            outline = FALSE, horizontal = TRUE, yaxt = "n",
            main = paste("2007 Flight Delays by Airport, top ",n, " busiest",sep=""),
            xlab = "Delay (minutes)", ylab = "Airport")
labels <- levels(dest)
text(par("usr")[1] - 3, 1:length(labels), srt = 0, adj = 1,
     labels = labels, xpd = TRUE, cex = 0.75)
topbusiest <-
data.frame(UNIQUECARRIER=res$names,
           MIN=res$stats[1,],       Q1 =res$stats[2,],   CONF_LOW=res$conf[1,],   MEDIAN=res$stats[3,],
           CONF_HIGH=res$conf[2,],  Q3 =res$stats[4,],   MAX=res$stats[5,])
topbusiest } ');
end;
```

Next we will execute a SQL script to create a table output that we can place on our OBIEE dashboard to accompany our Box Plot graph.

Highlight the next block of code on the **rquser** SQL worksheet and run it by clicking on the **green arrow**. Take a moment to read through the script and notice how we are using a combination of cursor select statements, column definitions, and the above R script that we just registered. Understanding the exact syntax of the rqTableEval function is beyond the scope of this hands-on lab, but you should be able to follow what is happening.



```
             CONF_HIGH=res$conf[2,],  Q3 =res$stats[4,],   MAX=res$stats[5,])
topbusiest } ');
end;

--This block of code executes the above script and outputs a table.
select UNIQUECARRIER, MINI, Q1, CONF_LOW, MEDIANI, CONF_HIGH, Q3, MAXI
from table(rqTableEval(cursor(select * from ONTIME_S),
                       cursor(select 1 "ore.connect" from dual),
                       'select cast(''a'' as varchar2(10)) UNIQUECARRIER, 1 MINI, 2 Q1,
                           3 CONF_LOW, 4 MEDIANI, 5 CONF_HIGH, 6 Q3, 7 MAXI
                       from dual','BusiestAirports2'));

-- This block of code executes the script and generates PNG output file
set long 20000
```

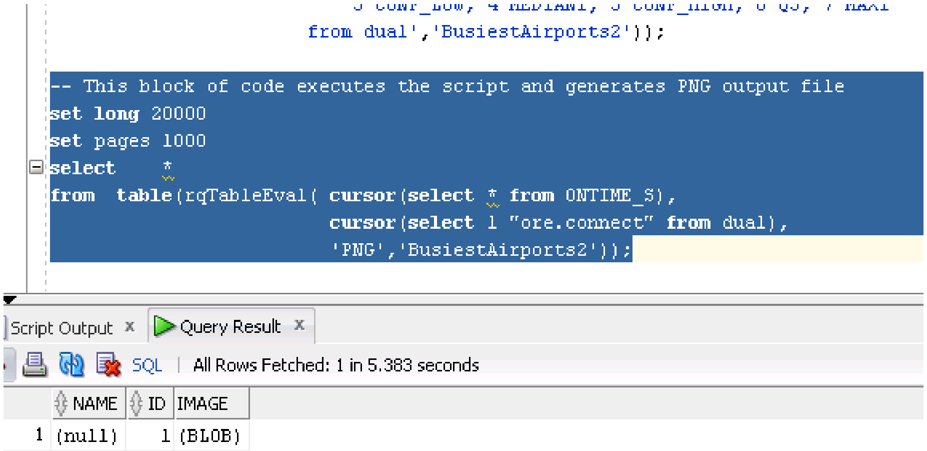
All Rows Fetched: 36 in 17.879 seconds

	UNIQUECARRIER	MINI	Q1	CONF_LOW	MEDIANI	CONF_HIGH	Q3	MAXI
1	IAH	-37	-11	-5.38175261599431	-3.5	-1.61824738400569	9	31
2	SLC	-34	-10	-4.87142487638728	-3	-1.12857512361272	7	26
3	BWI	-27	-8.5	-3.96076924334329	-2	-0.0392307566567078	7	28
4	MDW	-26	-11	-5.01407425406584	-2	1.01407425406584	11	40
5	ATL	-42	-9	-2.54711516730626	-1	0.547115167306265	13.5	46
6	BNA	-23	-7	-4.55526438106607	-1	2.55526438106607	13.5	39
7	BOS	-40	-12	-4.61112188138756	-1	2.61112188138756	17	55
8	CLE	-25	-10	-4.19955859358732	-1	2.19955859358732	13	46
9	CVG	-27	-9	-3.91699440143929	-1	1.91699440143929	13	46
10	EWR	-68	-13	-5.34633968492068	-1	3.34633968492068	26	83

Notice that we have all the information that is included in our graph, the airport code, minimum delay, in the interquartile range (Q1 and Q3), and the low and high confidence intervals, and the maximum delay.

Next we'll create our graph and save it as a PNG BLOB image that can be published on an OBIEE dashboard.

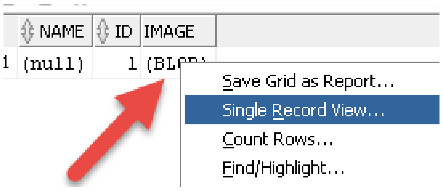
Highlight the next block of code and run it.



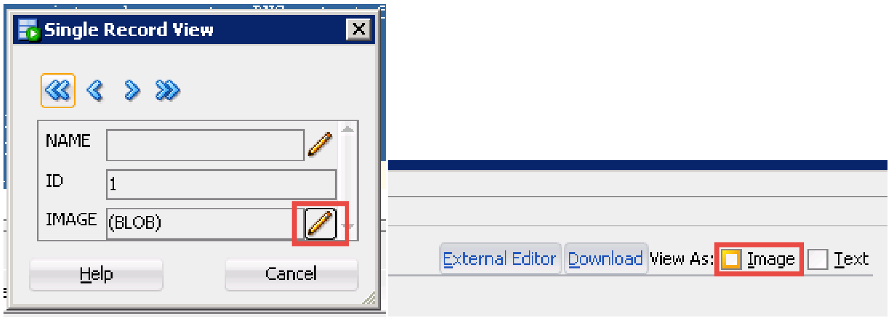
```
                         5 CONF_LOW, 4 MEDIAN1, 5 CONF_HIGH, 6 Q3, 7 MAX1
                     from dual','BusiestAirports2'));

-- This block of code executes the script and generates PNG output file
set long 20000
set pages 1000
select    *
from  table(rqTableEval( cursor(select * from ONTIME_S),
                         cursor(select 1 "ore.connect" from dual),
                         'PNG','BusiestAirports2'));
```

	NAME	ID	IMAGE
1	(null)	1	(BLOB)

SQL Developer now has the ability to preview images. Right-click on the **(BLOB)** field and select **Single Record View**.



Then click on the **pencil** icon and then click the **Image** box.

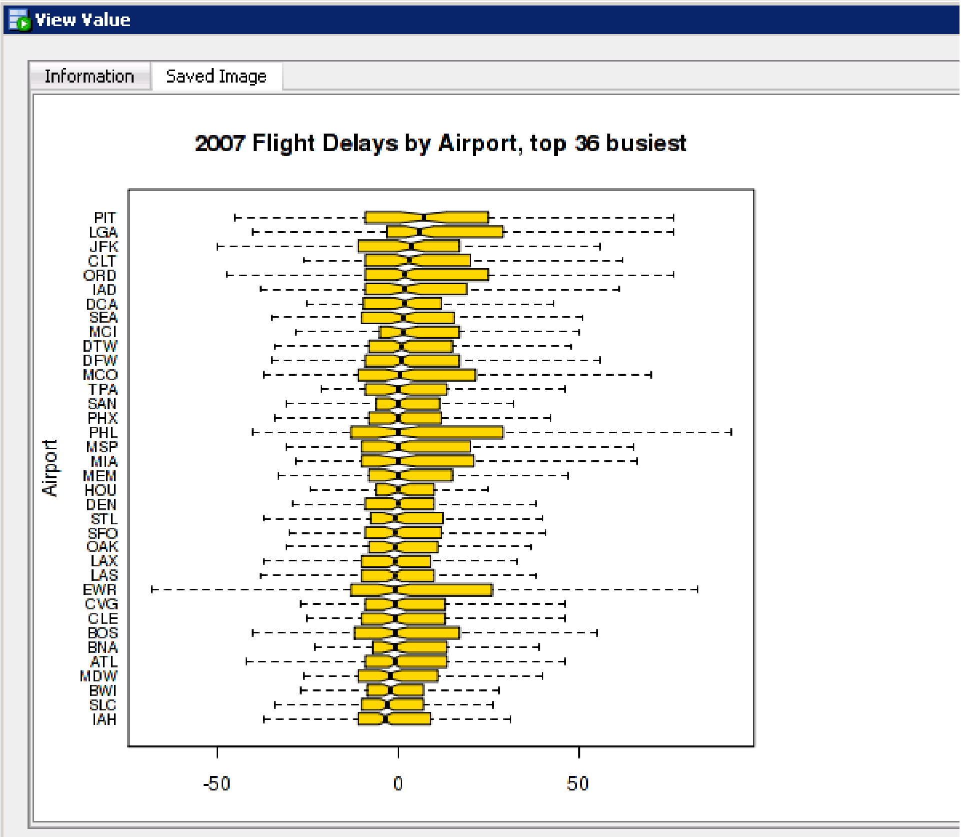


NAME:

ID: 1

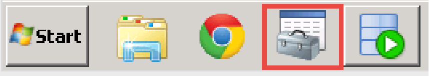
IMAGE: (BLOB)

You should see our box plot graph!

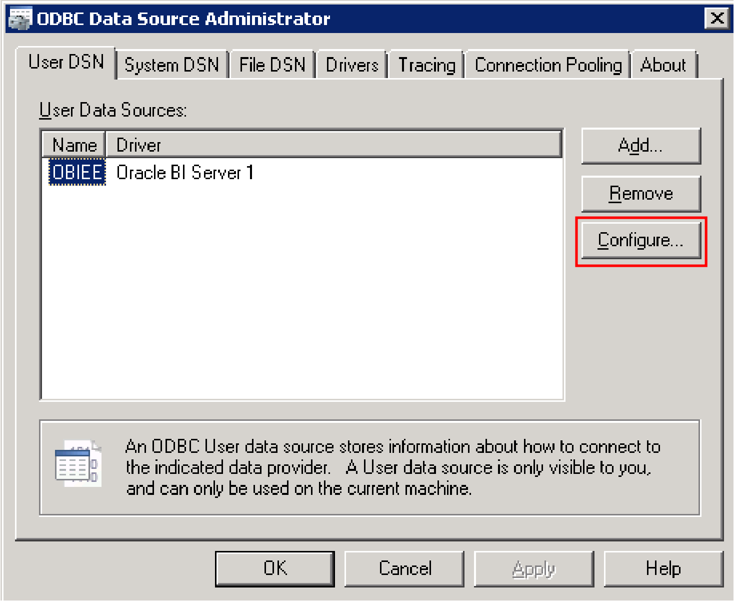


Next we will establish a connection with the BI Server and modify the repository .rpd file.

Find the Data Sources ODBC connections icon at the bottom of your desktop and open it.



Click **Configure** to open the OBIEE connection.

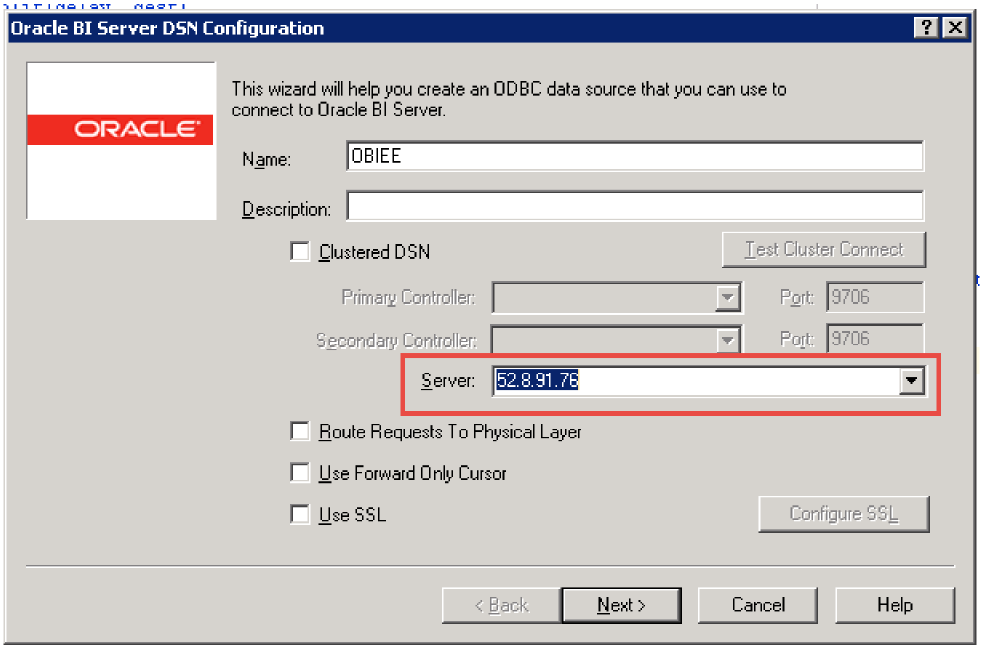


User Data Sources:

Name	Driver
OBIEE	Oracle BI Server 1

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.

Next, enter your Server VM IP in the **Server** field and click **Next**.



This wizard will help you create an ODBC data source that you can use to connect to Oracle BI Server.

Name: OBIEE

Description:

□ Clustered DSN

Primary Controller: Port: 9706

Secondary Controller: Port: 9706

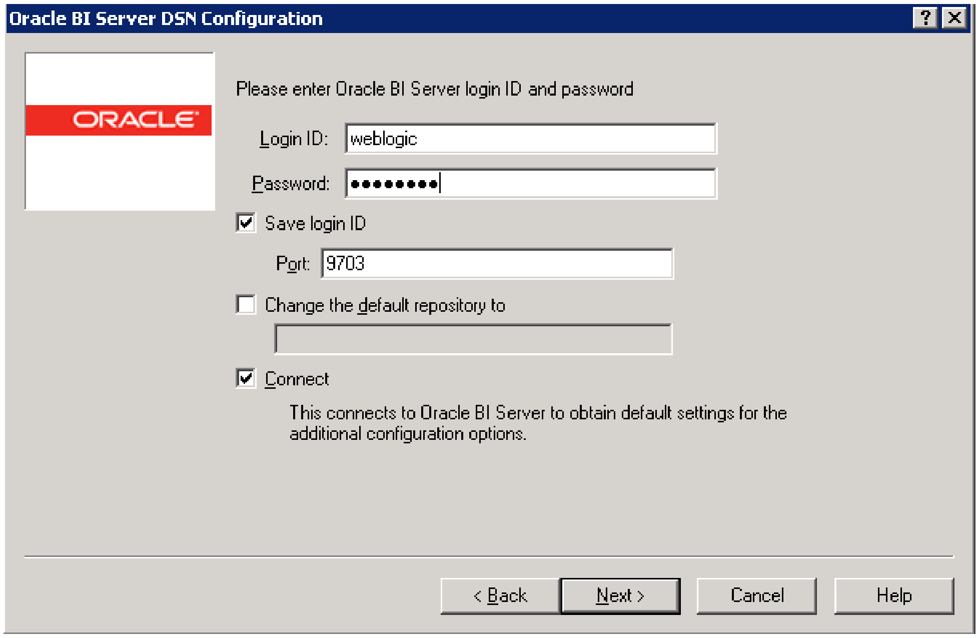
Server: 52.8.91.76

□ Route Requests To Physical Layer

□ Use Forward Only Cursor

□ Use SSL

Enter **Admin123** in the password field and click next.



Please enter Oracle BI Server login ID and password

Login ID: weblogic

Password: ●●●●●●●●

☑ Save login ID

Port: 9703

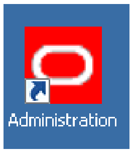
□ Change the default repository to

☑ Connect

This connects to Oracle BI Server to obtain default settings for the additional configuration options.

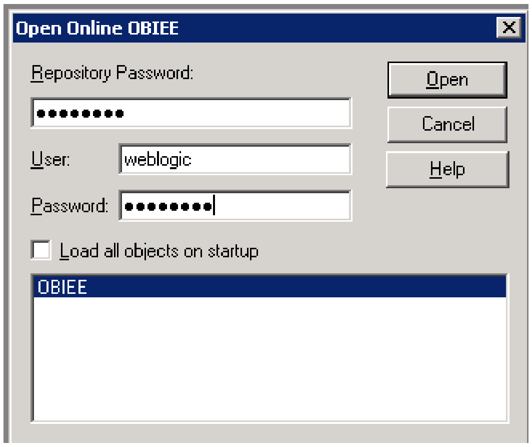
It may take a few minutes to establish a connection, so please be patient. Click **Finish** and close the configuration interface.

Open the BI Admin tool by double-clicking the red Oracle "O" square labeled **Administration** on the desktop.



Click **File**, then **Open**, then select **Online**. In real world settings, we recommend making all the changes in the **Offline** mode, but in the interests of time and avoiding stopping and restarting services, we're going to do everything in online mode today. Remember, you have your own individual client and your own individual server with no one else on it. Adding and editing physical sources in the repository is not recommended for most all other situations.

Enter **Admin123** as the password for in the **Repository Password** field and in the **Password** field and click **Open**.


Open Online OBIEE

Repository Password: ••••••••

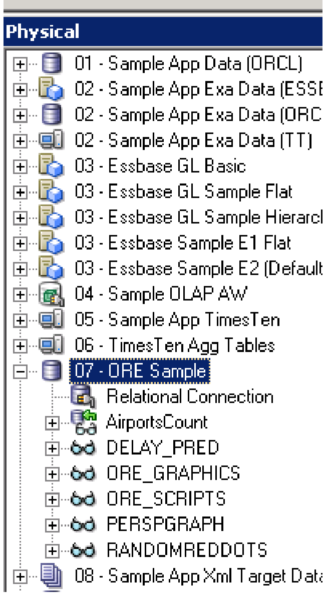
User: weblogic

Password: ••••••••

□ Load all objects on startup

OBIEE

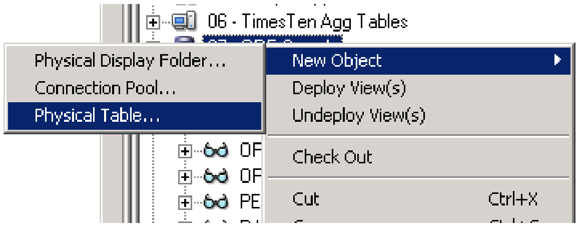
In the **Physical** layer panel, click on the **07 – ORE Sample** data source node to expand it.


Physical

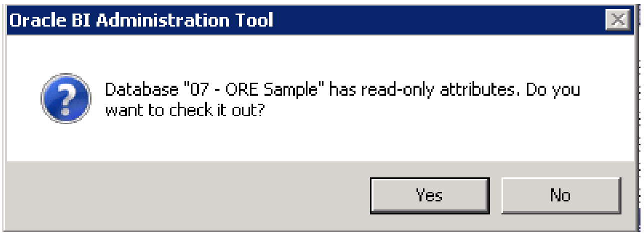
- 01 - Sample App Data (ORCL)
- 02 - Sample App Exa Data (ESSE
- 02 - Sample App Exa Data (ORC
- 02 - Sample App Exa Data (TT)
- 03 - Essbase GL Basic
- 03 - Essbase GL Sample Flat
- 03 - Essbase GL Sample Hierarcl
- 03 - Essbase Sample E1 Flat
- 03 - Essbase Sample E2 (Default
- 04 - Sample OLAP AW
- 05 - Sample App TimesTen
- 06 - TimesTen Agg Tables
- 07 - ORE Sample
  - Relational Connection
  - AirportsCount
  - DELAY_PRED
  - ORE_GRAPHICS
  - ORE_SCRIPTS
  - PERSPGRAPH
  - RANDOMREDDOTS
- 08 - Sample App Xml Target Data

We will now add a physical table to hold the table that we created in SQL Developer with the data fields that are used to create our Box Plot graph. This demonstrates how to import structured data from an embedded R process that is run in the database into OBIEE.

Right-click the **07 – ORE Sample** node and select **New Object** and then **Physical Table**.



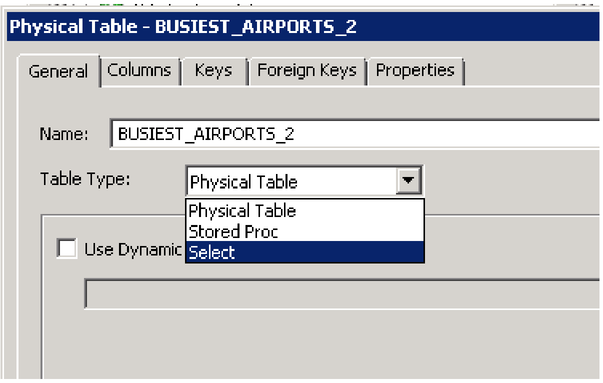
Click **Yes** you want to check it out.



Oracle BI Administration Tool

Database "07 - ORE Sample" has read-only attributes. Do you want to check it out?

Name your table **BUSIEST_AIRPORTS_2**. Choose **Select** in the **Table Type** drop down box.



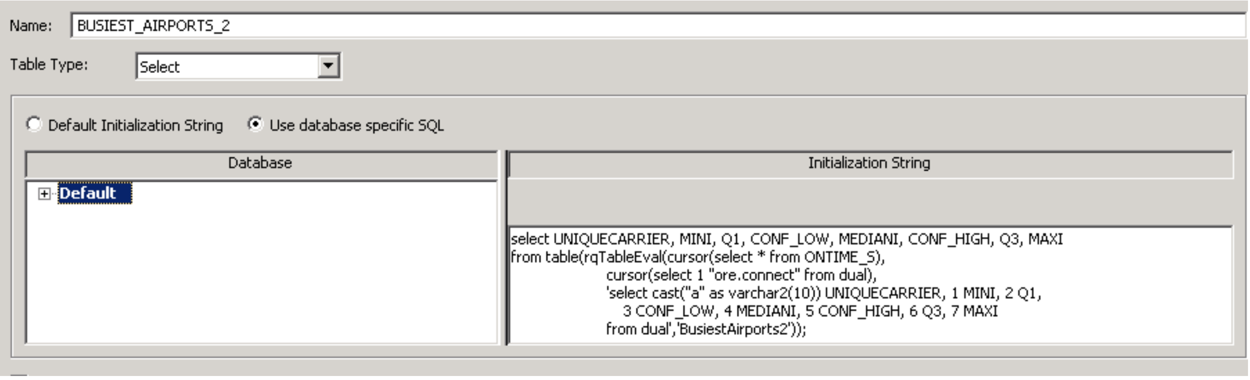
Physical Table - BUSIEST_AIRPORTS_2

Name: BUSIEST_AIRPORTS_2

Table Type: Physical Table

Click the **Use database specific SQL** radio button and paste the following code into the **Initialization String** field as shown.

```
select UNIQUECARRIER, MINI, Q1, CONF_LOW, MEDIANI, CONF_HIGH, Q3, MAXI
from table(rqTableEval(cursor(select * from ONTIME_S),
                    cursor(select 1 "ore.connect" from dual),
                    'select cast("a" as varchar2(10)) UNIQUECARRIER, 1 MINI, 2 Q1,
                       3 CONF_LOW, 4 MEDIANI, 5 CONF_HIGH, 6 Q3, 7 MAXI
                    from dual','BusiestAirports2'));
```



Name: BUSIEST_AIRPORTS_2

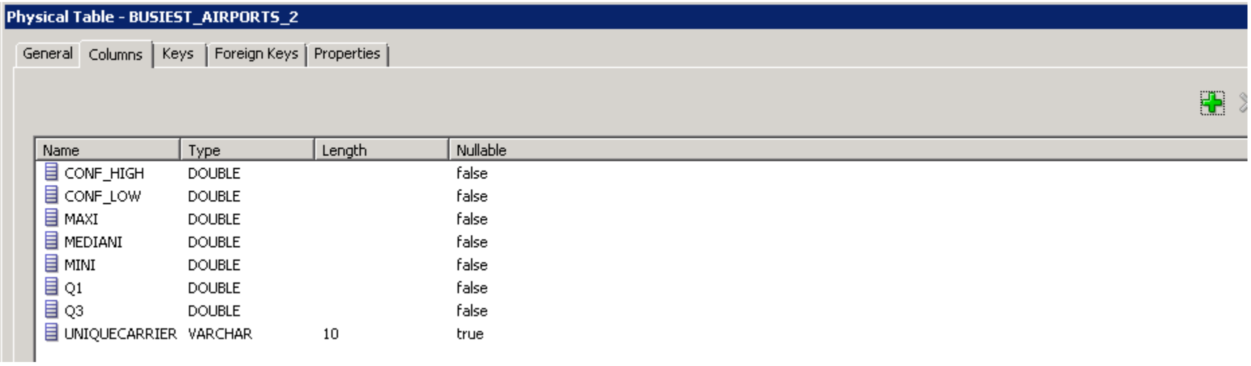
Table Type: Select

○ Default Initialization String ● Use database specific SQL

Database	Initialization String
⊞ Default	

```
select UNIQUECARRIER, MINI, Q1, CONF_LOW, MEDIANI, CONF_HIGH, Q3, MAXI
from table(rqTableEval(cursor(select * from ONTIME_S),
               cursor(select 1 "ore.connect" from dual),
               'select cast("a" as varchar2(10)) UNIQUECARRIER, 1 MINI, 2 Q1,
                  3 CONF_LOW, 4 MEDIANI, 5 CONF_HIGH, 6 Q3, 7 MAXI
               from dual','BusiestAirports2'));
```

Go to the Columns tab and click the Green plus sign to define the columns as shown below.

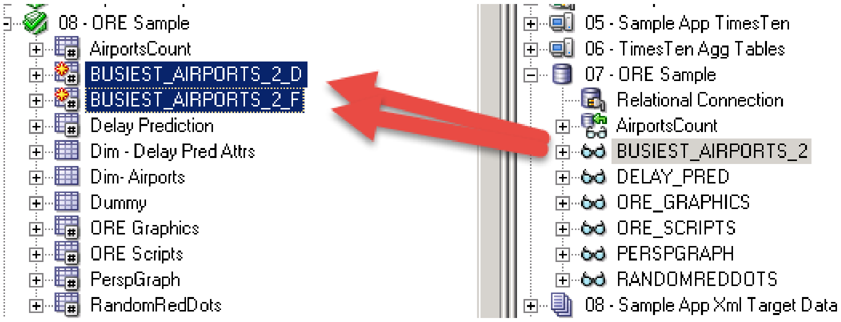


Physical Table - BUSIEST_AIRPORTS_2

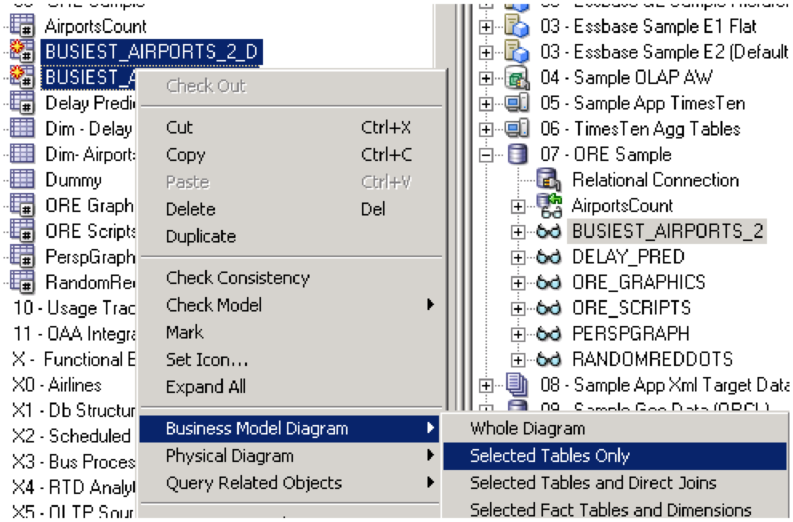
Name	Type	Length	Nullable
CONF_HIGH	DOUBLE		false
CONF_LOW	DOUBLE		false
MAXI	DOUBLE		false
MEDIANI	DOUBLE		false
MINI	DOUBLE		false
Q1	DOUBLE		false
Q3	DOUBLE		false
UNIQUECARRIER	VARCHAR	10	true

Go ahead and check in your changes by clicking on the **check in changes** icon in the menu bar. Do NOT check global consistency.

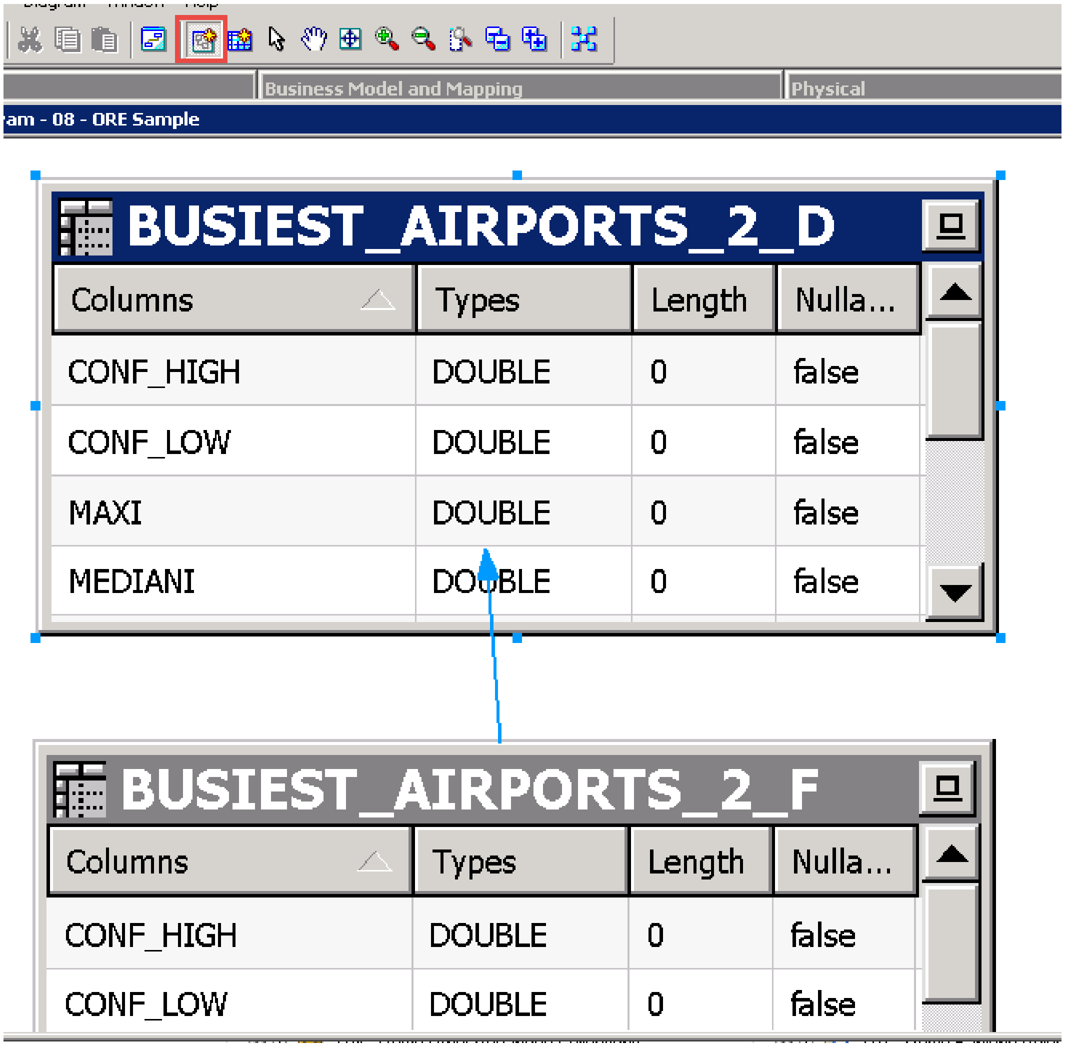
Drag your BUSIEST_AIRPORTS_2 table from the Physical layer to the 08 – ORE Sample node in the Business Model and Mapping layer twice. Name one BUSIEST_AIRPORTS_2_F and the other BUSIEST_AIRPORTS_2_D.



Highlight both tables in the business model logical layer then right-click and select Business Model Diagram and then Selected Tables Only.



Click on the **Join** tool in the tool bar and then click and drag to join the two tables as shown. Then click **OK** and close the business layer diagram.

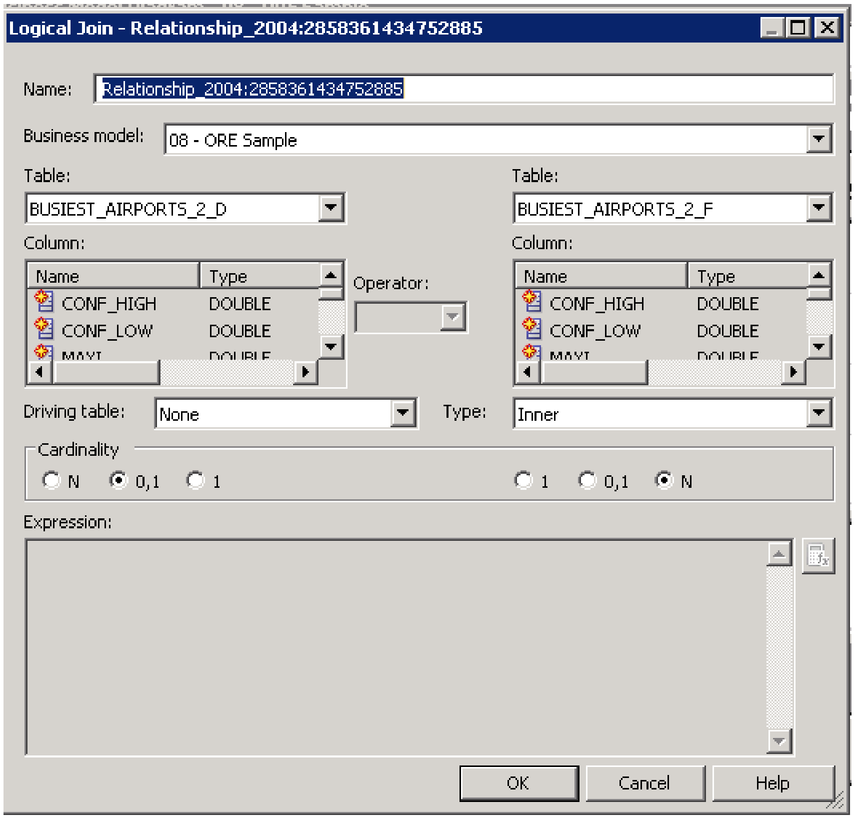


BUSIEST_AIRPORTS_2_D

Columns	Types	Length	Nulla...
CONF_HIGH	DOUBLE	0	false
CONF_LOW	DOUBLE	0	false
MAXI	DOUBLE	0	false
MEDIANI	DOUBLE	0	false

BUSIEST_AIRPORTS_2_F

Columns	Types	Length	Nulla...
CONF_HIGH	DOUBLE	0	false
CONF_LOW	DOUBLE	0	false


Logical Join - Relationship_2004:2858361434752885

Name: Relationship_2004:2858361434752885

Business model: 08 - ORE Sample

Table:	Table:
BUSIEST_AIRPORTS_2_D	BUSIEST_AIRPORTS_2_F

Column:

Name	Type
CONF_HIGH	DOUBLE
CONF_LOW	DOUBLE
MAXI	DOUBLE

Operator:

Column:

Name	Type
CONF_HIGH	DOUBLE
CONF_LOW	DOUBLE
MAXI	DOUBLE

Driving table: None

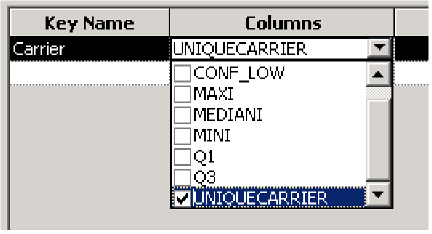
Type: Inner

Cardinality

○ N ● 0,1 ○ 1 ○ 1 ○ 0,1 ● N

Expression:

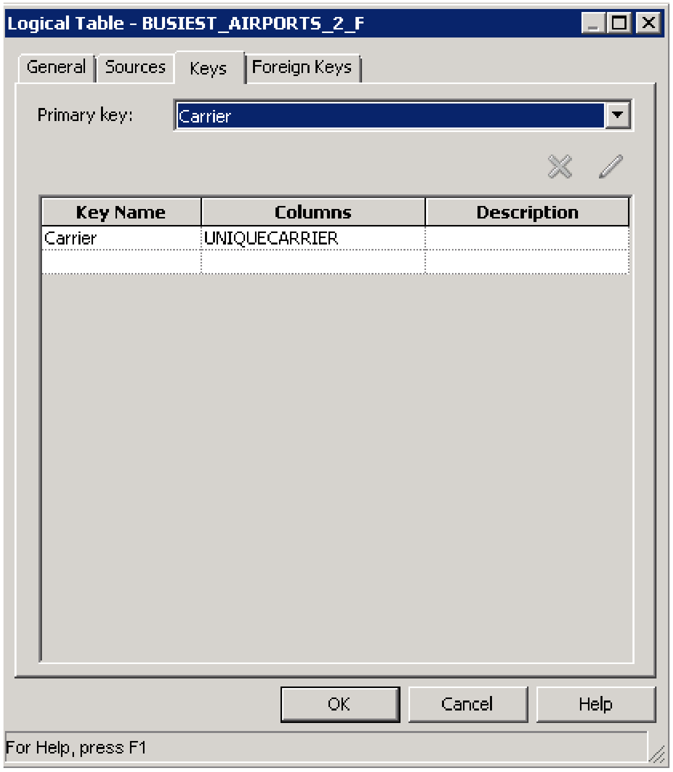
Open the **BUSIEST_AIRPORTS_2_F** table by double-clicking it. Then click on the **Keys** tab. Type **Carrier** in the **Key Name** field and select **UNIQUECARRIER** in the columns field.



Key Name	Columns
Carrier	UNIQUECARRIER

□ CONF_LOW
□ MAXI
□ MEDIANI
□ MINI
□ Q1
□ Q3
☑ UNIQUECARRIER

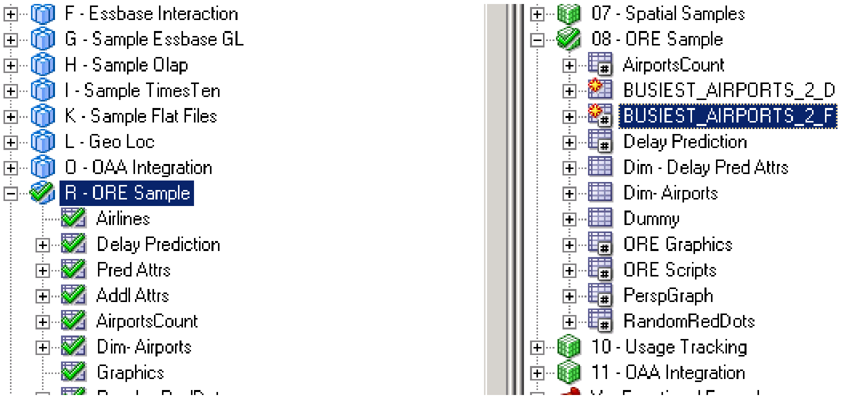
Next select **Carrier** in the **Primary Key** dropdown as shown and click **OK**. Repeat the procedure for the BUSIEST_AIRPORT_2_D table.



Primary key: Carrier

Key Name	Columns	Description
Carrier	UNIQUECARRIER	

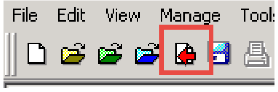
Drag the BUSIEST_AIRPORTS_2_F table to the R – ORE SAMPLE subject area node in the presentation layer as shown.



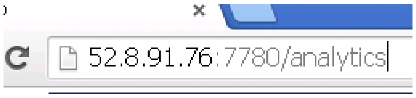
- F - Essbase Interaction
- G - Sample Essbase GL
- H - Sample Olap
- I - Sample TimesTen
- K - Sample Flat Files
- L - Geo Loc
- O - OAA Integration
- R - ORE Sample
  - Airlines
  - Delay Prediction
  - Pred Attrs
  - Addl Attrs
  - AirportsCount
  - Dim- Airports
  - Graphics

- 07 - Spatial Samples
- 08 - ORE Sample
  - AirportsCount
  - BUSIEST_AIRPORTS_2_D
  - BUSIEST_AIRPORTS_2_F
  - Delay Prediction
  - Dim - Delay Pred Attrs
  - Dim- Airports
  - Dummy
  - ORE Graphics
  - ORE Scripts
  - PerspGraph
  - RandomRedDots
- 10 - Usage Tracking
- 11 - OAA Integration

Click the Check in Changes icon on the tool bar and check global consistency. (You should have 0 errors and 11 warnings.)



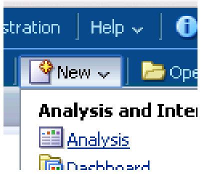
Open a browser window and type the IP address for your Server VM followed by **:7780/analytics** as shown



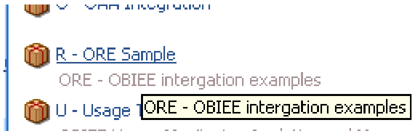
52.8.91.76:7780/analytics

Log in as Prodney/Admin123.

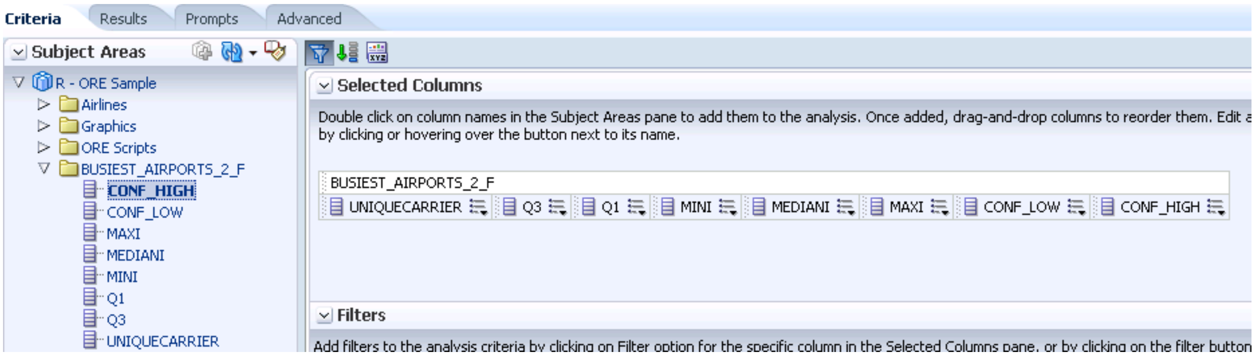
Click on the **New** icon in the menu bar and select **Analysis**.



Select the **R – ORE Sample** subject area.


R - ORE Sample
ORE - OBIEE intergation examples
U - Usage
ORE - OBIEE intergation examples

You should see your BUSIEST_AIRPORTS_2_F table in the left hand selection pane. Open it and drag all the columns to the Selected Columns pane on the Criteria tab. You Criteria tab should look something like this.

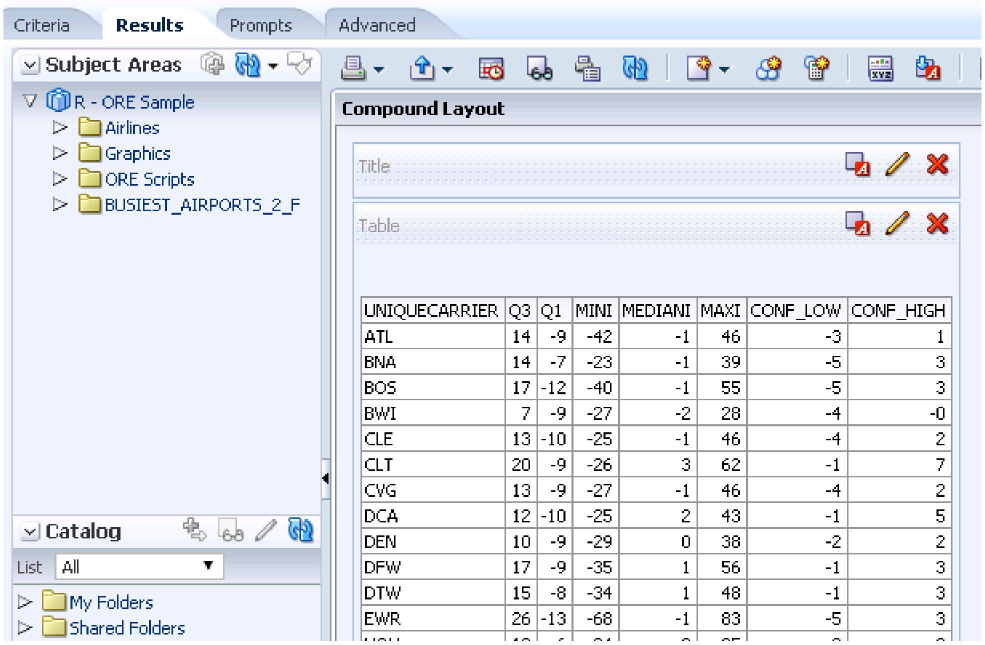


Double click on column names in the Subject Areas pane to add them to the analysis. Once added, drag-and-drop columns to reorder them. Edit a by clicking or hovering over the button next to its name.

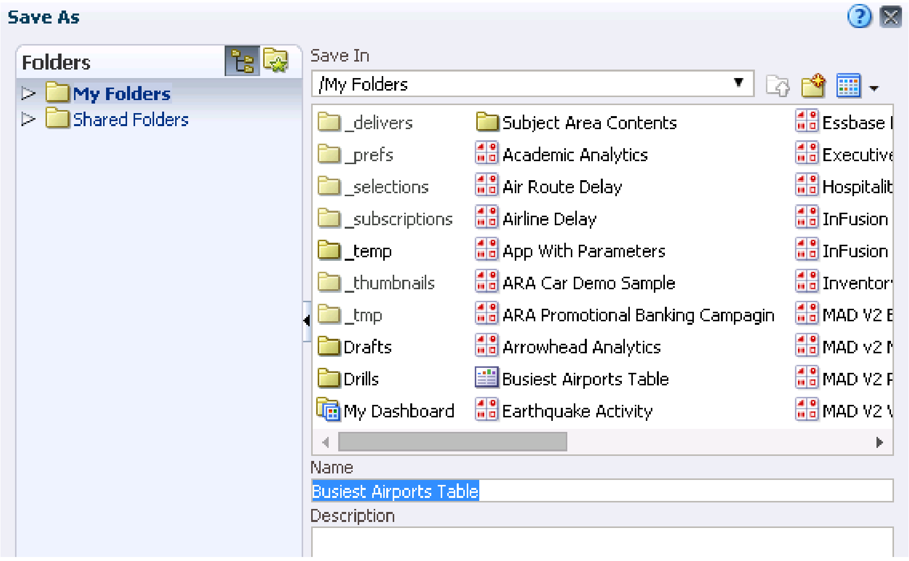
BUSIEST_AIRPORTS_2_F
UNIQUECARRIER | Q3 | Q1 | MINI | MEDIANI | MAXI | CONF_LOW | CONF_HIGH

Add filters to the analysis criteria by clicking on Filter option for the specific column in the Selected Columns pane, or by clicking on the filter button

Click on the Results tab and Voila! You should have a table with the structured results of our R scripts. Save your analysis for use later on a dashboard.



UNIQUECARRIER	Q3	Q1	MINI	MEDIANI	MAXI	CONF_LOW	CONF_HIGH
ATL	14	-9	-42	-1	46	-3	1
BNA	14	-7	-23	-1	39	-5	3
BOS	17	-12	-40	-1	55	-5	3
BWI	7	-9	-27	-2	28	-4	-0
CLE	13	-10	-25	-1	46	-4	2
CLT	20	-9	-26	3	62	-1	7
CVG	13	-9	-27	-1	46	-4	2
DCA	12	-10	-25	2	43	-1	5
DEN	10	-9	-29	0	38	-2	2
DFW	17	-9	-35	1	56	-1	3
DTW	15	-8	-34	1	48	-1	3
EWR	26	-13	-68	-1	83	-5	3



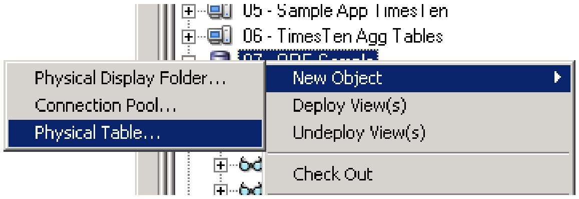
Save In: /My Folders

Name: Busiest Airports Table

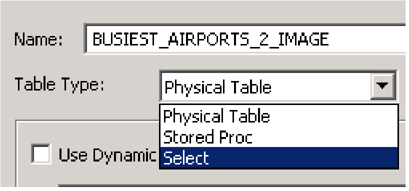
Description

Return to the BI Admin tool so that we can bring in the Box Plot graph image. We will a similar path to create a table in the physical layer, join it to a second copy in the business layer, and promote to the presentation layer.

Right-click the **ORE Sample** node and select **New Object** and then **Physical Table**. Click **Yes** you want to check it out.



Name your new table BUSIEST_AIRPORTS_2_IMAGE. Choose Select in the Table Type field.

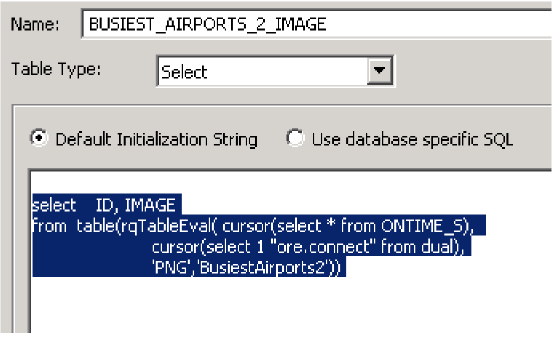


Name: BUSIEST_AIRPORTS_2_IMAGE

Table Type: Physical Table

Click the Default Initialization String button and enter the SQL code below in the field. Take a minute to read the code and understand what is happening. We are specifying and select only two columns, ID and IMAGE. We are using the rqTableEval function to spin up an R engine as an embedded process on the database server machine and to execute our BusiestAirports2 script. We are also specifying the data type for our second column as PNG which is a graphic type that OBIEE can display.

```
select   ID, IMAGE
from  table(rqTableEval( cursor(select * from ONTIME_S),
                        cursor(select 1 "ore.connect" from dual),
                        'PNG','BusiestAirports2'))
```



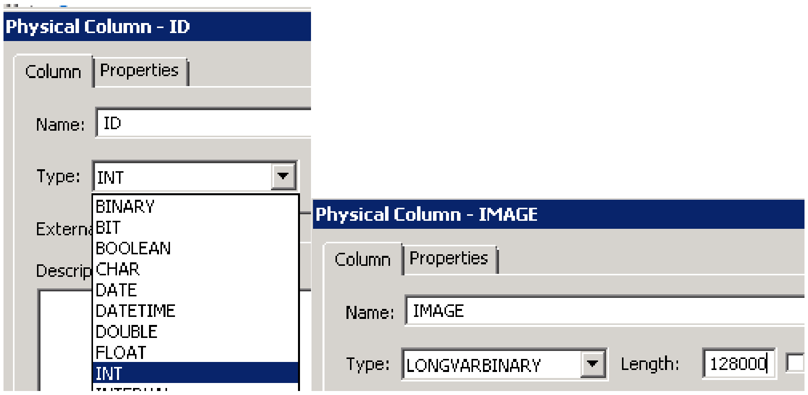
Name: BUSIEST_AIRPORTS_2_IMAGE

Table Type: Select

● Default Initialization String ○ Use database specific SQL

```
select   ID, IMAGE
from  table(rqTableEval( cursor(select * from ONTIME_S),
                        cursor(select 1 "ore.connect" from dual),
                        'PNG','BusiestAirports2'))
```

Click on the **Columns** tab. We will now create the ID and IMAGE columns for our table. Click on the green plus sign and enter ID in the name field and select INT in the Type drop down field. Repeat the process and create a column named IMAGE of Type LONGVARBINARY and Length 128000.



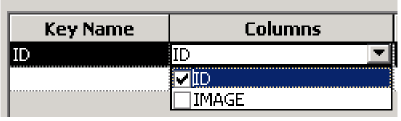
Name: ID

Type: INT

Name: IMAGE

Type: LONGVARBINARY Length: 128000

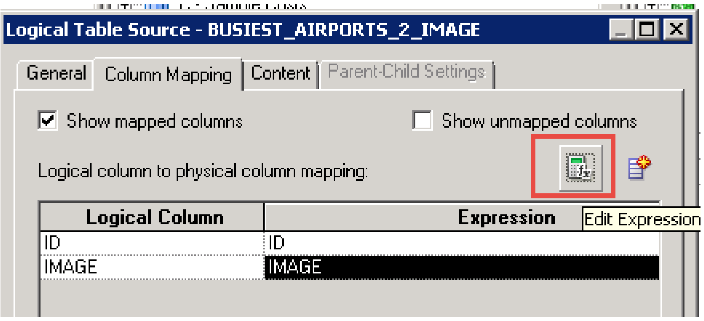
Go to the **Keys** tab and type **ID** in the **Key Name** field and then select **ID** in the column drop down and click **OK**. Check in your changes and do not check global consistency. You should now see a key icon next to the ID column.



Key Name	Columns
ID	ID
	☑ ID ☐ IMAGE

Drag **BUSIEST_AIRPORTS_2_IMAGE** table to **the 08 – ORE Sample** node in the business layer twice. Join the **BUSIEST_AIRPORTS_2_IMAGE** and **BUSIEST_AIRPORTS_2_IMAGE#1** tables in the Business Layer diagram using the join tool as we did in the previous example.

Expand **BUSIEST_AIRPORTS_2_IMAGE** and double-click the logical table source icon in the **Sources** folder. On the **Column Mapping** tab, click on the **IMAGE** column and then on the **Edit Expression** icon.



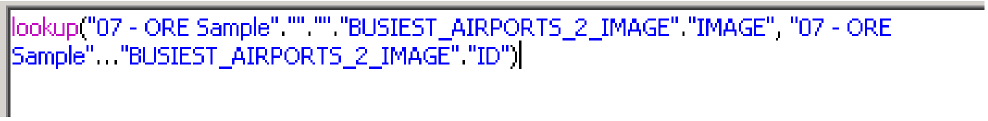
☑ Show mapped columns ☐ Show unmapped columns

Logical column to physical column mapping:

Logical Column	Expression
ID	ID
IMAGE	IMAGE

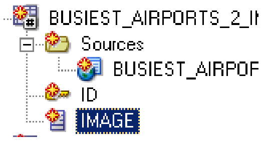
Place your cursor at the very beginning before the “07-ORE…” string and type **lookup(**

Place a comma at the end of the “…IMAGE” string. Then double-click **ID** in the **Columns** field on the left to add it to the expression. Close the expression with a “)” parenthesis mark at the very end. Your screen should look like this. Click **OK** and **OK**.

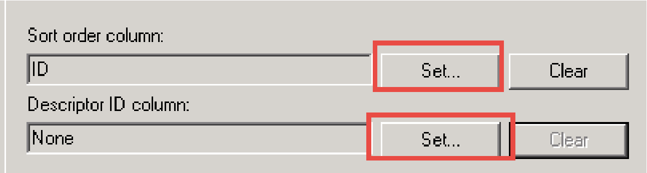


```
lookup("07 - ORE Sample"."".""."BUSIEST_AIRPORTS_2_IMAGE"."IMAGE", "07 - ORE
Sample"..."BUSIEST_AIRPORTS_2_IMAGE"."ID")
```

Double-click on the **IMAGE** column.

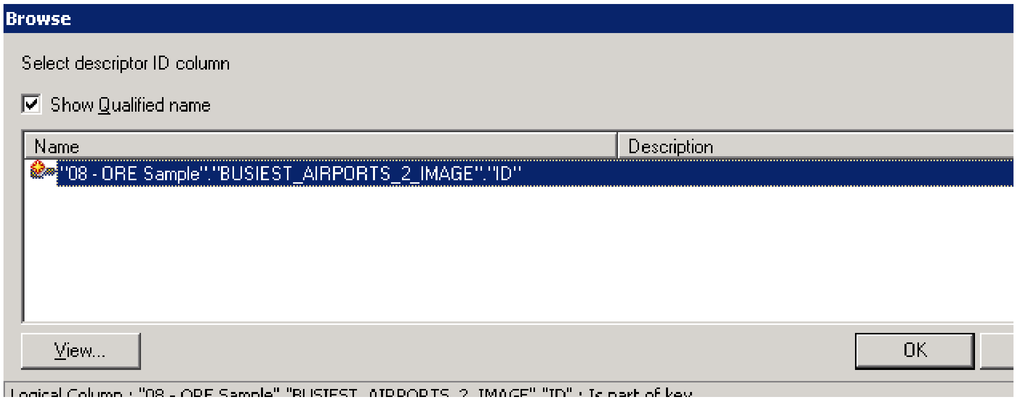


Click on the **General** tab. Click on **Set** and the **ID** column for both the **Sort Order Column and Descriptor ID Column**.



Sort order column: ID

Descriptor ID column: None



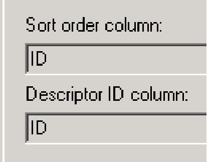
Browse

Select descriptor ID column

☑ Show Qualified name

Name	Description
"08 - ORE Sample"."BUSIEST_AIRPORTS_2_IMAGE"."ID"	

Logical Column : "08 - ORE Sample"."BUSIEST_AIRPORTS_2_IMAGE"."ID" : Is part of key



Sort order column: ID

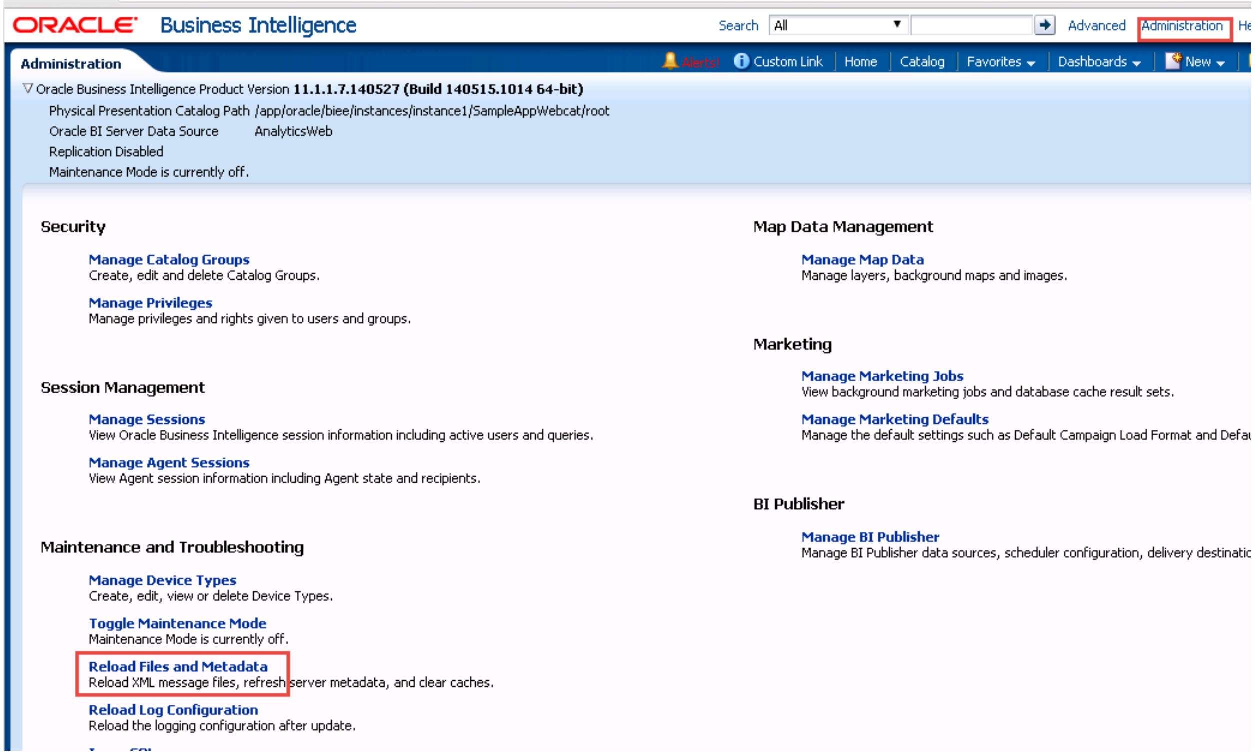
Descriptor ID column: ID

Drag **BUSIEST_AIRPORTS_2_IMAGE** table to **R _ ORE Sample** node in the **Presentation** layer.

Check in your changes and check Global Consistency. You should have 0 Errors and 11 Warnings.

Save your changes and exit BI Admin tool.

Open your browser and sign back in to OBIEE Prodney/Admin123 if you exited. Click the **Administration** link in the menu bar and click the **Reload Files and Metadata** link.


ORACLE Business Intelligence
Search All Advanced Administration He
Administration
Alerts! Custom Link Home Catalog Favorites Dashboards New

▽ Oracle Business Intelligence Product Version 11.1.1.7.140527 (Build 140515.1014 64-bit)
Physical Presentation Catalog Path /app/oracle/biee/instances/instance1/SampleAppWebcat/root
Oracle BI Server Data Source AnalyticsWeb
Replication Disabled
Maintenance Mode is currently off.

Security

Manage Catalog Groups
Create, edit and delete Catalog Groups.

Manage Privileges
Manage privileges and rights given to users and groups.

Session Management

Manage Sessions
View Oracle Business Intelligence session information including active users and queries.

Manage Agent Sessions
View Agent session information including Agent state and recipients.

Maintenance and Troubleshooting

Manage Device Types
Create, edit, view or delete Device Types.

Toggle Maintenance Mode
Maintenance Mode is currently off.

Reload Files and Metadata
Reload XML message files, refresh server metadata, and clear caches.

Reload Log Configuration
Reload the logging configuration after update.

Map Data Management

Manage Map Data
Manage layers, background maps and images.

Marketing

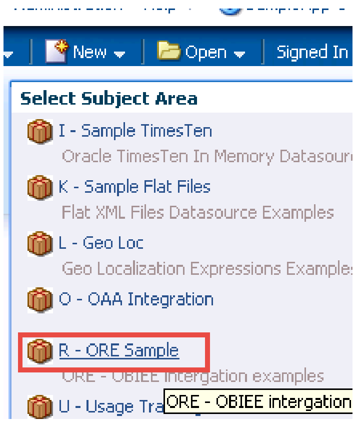
Manage Marketing Jobs
View background marketing jobs and database cache result sets.

Manage Marketing Defaults
Manage the default settings such as Default Campaign Load Format and Defau

BI Publisher

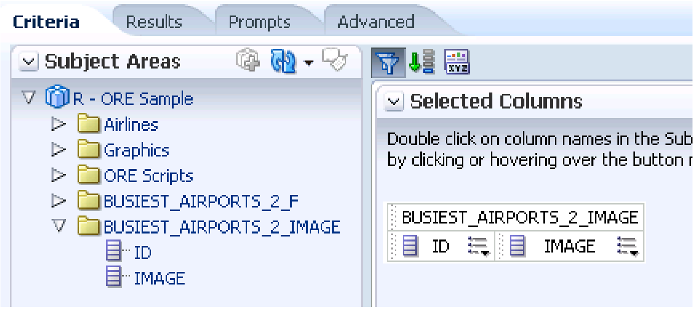
Manage BI Publisher
Manage BI Publisher data sources, scheduler configuration, delivery destinatic

Click on **New** then **Analysis** and then choose the **ORE Sample** subject area.



Select Subject Area
I - Sample TimesTen
Oracle TimesTen In Memory Datasour
K - Sample Flat Files
Flat XML Files Datasource Examples
L - Geo Loc
Geo Localization Expressions Example
O - OAA Integration
R - ORE Sample
ORE - OBIEE intergation examples
U - Usage Tra
ORE - OBIEE intergation

Open the **BUSIEST_AIRPORTS_2_IMAGE** folder and add both columns to the Selected Columns pane on the Criteria tab.



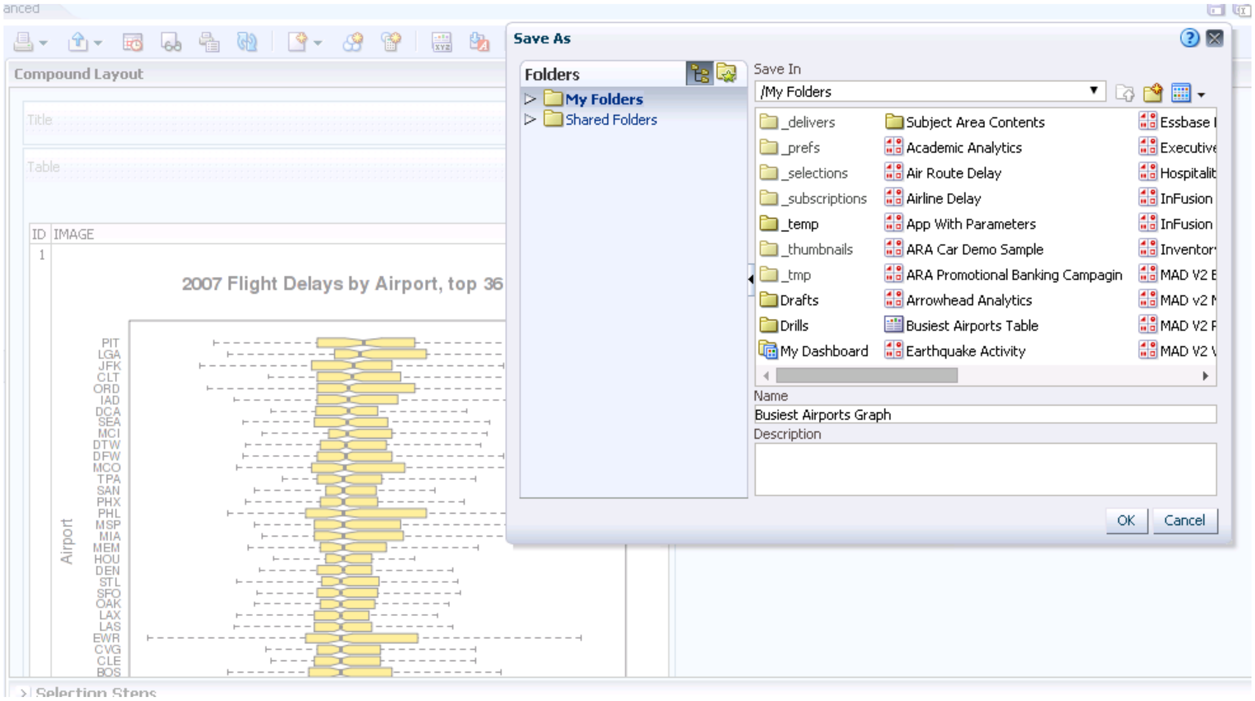
Double click on column names in the Sub
by clicking or hovering over the button r

BUSIEST_AIRPORTS_2_IMAGE

ID	IMAGE

Click on the **Results** tab.

You should see your box plot graph! Save your analysis.

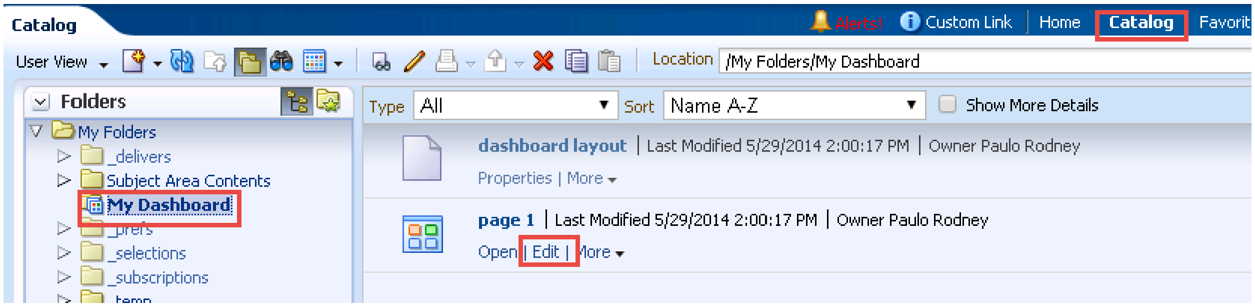


Save In: /My Folders

Name: Busiest Airports Graph

Description:

Click on Catalog in the menu bar and then edit page 1 of My Dashboard.

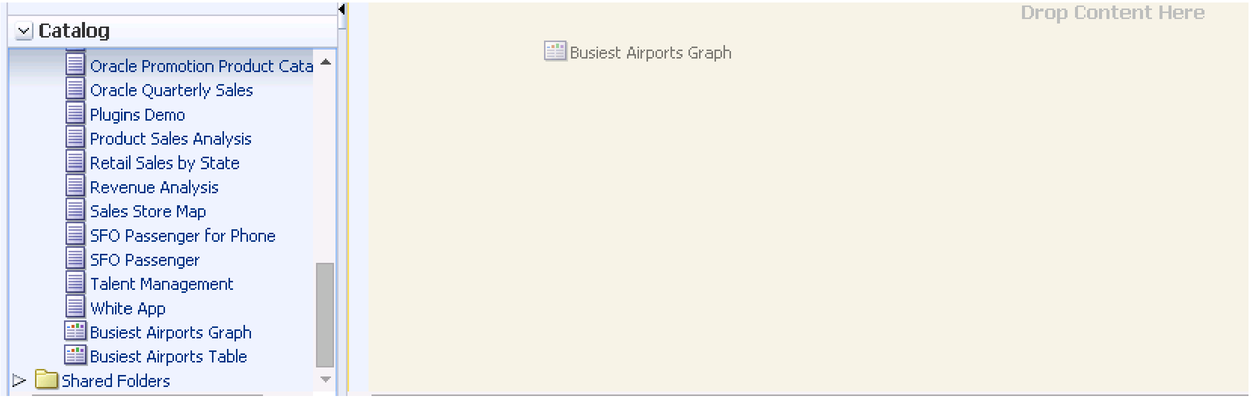


Location /My Folders/My Dashboard

dashboard layout | Last Modified 5/29/2014 2:00:17 PM | Owner Paulo Rodney

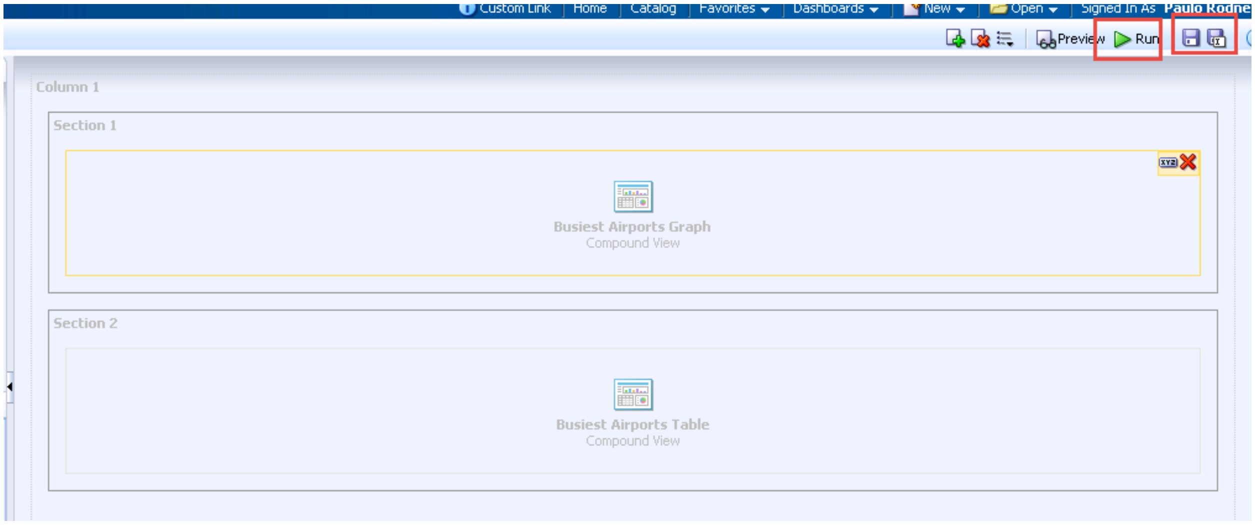
page 1 | Last Modified 5/29/2014 2:00:17 PM | Owner Paulo Rodney

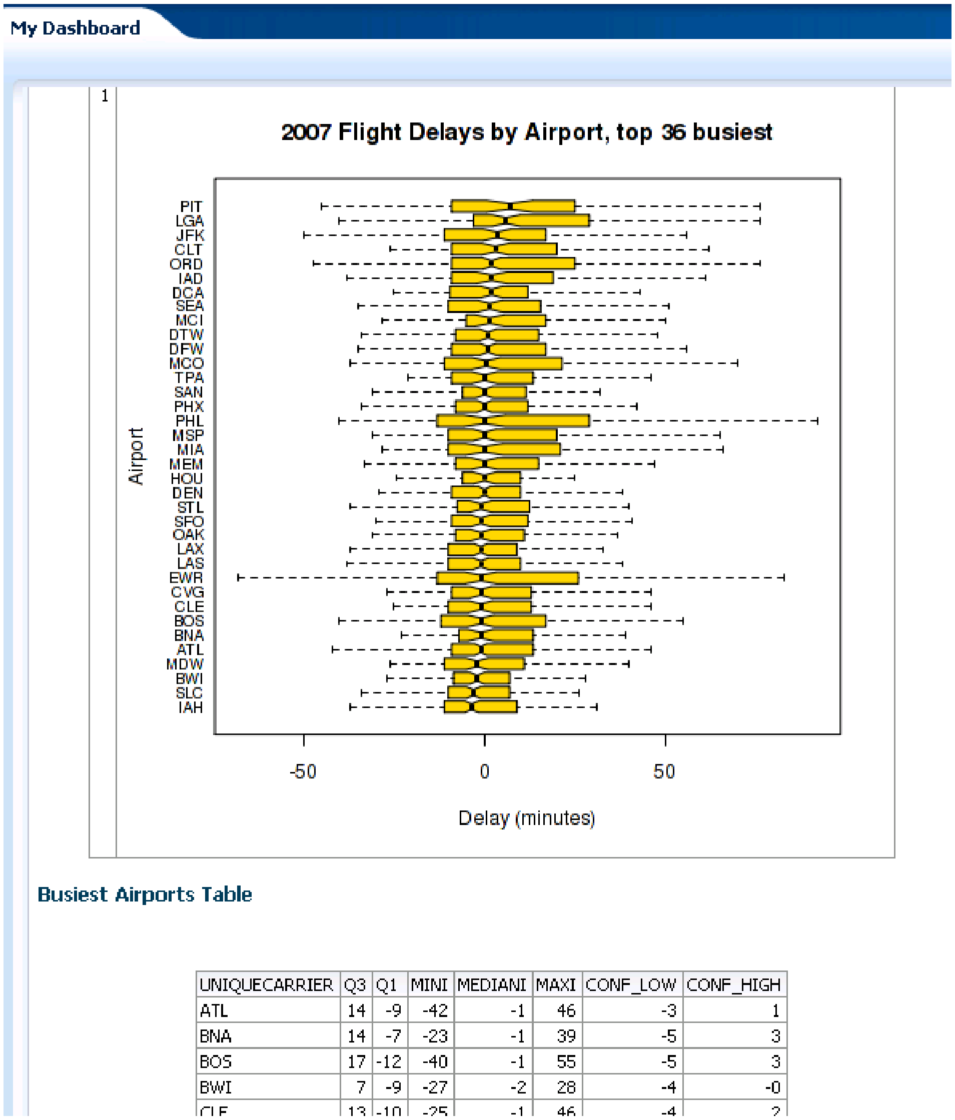
Navigate to your two saved analyses in the catalog directory on the left hand side and drag them on to your dashboard.



- Oracle Promotion Product Cata
- Oracle Quarterly Sales
- Plugins Demo
- Product Sales Analysis
- Retail Sales by State
- Revenue Analysis
- Sales Store Map
- SFO Passenger for Phone
- SFO Passenger
- Talent Management
- White App
- Busiest Airports Graph
- Busiest Airports Table
- Shared Folders

Click **Save** and then **Run**.





Busiest Airports Table

UNIQUECARRIER	Q3	Q1	MINI	MEDIANI	MAXI	CONF_LOW	CONF_HIGH
ATL	14	-9	-42	-1	46	-3	1
BNA	14	-7	-23	-1	39	-5	3
BOS	17	-12	-40	-1	55	-5	3
BWI	7	-9	-27	-2	28	-4	-0
CLE	13	-10	-25	-1	46	-4	2

We’re Finished!