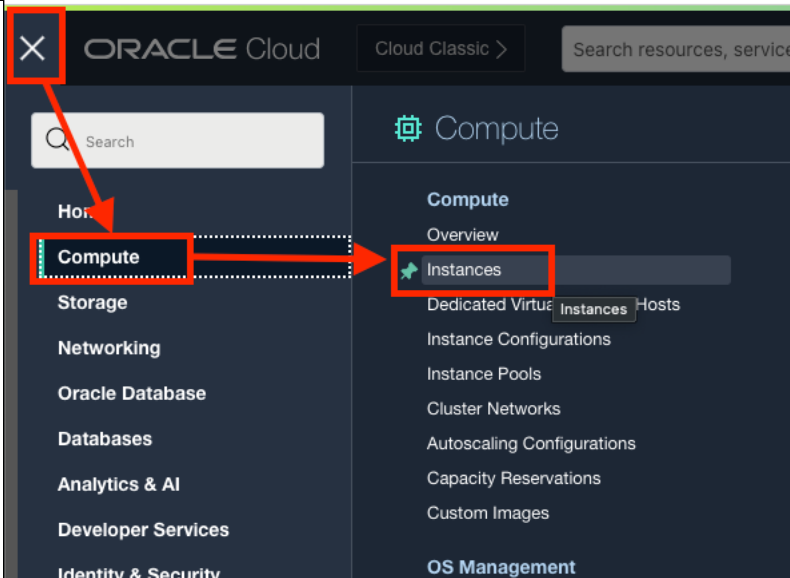
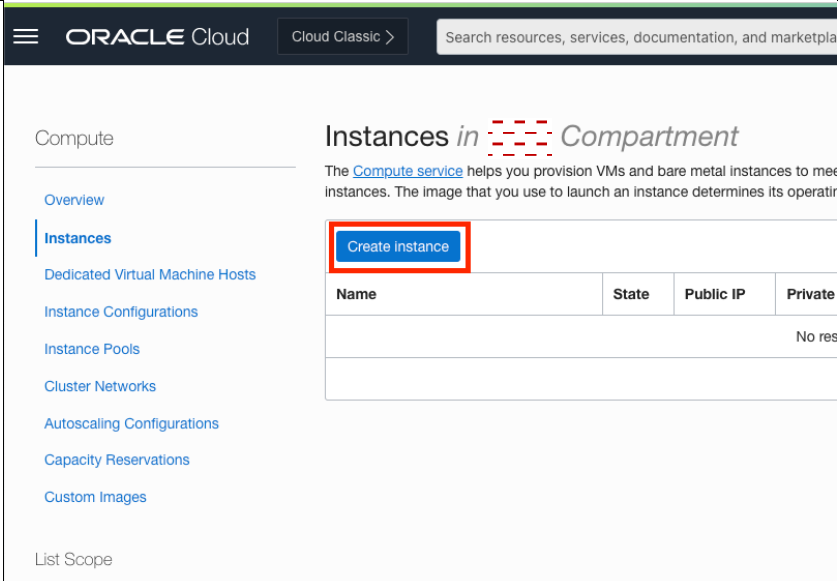


R Studio Install on Oracle Linux 7

This documentation will walk you through the steps to install R Studio Server within an Oracle Linux Virtual Machine running on Oracle Cloud Infrastructure. This documentation will then lay out the steps to establish a connection from your R Studio Server to the Autonomous Data Warehouse and query some data.

Step	Screenshot
<p>Login to your OCI Console at cloud.oracle.com</p> <p>Navigate to Menu > Compute > Instances.</p>	
<p>Click on 'Create Instance'.</p>	

Enter the following:

- VM Name
- Compartment
- Availability Domain
- OS Image – Oracle Linux 7.9

ORACLE Cloud Cloud Classic > Search resources, services, documentation, and marketplace

Create compute instance

Create an instance to deploy and run applications, or save as a reusable Terraform stack for creating an instance with Resource Manager.

Name: OL7-RStudio

Create in compartment: [Redacted]

Placement [Collapse](#)

The availability domain helps determine which shapes are available.

Availability domain: AD 1, AD 2, AD 3

[Show advanced options](#)

Image and shape [Collapse](#)

A [shape](#) is a template that determines the number of CPUs, amount of memory, and other resources allocated to an instance. The image is the operating system that runs on top of the shape.

Image: Oracle Linux 7.9
Image build: 2022.04.04-0

[Change image](#)

Select your VM Shape Configuration.**I have gone for:**

- Virtual Machine
- AMD Series
- 2 OCPU
- 32 GB Memory

Browse all shapes

A [shape](#) is a template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance.

Instance type

Virtual machine ☒ A virtual machine is an independent computing environment that runs on top of physical bare metal hardware.

Bare metal machine A bare metal compute instance gives you dedicated physical server access for highest performance and strong isolation.

Shape series

AMD Flexible OCPU count. Current generation AMD processors.

Intel Flexible OCPU count. Current generation Intel processors.

Ampere Arm-based processor.

Specialty and previous generation Always Free, Dense I/O, GPU, HPC, and earlier generation AMD and Intel standard shapes.

Image: Oracle Linux 7.9

Shape name	OCPU	Memory (GB)	Network bandwidth (Gbps)	Max. total VNICS
<input checked="" type="checkbox"/> VM.Standard.E4.Flex	2	32	2	2

You can customize the number of OCPUs and the amount of memory allocated to a flexible shape. The other resources scale proportionately. [Learn more about flexible shapes.](#)

Number of OCPUs: 1 2 22 43 64

☐ Burstable [?](#)

Amount of memory (GB) [?](#): 1 32 342 683 1024

1 Selected Showing 1 item

Don't see the shape you want? [View your service limits and request an increase.](#) If you're looking for an older shape, check the **Specialty and previous generation** section.

[Select shape](#) [Cancel](#)



Select an existing **VCN** and **Public Subnet** if you have one available.

Else you can create a new VCN and Public Subnet.

Ensure **“Assign a public IPv4 address”** is selected.

Upload existing SSH Public Key File if you have an existing key.

Else we can generate a new pair.

Click on **‘Save Private Key’** and **‘Save Public Key’**.

On Mac – it is recommended to save these keys under the directory **~/ssh**.

It is also recommended to update the permissions to the files as **chmod 600**.

```

-rw-----@ 1 isyed staff 1675 28 Apr 09:28 rstudio-server-private.key
-rw-----@ 1 isyed staff 399 28 Apr 09:28 rstudio-server-public.pub
  
```

You have the option to specify a bigger boot volume that the default (46.6GB).

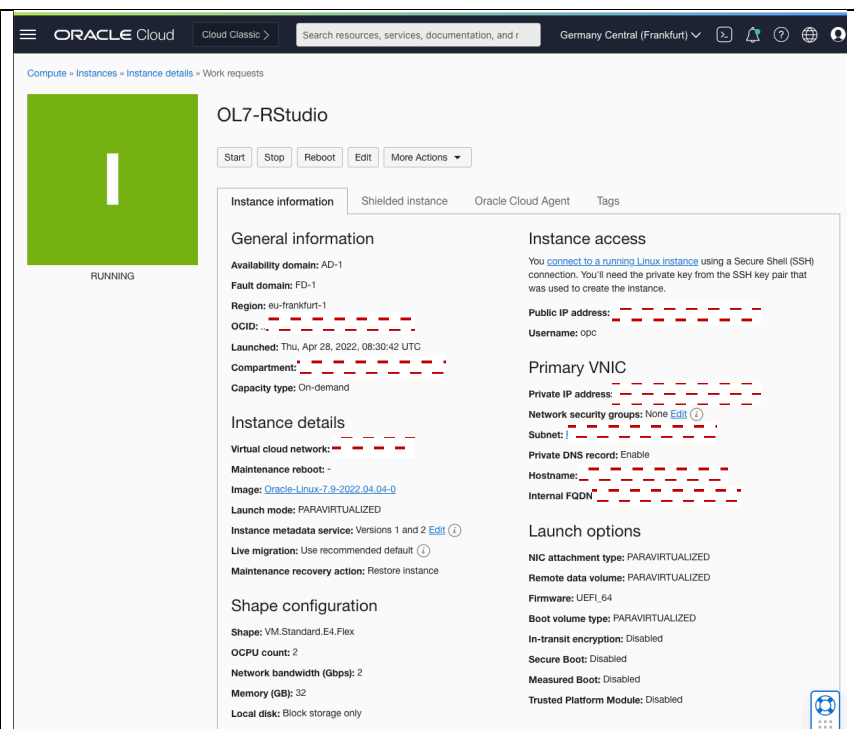
I will leave these settings as default.

Click on **‘Create’** and the creation of your VM will begin.



Once the VM is up and running you should see the status turn from **'Creating'** to **'Running'**.

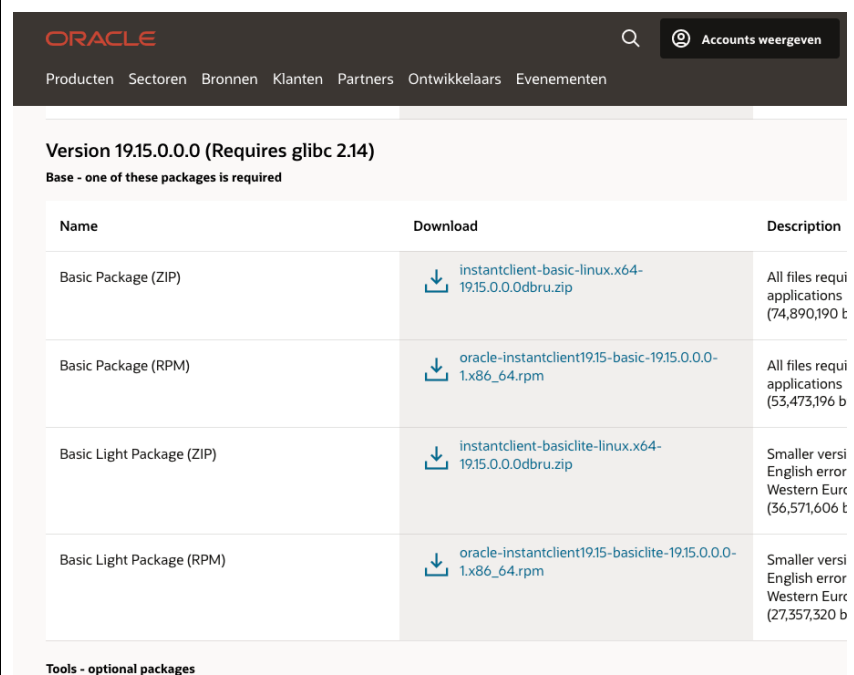
Make note of the Public IP Address and Username displayed. Under '**Instance Information**'.



Now we will download a series of artifacts that will allow us to connect into the Autonomous Data Warehouse from our R Studio Server.

First download the
Oracle Instant Client
Version 19c.

<https://www.oracle.com/uk/database/technologies/instant-client/linux-x86-64-downloads.html>



Download the following packages:

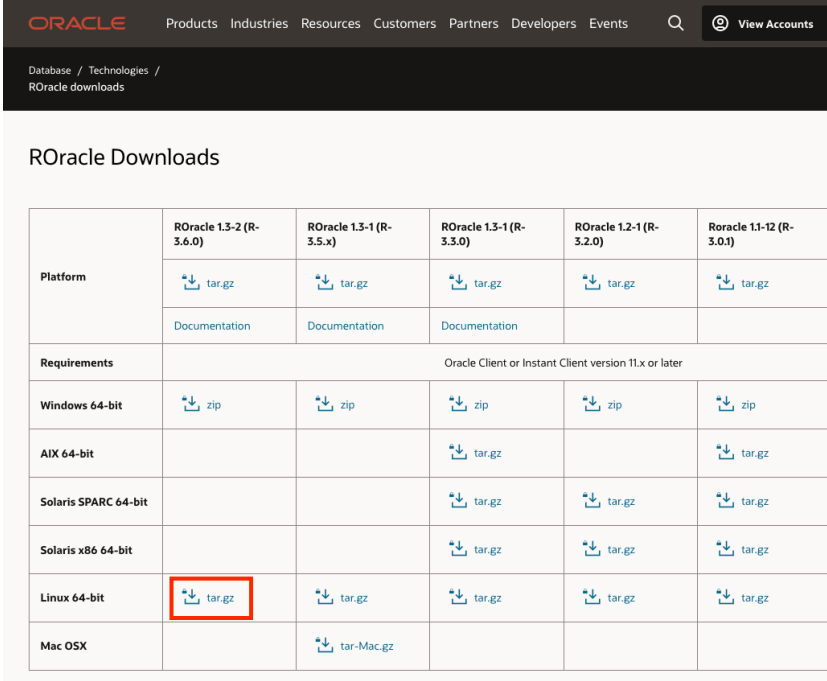
```
oracle-instantclient19.15-basic-19.15.0.0.0-1.x86_64.rpm
oracle-instantclient19.15-sqlplus-19.15.0.0.0-1.x86_64.rpm
oracle-instantclient19.15-tools-19.15.0.0.0-1.x86_64.rpm
oracle-instantclient19.15-devel-19.15.0.0.0-1.x86_64.rpm
oracle-instantclient19.15-jdbc-19.15.0.0.0-1.x86_64.rpm
oracle-instantclient19.15-odbc-19.15.0.0.0-1.x86_64.rpm
```



Now we will download R Oracle -
<https://www.oracle.com/database/technologies/roacle-downloads.html>

I have downloaded **R Oracle 1.3-2**.

ROracle is an open-source R package supporting a DBI-compliant Oracle driver based on the high performance OCI library. ROracle is publicly available on the Comprehensive R Archive Network (CRAN) and is maintained by Oracle. It enables highly scalable and performant connectivity to Oracle Database for data transfer, along with enables transaction level control and execution of user-provided SQL.



ORACLE Products Industries Resources Customers Partners Developers Events View Accounts

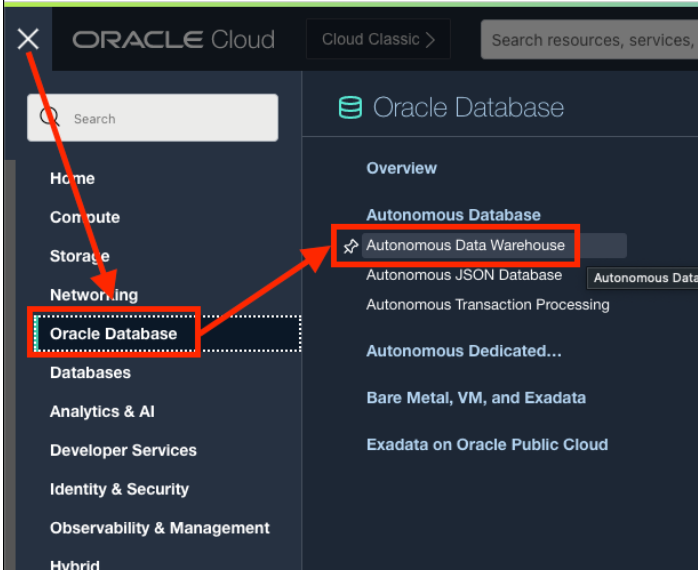
Database / Technologies / ROracle downloads

ROracle Downloads

Platform	ROracle 1.3-2 (R-3.6.0)	ROracle 1.3-1 (R-3.5.x)	ROracle 1.3-1 (R-3.3.0)	ROracle 1.2-1 (R-3.2.0)	ROracle 1.1-12 (R-3.0.1)
	tar.gz	tar.gz	tar.gz	tar.gz	tar.gz
	Documentation	Documentation	Documentation		
Requirements	Oracle Client or Instant Client version 11.x or later				
Windows 64-bit	zip	zip	zip	zip	zip
AIX 64-bit			tar.gz		tar.gz
Solaris SPARC 64-bit			tar.gz	tar.gz	tar.gz
Solaris x86 64-bit			tar.gz	tar.gz	tar.gz
Linux 64-bit	tar.gz	tar.gz	tar.gz	tar.gz	tar.gz
Mac OSX		tar-Mac.gz			

Finally, we will download our Autonomous Data Warehouse Wallet File which contains the connection details for our Database.

Within the OCI Console Navigate to **Menu > Oracle Database > Autonomous Data Warehouse**.



ORACLE Cloud Cloud Classic > Search resources, services, >

Search

- Home
- Compute
- Storage
- Networking
- Oracle Database**
- Databases
- Analytics & AI
- Developer Services
- Identity & Security
- Observability & Management
- Hybrid

Oracle Database

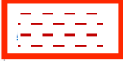

Overview

- Autonomous Database**
- Autonomous Data Warehouse**
- Autonomous JSON Database
- Autonomous Data Warehouse
- Autonomous Transaction Processing
- Autonomous Dedicated...
- Bare Metal, VM, and Exadata
- Exadata on Oracle Public Cloud

Select your Autonomous Data Warehouse.

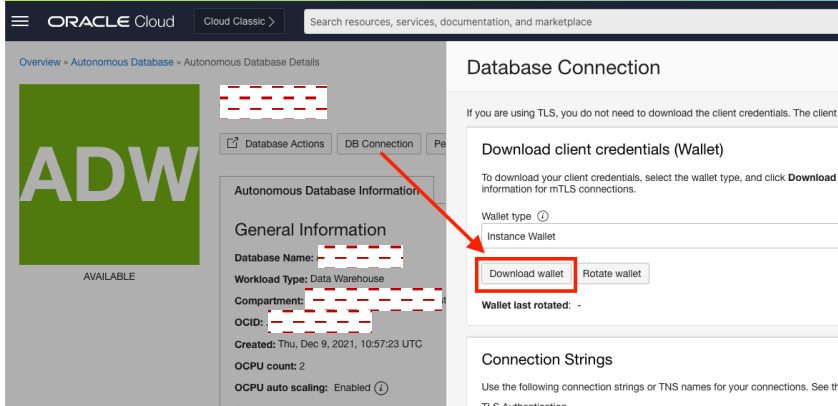
Autonomous Databases in Compartment

Autonomous Database delivers fast performance and requires no database administration. It performs all rout while the system is running, without human intervention. Autonomous Databases located in the Oracle cloud c infrastructure. [Learn more](#).

Create Autonomous Database						
Display Name	State	Dedicated	OCPUs	Storage	Workload Type	Auto
	 Available	No	2	1 TB	Data Warehouse	—

Displaying 1 Auto

Under the DB Connection Tab, click on 'Download Wallet'.



ORACLE Cloud Cloud Classic > Search resources, services, documentation, and marketplace


Overview > Autonomous Database > Autonomous Database Details

ADW AVAILABLE


Database Actions DB Connection Pe


Autonomous Database Information

General Information

Database Name: 

Workload Type: Data Warehouse

Compartment: 

OCID: 

Created: Thu, Dec 9, 2021, 10:57:23 UTC

OCPU count: 2

OCPU auto scaling: Enabled ⓘ

Database Connection

If you are using TLS, you do not need to download the client credentials. The client

Download client credentials (Wallet)

To download your client credentials, select the wallet type, and click **Download** information for mTLS connections.

Wallet type ⓘ

Instance Wallet

Download wallet Rotate wallet

Wallet last rotated: -

Connection Strings

Use the following connection strings or TNS names for your connections. See th

TLS Authentication

Enter a Password, Confirm the Password and then Click on Download.

Download Wallet

Database connections to your Autonomous Database use a secure connection. The wallet file database clients and tools to access Autonomous Database.

Please create a password for this wallet. Some database clients will require that you provide to your database (other clients will auto-login using the wallet without a password).

Password

Confirm password

Download Cancel

Using the Public IP Address and Username stated within your VM Instance Details Page, use your local machine to SSH into your VM.

ssh -i ~/.ssh/rstudio-server-private.key opc@<host-public-ip>

```
$ ssh -i rstudio-server-private.key opc@
The authenticity of host
ED25519 key fingerprint
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added (to the list of known hosts.
```



We will now switch user to root within the environment and create a temp directory to store our Oracle Instant Client downloaded artifacts.

`sudo su -`

`mkdir /tmp/oracInt`

`chmod 0777 /tmp/oracInt`

We can then exit from out VM.

`Exit`

```
$
$
$ sudo su -
[# mkdir /tmp/oracInt
[# chmod 0777 /tmp/oracInt
[# exit

logout
$ exit

logout
Connection to [redacted] closed.
$
$
$
```

From our local machine let's upload the downloaded Oracle Instant Client artifacts into the VM tmp folder we just created.

`scp -i ~/.ssh/rstudio-server-private.key ~/Downloads/oracle-* opc@<host-public-ip>:/tmp/oracInt`

```
[redacted]$ scp -i ~/.ssh/rstudio-server-private.key ~/Downloads/oracle-*
[redacted]:/tmp/oracInt
oracle-instantclient19.15-basic-19.15.0.0-1.x86_64.rpm      100% 51MB 3.7MB/s 00:13
oracle-instantclient19.15-devel-19.15.0.0-1.x86_64.rpm   100% 600KB 2.8MB/s 00:00
oracle-instantclient19.15-jdbc-19.15.0.0-1.x86_64.rpm    100% 1489KB 4.0MB/s 00:00
oracle-instantclient19.15-odbc-19.15.0.0-1.x86_64.rpm    100% 241KB 2.8MB/s 00:00
oracle-instantclient19.15-sqlplus-19.15.0.0-1.x86_64.rpm 100% 687KB 3.6MB/s 00:00
oracle-instantclient19.15-tools-19.15.0.0-1.x86_64.rpm   100% 817KB 3.7MB/s 00:00
```

Now let's upload the downloaded Autonomous Data Warehouse Wallet file to the /tmp directory on our VM.

`scp -i ~/.ssh/rstudio-server-private.key ~/Downloads/Wallet* opc@<host-public-ip>:/tmp`

```
[redacted]$
[redacted]:/tmp
[redacted]$ scp -i ~/.ssh/rstudio-server-private.key ~/Downloads/Wallet*
[redacted]:/tmp
100% 21KB 299.7KB/s 00:00
```



Now let's upload the downloaded R Oracle Package to the /tmp directory on our VM.

```
scp -i ~/.ssh/rstudio-server-private.key
~/Downloads/ROracle*
opc@<host-public-ip>:/tmp
```

```
$
$ scp -i ~/.ssh/rstudio-server-private.key ~/Downloads/ROracle*
~/tmp
ROracle_1.3-2_R_x86_64-linux-gnu.tar.gz 100% 512KB 2.2MB/s 00:00
```

SSH back into your VM to check if the files have all be copied across correctly.

```
ssh -i ~/.ssh/rstudio-server-private.key
opc@<host-public-ip>
```

```
ls -la /tmp
```

```
ls -la /tmp/oracle
```

```
$ ssh -i ~/.ssh/rstudio-server-private.key
Last login: Thu Apr 28 09:27:49 2022 from d
$
$ ls -la /tmp
total 572
drwxrwxrwt. 11 root root 4096 Apr 28 09:26
dr-xr-xr-x. 17 root root 4096 Mar 24 03:39 ..
-rw----- 1 root root 1132 Apr 28 08:31 dhclient-exit-hooksRza.log
drwxrwxrwt. 2 root root 6 Mar 24 03:06 .font-unix
drwxrwxrwt. 2 root root 6 Mar 24 03:06 .ICE-unix
-rw----- 1 root root 16178 Apr 28 09:27 .nstat.u0
drwxrwxrwt. 2 root root 4096 Apr 28 09:23 .X11-unix
-rw-r--r-- 1 opc opc 524658 Apr 28 09:26 ROracle_1.3-2_R_x86_64-linux-gnu.tar.gz
drwx----- 3 root root 17 Apr 28 08:31 systemd-private-2ab8e7cb44b46d09c56a166968472ec-chrony
d.service-1QTau1
drwx----- 3 root root 17 Apr 28 08:38 systemd-private-2ab8e7cb44b46d09c56a166968472ec-unifie
d-monitoring-agent.service-JkUFZv
drwxrwxrwt. 2 root root 6 Mar 24 03:06 .test-unix
drwxr-xr-x. 2 root root 6 Apr 28 08:38 unified-monitoring-agent
-rw-r--r-- 1 opc opc 21645 Apr 28 09:25 Wallet- zip
drwxrwxrwt. 2 root root 6 Mar 24 03:06 .X11-unix
drwxrwxrwt. 2 root root 6 Mar 24 03:06 .XIM-unix
$
$ ls -la /tmp/oracle
total 56072
drwxrwxrwt. 2 root root 4096 Apr 28 09:23
drwxrwxrwt. 11 root root 4096 Apr 28 09:26 ..
-rw-r--r-- 1 opc opc 53473196 Apr 28 09:23 oracle-instantclient19.15-basic-19.15.0.0-1.x86_64.
rpm
-rw-r--r-- 1 opc opc 613996 Apr 28 09:23 oracle-instantclient19.15-devel-19.15.0.0-1.x86_64.
rpm
-rw-r--r-- 1 opc opc 1524740 Apr 28 09:23 oracle-instantclient19.15-jdbc-19.15.0.0-1.x86_64.r
pm
-rw-r--r-- 1 opc opc 246700 Apr 28 09:23 oracle-instantclient19.15-odbc-19.15.0.0-1.x86_64.r
pm
-rw-r--r-- 1 opc opc 703356 Apr 28 09:23 oracle-instantclient19.15-sqlplus-19.15.0.0-1.x86.6
4.rpm
-rw-r--r-- 1 opc opc 836528 Apr 28 09:23 oracle-instantclient19.15-tools-19.15.0.0-1.x86_64.
rpm
```

While we are logged into the VM let's modify some of the system settings.

```
Switch to root:
sudo su -
```

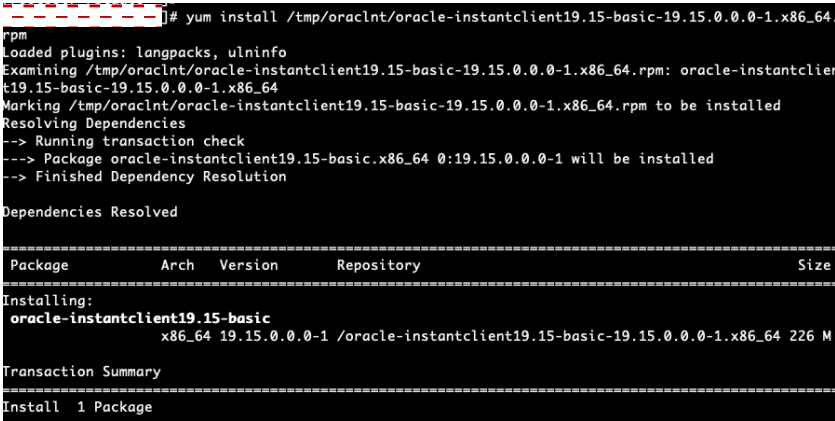
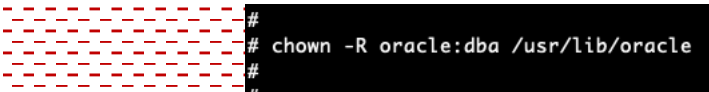
```
Disable the firewall:
systemctl disable
firewalld
systemctl stop
firewalld
```

```
$
$ sudo su -
Last login: Thu Apr 28 09:12:37 GMT 2022 on pts/0
#
# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
#
# systemctl stop firewalld
#
```



<p>Next, we will edit the <code>/etc/selinux/config</code> file and set the SELINUX parameter to permissive.</p> <p><code>vi /etc/selinux/config</code></p> <p><code>Save and Quit (:wq)</code></p>	 <pre># enforcing - SELinux security policy is enforced. # permissive - SELinux prints warnings instead of enforcing. # disabled - No SELinux policy is loaded. SELINUX=permissive # SELINUXTYPE can take one of three values: # targeted - Targeted processes are protected, # minimum - Modification of targeted policy. Only selected processes are protected. # mls - Multi Level Security protection. SELINUXTYPE=targeted :wq</pre>
<p>We will now install R on the VM.</p> <p><code>yum install R</code></p>	 <pre>tk.x86_64 1:8.5.13-6.el7 tk-devel.x86_64 1:8.5.13-6.el7 urw-base35-bookman-fonts.noarch 0:20170801-10.el7 urw-base35-c059-fonts.noarch 0:20170801-10.el7 urw-base35-d0500001-fonts.noarch 0:20170801-10.el7 urw-base35-fonts.noarch 0:20170801-10.el7 urw-base35-fonts-common.noarch 0:20170801-10.el7 urw-base35-gothic-fonts.noarch 0:20170801-10.el7 urw-base35-nimbus-mono-ps-fonts.noarch 0:20170801-10.el7 urw-base35-nimbus-roman-fonts.noarch 0:20170801-10.el7 urw-base35-nimbus-sans-fonts.noarch 0:20170801-10.el7 urw-base35-p052-fonts.noarch 0:20170801-10.el7 urw-base35-standard-symbols-ps-fonts.noarch 0:20170801-10.el7 urw-base35-z003-fonts.noarch 0:20170801-10.el7 xorg-x11-font-utils.x86_64 1:7.5-21.el7 xorg-x11-proto-devel.noarch 0:2018.4-1.el7 xorg-x11-server-utils.x86_64 0:7.7-20.el7 zlib-devel.x86_64 0:1.2.7-19.el7_9 zziplib.x86_64 0:0.13.62-12.el7 Dependency Updated: expat.x86_64 0:2.1.0-14.0.1.el7_9 Complete!</pre>
<p>We will now download and install the Oracle Preinstallation RPM and the Oracle Database RPM packages.</p> <p><code>yum install oracle-database-preinstall-21c</code></p>	 <pre>yum install oracle-database-preinstall-21c Loaded plugins: langpacks, ulninfo Resolving Dependencies --> Running transaction check --> Package oracle-database-preinstall-21c.x86_64 0:1.0-1.el7 will be installed --> Processing Dependency: psmisc for package: oracle-database-preinstall-21c-1.0-1.el7.x86_64 --> Processing Dependency: xorg-x11-utils for package: oracle-database-preinstall-21c-1.0-1.el7.x86_64 --> Processing Dependency: xorg-x11-xauth for package: oracle-database-preinstall-21c-1.0-1.el7.x86_64 --> Running transaction check --> Package psmisc.x86_64 0:22.20-17.el7 will be installed --> Package xorg-x11-utils.x86_64 0:7.5-23.el7 will be installed --> Processing Dependency: libXtst.so.6(C)(64bit) for package: xorg-x11-utils-7.5-23.el7.x86_64 --> Processing Dependency: libXv.so.1(C)(64bit) for package: xorg-x11-utils-7.5-23.el7.x86_64 --> Processing Dependency: libXxf86dga.so.1(C)(64bit) for package: xorg-x11-utils-7.5-23.el7.x86_64 --> Processing Dependency: libdmx.so.1(C)(64bit) for package: xorg-x11-utils-7.5-23.el7.x86_64 --> Package xorg-x11-xauth.x86_64 1:1.0.9-1.el7 will be installed --> Running transaction check --> Package libXtst.x86_64 0:1.2.3-1.el7 will be installed --> Package libXv.x86_64 0:1.0.11-1.el7 will be installed --> Package libXxf86dga.x86_64 0:1.1.4-2.1.el7 will be installed</pre>



<p>We can now yum install all the Oracle Instant Client Packages we downloaded earlier and saved within the /tmp/oracInt directory.</p>	<pre># yum install /tmp/oracInt/oracle-instantclient19.15-basic-19.15.0.0-1.x86_64.rpm</pre>  <pre> yum install /tmp/oracInt/oracle-instantclient19.15-basic-19.15.0.0-1.x86_64.rpm yum install /tmp/oracInt/oracle-instantclient19.15-devel-19.15.0.0-1.x86_64.rpm yum install /tmp/oracInt/oracle-instantclient19.15-jdbc-19.15.0.0-1.x86_64.rpm yum install /tmp/oracInt/oracle-instantclient19.15-odbc-19.15.0.0-1.x86_64.rpm yum install /tmp/oracInt/oracle-instantclient19.15-sqlplus-19.15.0.0-1.x86_64.rpm yum install /tmp/oracInt/oracle-instantclient19.15-tools-19.15.0.0-1.x86_64.rpm </pre>
<p>Now, change ownership of the /usr/lib/oracle directory to the oracle user. This directory contains the installation of Oracle Instant Client.</p> <p><i>chown -R oracle:dba /usr/lib/oracle</i></p> <p>Then switch to the Oracle User.</p> <p><i>sudo su - oracle</i></p>	 <pre># chown -R oracle:dba /usr/lib/oracle</pre>  <pre># sudo su - oracle</pre>
<p>Now create a directory within our Instant Client Installation to store the Wallet File for the Autonomous Data Warehouse.</p> <p><i>mkdir -p /usr/lib/oracle/19.15/client64/network/admin</i></p>	 <pre>\$ mkdir -p /usr/lib/oracle/19.15/client64/network/admin</pre>



<p>Navigate into our newly created directory and unzip the Wallet file stored within the tmp directory.</p> <pre>cd /usr/lib/oracle/19.15/ client64/network/adm in unzip /tmp/Wallet_*</pre>	<pre>\$ \$ cd /usr/lib/oracle/19.15/client64/network/admin \$ \$ \$ unzip /tmp/Wallet_*.zip Archive: /tmp/Wallet_*.zip inflating: README inflating: cwallet.sso inflating: tnsnames.ora inflating: truststore.jks inflating: ojdbc.properties inflating: sqlnet.ora inflating: ewallet.p12 inflating: keystore.jks \$ ls cwallet.sso keystore.jks README tnsnames.ora ewallet.p12 ojdbc.properties sqlnet.ora truststore.jks</pre>
<p>Logout from Oracle user – exit</p> <p>Login as root – sudo su</p> <p>-</p> <p>Download R Studio Server: wget https://download2.rstudio.org/server/centos7/x86_64/rstudio-server-rhel-2021.09.2-382-x86_64.rpm</p> <p>Install R Studio Server: yum install rstudio-server-rhel-2021.09.2-382-x86_64.rpm</p>	<pre>\$ clear \$ exit logout Last login: Thu Apr 28 09:33:05 GMT 2022 on pts/0 # wget https://download2.rstudio.org/server/centos7/x86_64/rstudio-server-rhel-2021.09.2-382-x86_64.rpm --2022-04-28 10:14:03-- https://download2.rstudio.org/server/centos7/x86_64/rstudio-server-rhel-2021.09.2-382-x86_64.rpm Resolving download2.rstudio.org (download2.rstudio.org)... 13.224.195.109, 13.224.195.14, 13.224.195.101, ... Connecting to download2.rstudio.org (download2.rstudio.org) 13.224.195.109 :443... connected. HTTP request sent, awaiting response... 200 OK Length: 61202392 (58M) [application/x-redhat-package-manager] Saving to: 'rstudio-server-rhel-2021.09.2-382-x86_64.rpm' 100%[=====] 61,202,392 83.5MB/s in 0.7s 2022-04-28 10:14:03 (83.5 MB/s) - 'rstudio-server-rhel-2021.09.2-382-x86_64.rpm' saved [61202392/61202392] # yum install rstudio-server-rhel-2021.09.2-382-x86_64.rpm Loaded plugins: langpacks, ulninfo Examining rstudio-server-rhel-2021.09.2-382-x86_64.rpm: rstudio-server-2021.09.2+382-1.x86_64 Marking rstudio-server-rhel-2021.09.2-382-x86_64.rpm to be installed Resolving Dependencies --> Running transaction check --> Package rstudio-server.x86_64 0:2021.09.2+382-1 will be installed --> Processing Dependency: postgresql-libs for package: rstudio-server-2021.09.2+382-1.x86_64</pre>
<p>We will now edit the R Studio Config File:</p> <pre>vi /etc/rstudio/rserver.conf</pre> <p>Set the following parameters listed on the right-hand side.</p> <p>Exit the editor</p> <pre>:wq</pre>	<pre># Server Configuration File www-port=8787 www-address=127.0.0.1 rsession-ld-library-path=/usr/lib64/R/lib:/usr/lib/oracle/19.15/client64/lib</pre> <pre>www-port=8787 www-address=127.0.0.1 rsession-ld-library-path=/usr/lib64/R/lib:/usr/lib/oracle/19.15/client64/lib</pre>



<p>Next, we will set the Oracle Home within the R Environment file.</p> <pre>vi /usr/lib64/R/etc/Renviron.site ORACLE_HOME=/usr/lib/oracle/19.15/client64</pre> <p>Exit the editor</p> <pre>:wq</pre>	 <pre># # vi /usr/lib64/R/etc/Renviron.site # ORACLE_HOME=/usr/lib/oracle/19.15/client64</pre>
<p>After making the above changes we will reboot the VM for the changes to take effect.</p> <p>From within the OCI Console, visit your VM from Menu > Compute > Instance > Your Instance.</p> <p>Click on Reboot.</p>	 <p>ORACLE Cloud Cloud Classic > Search resources, services, documentation,</p> <p>Compute » Instances » Instance details</p> <p>OL7-RStudio</p> <p>Start Stop Reboot Edit More Actions ▾</p> <p>Instance information Shielded instance Ora</p> <p>General information</p> <p>Availability domain: AD-1</p> <p>Fault domain: FD-1</p> <p>Region: ...</p> <p>OCID: ...</p> <p>Launched: ...</p> <p>Compar...</p>
<p>Once rebooted, SSH back into our R Studio Server VM and switch to the root user</p> <pre>ssh -i ~/.ssh/rstudio-server-private.key opc@<host-ip-address></pre> <p>sudo su -</p> <p>Set the Oracle user a password:</p> <pre>passwd oracle</pre>	 <pre>\$ ssh -i ~/.ssh/rstudio-server-private.key Last login: Thu Apr 28 10:37:28 2022 from ... \$ sudo su - Last login: Thu Apr 28 10:37:33 GMT 2022 on pts/0 # # passwd oracle Changing password for user oracle. New password: Retype new password: passwd: all authentication tokens updated successfully.</pre>



Before we make a connection from our local machine to the R Studio Server we must first configure a few networking components.

First, we will edit the security list of our Public Subnet where our R Studio Server lives to accept traffic coming on **port 8787**.

From your R Studio Server VM Instance Page, **Click on the Subnet Link**.

ORACLE Cloud Cloud Classic > Search resources, services, documentation, and marketplace Germany Central (Frankfurt)

Compute > Instances > Instance details

OL7-RStudio

Start Stop Reboot Edit More Actions

Instance information Shielded instance Oracle Cloud Agent Tags

General information

Availability domain: AD-1
Fault domain: FD-1
Region: ...
OCID: ...
Launched: ...
Compartment: ...
Capacity: ...

Instance details

Virtual cloud network: ...
Maintenance reboot: -

Instance access

You connect to a running Linux instance using connection. You'll need the private key from the create the instance.

Public IP address: ...
Username: opc

Primary VNIC

Private IP address: ...
Network security groups: None Edit
Subnet: Public S
Private DNS record: Enable

Click on the **'Default'** Security List.

Resources

Security Lists (1)

Logs

Tag filters add | clear
no tag filters applied

Security Lists

Add Security List

Name	State
Default Security List	Available

Click on **'Add Ingress Rule'**.

ORACLE Cloud Cloud Classic > Search resources, services, documentation

Networking > Virtual Cloud Networks > Security List Details

Default Security List for

Instance traffic is controlled by firewall rules on each instance

Move resource Add Tags Terminate

Security List Information Tags

OCID: ...
Created: Tue, Dec 21, 2021, 16:09:48 UTC

Resources

Ingress Rules (6)
Egress Rules (1)

Add Ingress Rules Edit Remove

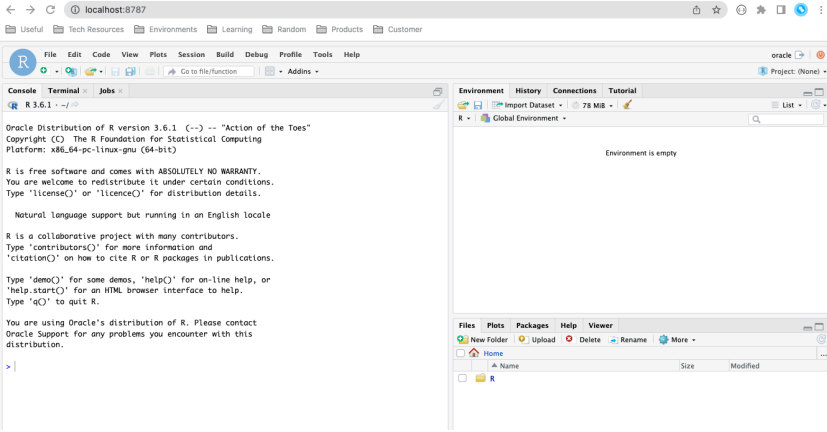
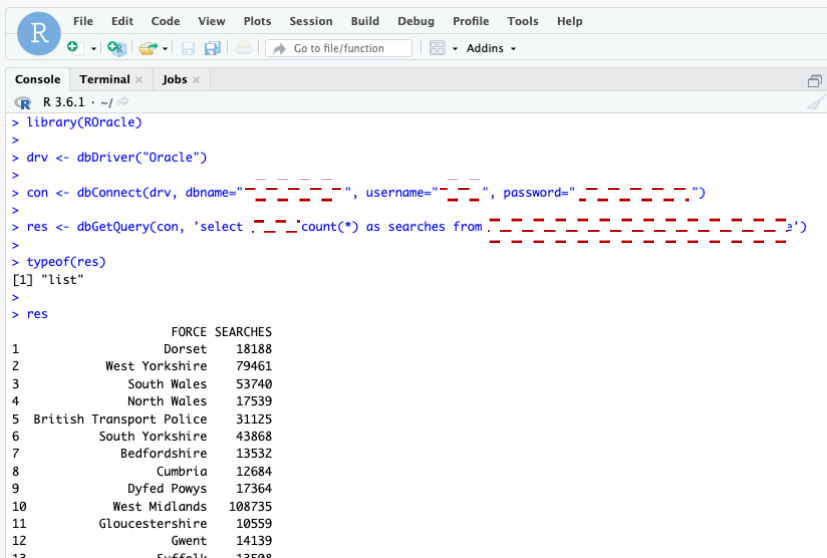
Protocol	Source	Destination	Port Range	Priority	State



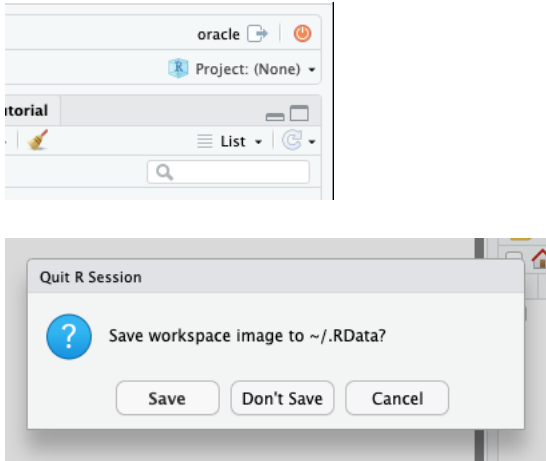
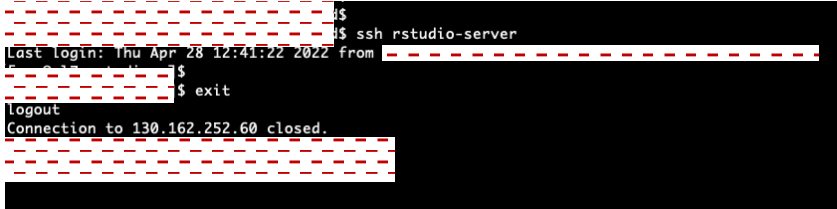
<p>Enter the following:</p> <ul style="list-style-type: none"> Source Type: CIDR Source CIDR: 0.0.0.0/0 IP Protocol: TCP Destination Port: 8787 <p>Click 'Add Ingress Rules'</p>	<h3>Add Ingress Rules</h3> <p>Ingress Rule 1</p> <p>Allows TCP traffic 8787</p> <p><input type="checkbox"/> Stateless ⓘ</p> <p>Source Type: CIDR</p> <p>Source CIDR: 0.0.0.0/0</p> <p>IP Protocol: TCP</p> <p>Specified IP addresses: 0.0.0.0-255.255.255.255 (4,294,967,296 IP addresses)</p> <p>Source Port Range: Optional ⓘ All</p> <p>Destination Port Range: Optional ⓘ 8787</p> <p>Examples: 80, 20-22</p> <p>Description: Optional R Studio Server</p> <p>Maximum 255 characters</p> <p>+ Another Ingress Rule</p> <p>Add Ingress Rules Cancel</p>
<p>On our local machine we will edit the ~/.ssh/config file and add an entry for our SSH tunnel for Port Forwarding.</p> <p>vi ~/.ssh/config</p>	<pre>Host rstudio-server HostName [redacted] User opc IdentityFile /Users/isyed/.ssh/rstudio-server-private.key LocalForward 8787 localhost:8787</pre> <p>Host rstudio-server</p> <p>HostName XXX.XXX.XXX.XX</p> <p>User opc</p> <p>IdentityFile /Users/isyed/.ssh/rstudio-server-private.key</p> <p>LocalForward 8787 localhost:8787</p>
<p>We can then open the SSH Tunnel with Port Forwarding by:</p> <p>ssh rstudio-server</p>	<pre>[redacted]\$ vi ~/.ssh/config [redacted]\$ [redacted]\$ ssh rstudio-server Last login: Thu Apr 28 11:04:31 2022 from [redacted]</pre>



<p>While leaving your SSH Tunnel open, from your local machine start a browser and visit your R Studio Server URL:</p> <p>localhost:8787</p> <p>Login as the oracle user with the password you just set.</p>	 <p>The image shows the RStudio login interface. At the top, it says 'R Studio'. Below that is a 'Sign in to RStudio' form. The form has fields for 'Username:' (with 'oracle' entered) and 'Password:' (with masked characters). There is a checkbox for 'Stay signed in when browser closes' and a note: 'You will automatically be signed out after 60 minutes of inactivity.' A blue 'Sign In' button is at the bottom.</p>
<p>We will now install a couple R Packages (Libraries), one being the R Oracle Package.</p> <p>From the R Console, run the following:</p> <p><code>install.packages("DBI")</code></p> <p><code>install.packages("/tmp/ROracle_1.3-2_R_x86_64-linux-gnu.tar.gz", repos=NULL)</code></p>	<p>You are using Oracle's distribution of R. Please contact Oracle Support for any problems you encounter with this distribution.</p> <pre>> install.packages("DBI") Installing package into '/home/oracle/R/x86_64-pc-linux-gnu-library/3.6' (as 'lib' is unspecified) --2022-04-28 12:41:58-- https://cran.rstudio.com/src/contrib/DBI_1.1.2.tar.gz Resolving cran.rstudio.com (cran.rstudio.com)... 13.224.195.24, 13.224.195.125, 13.224.195.7, Connecting to cran.rstudio.com (cran.rstudio.com) 13.224.195.24 :443... connected. HTTP request sent, awaiting response... 200 OK Length: 706518 (690K) [application/x-gzip] Saving to: '/tmp/Rtmpj0L0jZ/downloaded_packages/DBI_1.1.2.tar.gz' 0K 7% 48.3M 0s OK</pre> <pre>> install.packages("/tmp/ROracle_1.3-2_R_x86_64-linux-gnu.tar.gz", repos=NULL) Installing package into '/home/oracle/R/x86_64-pc-linux-gnu-library/3.6' (as 'lib' is unspecified) * installing *binary* package 'ROracle' ... * DONE (ROracle) ></pre>
<p>After making the above changes we will reboot the VM for the changes to take effect.</p> <p>From within the OCI Console, visit your VM from Menu > Compute > Instance > Your Instance.</p> <p>Click on Reboot.</p>	 <p>The image shows the Oracle Cloud Console interface. At the top, it says 'ORACLE Cloud' and 'Cloud Classic >'. Below that is a search bar. The main content area shows 'Compute > Instances > Instance details'. On the left is a green square icon with a white 'I' and the text 'RUNNING'. On the right, there's a section for 'OL7-RStudio' with buttons for 'Start', 'Stop', 'Reboot' (highlighted with a red box), 'Edit', and 'More Actions'. Below that is a tabbed interface with 'Instance information' selected, showing 'General information' with details like 'Availability domain: AD-1', 'Fault domain: FD-1', 'Region: ...', 'OCID: ...', 'Launched: ...', and 'Compartment: ...'.</p>

<p>Re-Establish the SSH Tunnel and Login via the browser.</p> <p><i>ssh rstudio-server</i></p> <p><i>localhost:8787</i></p> <p><i>Login with the oracle user.</i></p>	
<p>We can then start querying our Autonomous Data Warehouse.</p> <p>Load in R Oracle Library</p> <p>Instantiate Driver</p> <p>Establish Connect</p> <p>Submit Query</p> <p>Print Results</p>	 <pre> > library(ROracle) > > drv <- dbDriver("Oracle") > > con <- dbConnect(drv, dbname=" ", username=" ", password=" ") > > res <- dbGetQuery(con, 'select ._.count(*) as searches from ._. ') > > typeof(res) [1] "list" > > res FORCE SEARCHES 1 Dorset 18188 2 West Yorkshire 79461 3 South Wales 53740 4 North Wales 17539 5 British Transport Police 31125 6 South Yorkshire 43868 7 Bedfordshire 13532 8 Cumbria 12684 9 Dyfed Powys 17364 10 West Midlands 108735 11 Gloucestershire 10559 12 Gwent 14139 13 Saffolk 13408 </pre>
<p>R Code Listed.</p>	<pre> > library(ROracle) > > drv <- dbDriver("Oracle") > > con <- dbConnect(drv, dbname="service_name", username=" username", password="password") > > res <- dbGetQuery(con, 'select 1 from dual') > > typeof(res) > > res </pre>
<p>You can then disconnect from the DB</p> <p><i>dbDisconnect(con)</i></p>	<pre> > > > > > dbDisconnect(con) [1] TRUE > </pre>



<p>You can then disconnect from the R Studio Session using the 'Red Power Icon' in the top right.</p> <p>You will then be prompted to save you workspace if you wish.</p>	
<p>Once done you can close the R Studio browser tab and exit the SSH Tunnel.</p> <p>exit</p>	
<p>Finally, you can shut down your R Studio Serve VM when it is not in use from the OCI Console.</p> <p>Menu > Compute > Instance > Your Instance</p> <p>Click Stop.</p>	