The first thing you have to do is create an SMTP Credential for your user. You do that by logging into your Oracle Cloud account, go to **Identity > Users** and select your user:

<https://blogs.oracle.com/apex/post/sending-email-from-your-oracle-apex-app-on-autonomous-database>

**Note**:

* The instructions in this blog post use the free promotion account that is within the trial period. When your trial is over, your account will be limited to Always Free resources. You need to upgrade to a paid account to continue to use the Email Delivery.
* If Email Delivery is not available in your current region, you may need to subscribe to additional OCI regions.
* The application that is sending email does not have to be in the region where email is sent for delivery.
* There is a limit of 5000 emails per workspace in a 24-hour period.

STARTING POINT

* You can use the [APEX\_MAIL](https://docs.oracle.com/en/database/oracle/application-express/19.2/aeapi/APEX_MAIL.html#GUID-14F51C6D-CB82-4B38-AB6E-61C46E75596F) package to send emails from your Oracle APEX applications deployed in Oracle Autonomous Database (ATP / ADW). However, to use the APEX\_MAIL package, you need to first configure an email service provider in your Oracle APEX instance. Currently, the only supported email service provider is the [Oracle Cloud Infrastructure Email Delivery service](https://docs.cloud.oracle.com/en-us/iaas/Content/Email/Concepts/overview.htm).
* Sending email from your APEX app in Autonomous Database involves the following four steps:

1. Generating Simple Mail Transfer Protocol (SMTP ) Credentials for Email Delivery

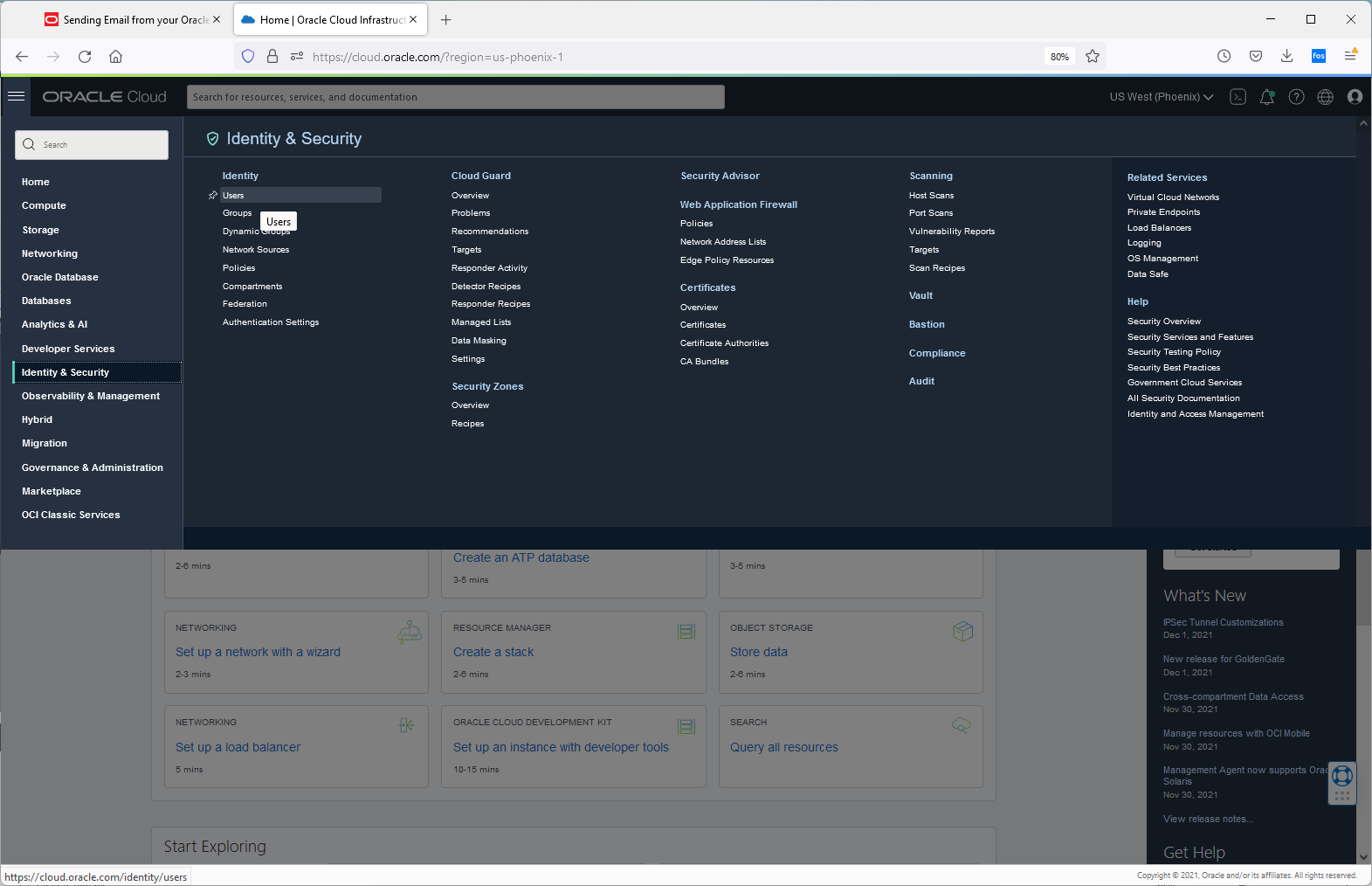
2. Creating an approved sender for Email Delivery

3. Configuring the SMTP parameters in the Application Express instance

4. Sending a test email using Oracle APEX SQL Workshop

**1. Generating SMTP Credentials for Email Delivery**:

SMTP credentials are necessary to send email through Email Delivery service. When you send an email from your APEX application,  the APEX instance uses these SMTP credentials to authenticate with email delivery servers. In this section, you generate SMTP credentials for your user in the OCI console. Perform the following steps:



Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Click on add ot idcs groups and add both. Go back and click on your new user.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Click on generate smtp credentials

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

[ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com](mailto:ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com)

ze0LkWvBU1)u;zqJ!xaJ

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

# ****2. Creating an Approved Sender for Email Delivery****: (in application integration )

Next, you set up an approved sender for Email Delivery.  You must set up an approved sender for all email addresses you use as the "From" with APEX\_MAIL.SEND calls. Perform the following steps:

1. In the left menu, navigate to **Email Delivery > Email Approved Senders**.
2. Graphical user interface, text, application, email

   Description automatically generated
3. Graphical user interface, text, application, email

   Description automatically generated
4. Graphical user interface, text, application, email

   Description automatically generated

ocid1.emailsender.oc1.phx.aaaaaaaa6v2ndcraaueuihp2sdyhdnh2fdvvmqzg5dv6gwyjggs7q3xpwqtq

You need to identify the SMTP connection endpoint for Email Delivery. You will configure this endpoint as SMTP Host in the APEX instance in the next section. Under Email Delivery, click **Email Configuration**. Make sure you copy the SMTP Server Name (for example, smtp.us-phoenix-1.oraclecloud.com) and save it to a text file.

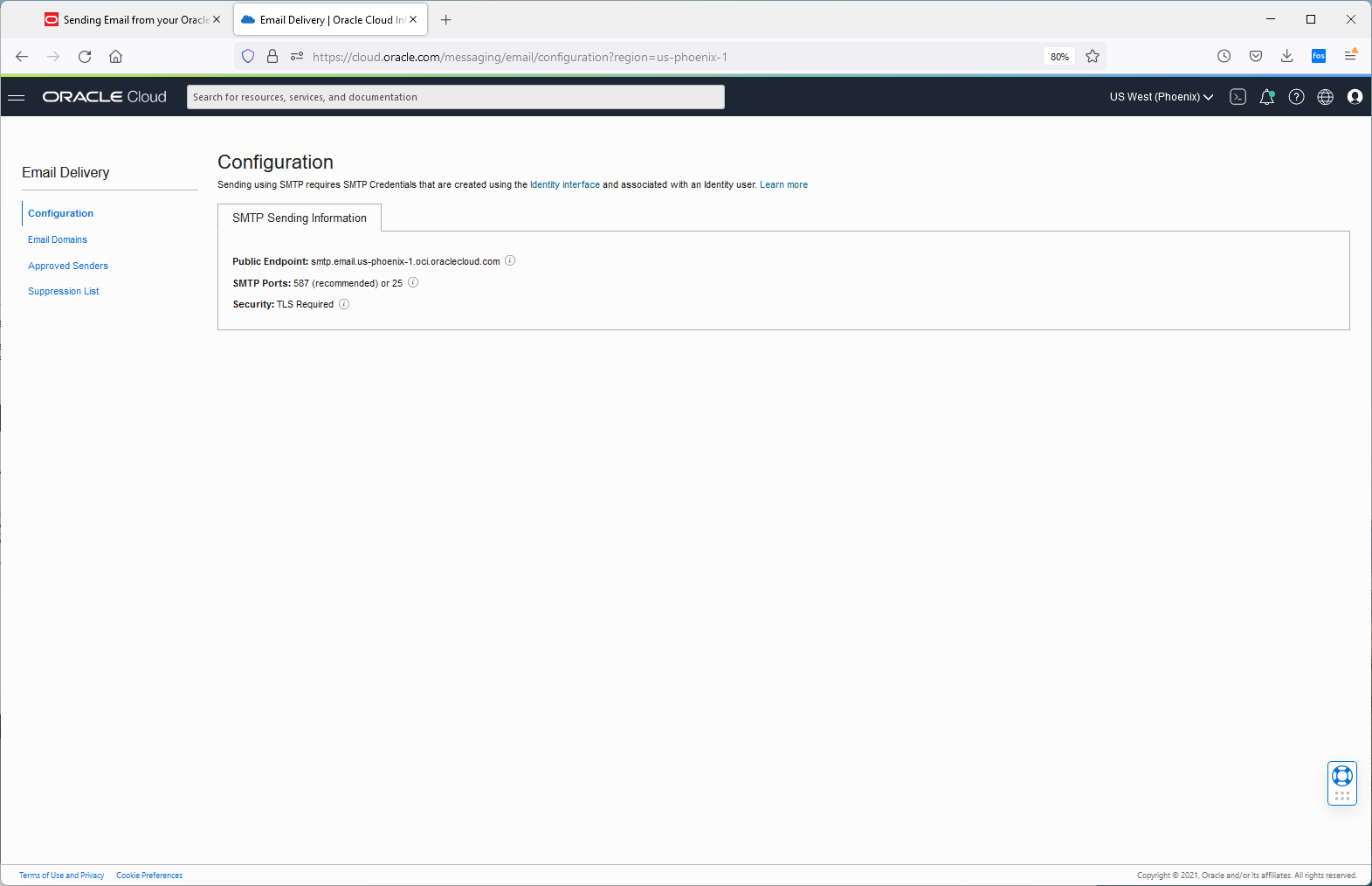
Graphical user interface, text, application

Description automatically generated

**Public Endpoint:** smtp.email.us-phoenix-1.oci.oraclecloud.com

  **SMTP Ports:** 587 (recommended) or 25

  **Security:** TLS Required



smtp.email.us-phoenix-1.oci.oraclecloud.com

# ****Configuring the SMTP Parameters in the Application Express Instance****:

Now that you have the SMTP credentials generated, you need to let your Application Express instance know about these parameters. To do this, you connect to your Autonomous Database as ADMIN user using a SQL client and configure the following SMTP parameters using [APEX\_INSTANCE\_ADMIN.SET\_PARAMETER](https://docs.oracle.com/en/database/oracle/application-express/19.2/aeapi/APEX_INSTANCE_ADMIN.html#GUID-1A894D25-A884-466B-9B88-B10888B2FFEA):

* SMTP\_HOST\_ADDRESS: Specifies the SMTP connection endpoint
* SMTP\_USERNAME: Specifies the SMTP credential user name
* SMTP\_PASSWORD: Specifies the SMTP credential
* Default values for SMTP\_HOST\_PORT parameter (587) and SMTP\_TLS\_MODE parameter (STARTTLS).

<https://blogs.oracle.com/apex/post/sending-email-from-your-oracle-apex-app-on-autonomous-database>

To connect to the Autonomous Database instance, you can use the built-in [SQL Developer Web](https://docs.oracle.com/en/database/oracle/sql-developer-web/19.4/sdweb/about-sdw.html#GUID-AF7601F9-7713-4ECC-8EC9-FB0296002C69). You can access SQL Developer Web either from the OCI console or from the ATP / ADW Service console.

Perform the following steps:

1. From the left menu, navigate to your Autonomous Database instance details page

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

The instance details page displays. Click **Tools** and then click **Open SQL Developer Web**.

Graphical user interface

Description automatically generated

IF sql developer web is not there, goto database actions to find it there.

Graphical user interface, application, Word

Description automatically generated

Click SQL top left

Change schema to miso.

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

**On the far right you should see admin user**

**Ifnot login as admin/Epfy44yqE\_WEB**

BEGIN

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_HOST\_ADDRESS', 'smtp.email.us-phoenix-1.oci.oraclecloud.com'   );

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_USERNAME',     'ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com');

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_PASSWORD',     'ze0LkWvBU1)u;zqJ!xaJ');

COMMIT;

 dbms\_output.put\_line('Done');

exception when others then raise;

END;

/

**4. Sending a Test Email using Oracle APEX SQL Workshop**

Finally,you want to test if the email delivery works. In this step, you navigate to Oracle APEX > SQL Workshop > SQL Commands and send a test email. You specify the approved sender that you created in the "Creating an Approved Sender for Email Delivery" section above. Perform the following steps:

1. From the left menu, navigate to your Autonomous Database instance details page
2. Click the database instance name.
3. The instance details page displays. Click **Tools** and then click **Open APEX**.
4.  Log in to your workspace.
5.  Click **SQL Workshop** and then click **SQL Commands**.
6.  Enter the following code in the SQL Commands area and click **Run**.

BEGIN   
apex\_mail.send(p\_from => 'approved sender email address',   
p\_to => 'recipient email address',   
p\_subj => 'Email from Autonomous',  
p\_body => 'This is a test email from Autonomous');

apex\_mail.push\_queue();   
END;   
/

BEGIN

 apex\_mail.send(p\_from => 'aplascencia@misorobotics.com', p\_to => 'rmander.ca@gmail.com', p\_subj => 'Email from APEX ',p\_body => 'This is a test email from Autonomous');

 apex\_mail.push\_queue();

END;

/

BEGIN

 apex\_mail.send(p\_from => 'aplascencia@misorobotics.com', p\_to => 'rmander.ca@gmail.com', p\_subj => 'Email from APEX ',p\_body => 'This is a test email from Autonomous');

 apex\_mail.push\_queue();

END;

/

 Now, verify if the email has been successfully delivered to the recipient's email address specified in the code snippet above.  
  
Graphical user interface, text, application

Description automatically generated

 You want to monitor the email delivery in your Application Express instance as well. You can simply query the APEX\_MAIL\_LOG and APEX\_MAIL\_QUEUE views.

A screenshot of a computer

Description automatically generated

1. You want to monitor the email delivery in your Application Express instance as well. You can simply query the APEX\_MAIL\_LOG and APEX\_MAIL\_QUEUE views.  
   **Note**: Alternatively, you can monitor the email delivery in the APEX instance administration. To do this, sign in to Administration Services and click **Manage Instance**. Under Manage Meta Data, click **Mail Queue**.  
     
   Graphical user interface, application, Word

   Description automatically generated

And so now, you are all ready to send emails from your Oracle APEX applications deployed in the Autonomous Database!

If no email then goto admin internal and update instance config

<https://gd49ef7d5e3cd2d-db202110212330.adb.us-phoenix-1.oraclecloudapps.com/ords>

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Require HTTPS

Options include:

* **Always**: Enforces HTTPS for all applications (including the Application Express development and administration applications) to require HTTPS. If set to Always, the Strict-Transport-Security Max Age attribute displays. Use this field to specify the time period in seconds during which the browser shall access the server with HTTPS only.
* **Development and Administration**: Forces all internal applications within Application Express (that is, App Builder, SQL Workshop, Instance Administration and so on) to require HTTPS.
* **Application specific**: Makes HTTPS dependent on the "Secure" session cookie attribute in the authentication schemes of your applications.

A screenshot of a computer

Description automatically generated with medium confidence

**Ifnot login as admin/Epfy44yqE\_WEB**

BEGIN

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_HOST\_ADDRESS',

'smtp.email.us-phoenix-1.oci.oraclecloud.com'   );

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_USERNAME',

'ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com');

 APEX\_INSTANCE\_ADMIN.SET\_PARAMETER('SMTP\_PASSWORD',

'ze0LkWvBU1)u;zqJ!xaJ');

ze0LkWvBU1)u;zqJ!xaJ

COMMIT;

 dbms\_output.put\_line('Done');

exception when others then raise;

END;

/

A screenshot of a computer

Description automatically generated

**Public Endpoint:** smtp.email.us-phoenix-1.oci.oraclecloud.com

  **SMTP Ports:** 587 (recommended) or 25

  **Security:** TLS Required

select mail\_subj, mail\_From, mail\_message\_created, mail\_send\_error from apex\_mail\_queue

zero in the queue

<http://dgielis.blogspot.com/2019/10/free-oracle-cloud-11-sending-emails.html>

says we have to subscribe

select mail\_subj, mail\_From, mail\_message\_created, mail\_send\_error from apex\_mail\_queue;

select \* from apex\_mail\_log;

BEGIN  
DBMS\_NETWORK\_ACL\_ADMIN.APPEND\_HOST\_ACE(  
host => '\*',  
ace => xs$ace\_type(privilege\_list => xs$name\_list('connect’, 'resolve','smtp'),  
principal\_name => 'APEX\_190100',  
principal\_type => xs\_acl.ptype\_db));  
END;

Graphical user interface, text, application, email

Description automatically generated

<https://docs.oracle.com/en-us/iaas/Content/Email/Concepts/overview.htm>

## Getting Started with Email Delivery

You can set up the Email Delivery service within the Console. To begin sending email with Email Delivery, complete the following steps:

1. [Generate SMTP credentials for a user.](https://docs.oracle.com/en-us/iaas/Content/Email/Tasks/generatesmtpcredentials.htm#Generate_SMTP_Credentials_for_a_User)
2. [Set up permissions.](https://docs.oracle.com/en-us/iaas/Content/Email/Reference/gettingstarted.htm#permissions)
3. [Create an approved sender.](https://docs.oracle.com/en-us/iaas/Content/Email/Tasks/managingapprovedsenders.htm#Managing_Approved_Senders)
4. [Configure SPF on the approved sender domain.](https://docs.oracle.com/en-us/iaas/Content/Email/Tasks/configurespf.htm#Configure_SPF)
5. [Configure the SMTP connection.](https://docs.oracle.com/en-us/iaas/Content/Email/Tasks/configuresmtpconnection.htm#Configure_SMTP_Connection)
6. [Begin sending email.](https://docs.oracle.com/en-us/iaas/Content/Email/Reference/gettingstarted.htm#send)

For more information, see [Getting Started with Email Delivery](https://docs.oracle.com/en-us/iaas/Content/Email/Reference/gettingstarted.htm#Getting_Started_with_Email_Delivery).

A screenshot of a computer

Description automatically generated

Graphical user interface, application

Description automatically generated

Click create policy

Graphical user interface, text, application, email

Description automatically generated

## TLS Requirements

Oracle maintains strict security policies and only accepts email traffic using Transport Layer Security (TLS). Use of TLS 1.2 is mandatory to send email using Oracle Cloud Infrastructure.

A screenshot of a computer

Description automatically generated

select \*

from ALL\_SCHEDULER\_JOB\_RUN\_DETAILS

where job\_name = 'ORACLE\_APEX\_MAIL\_QUEUE'

order by log\_date desc;

declare

    l\_body      clob;

    l\_body\_html clob;

begin

    l\_body := 'To armando , view the content of this message, please use an HTML enabled mail client.' || utl\_tcp.crlf;

    l\_body\_html := '<html><body>

                   <p>Welcome armando</p>

                   </body></html>';

    apex\_mail.send (

        p\_to        => 'rmander.ca@gmail.com',

        p\_from      => 'aplascencia@misorobotics.com',

        p\_body      => l\_body,

        p\_body\_html => l\_body\_html,

        p\_subj      => 'Welcome Email' );

end;

/

commit;

Nothing Scheduled so I run it manuyally

begin

dbms\_scheduler.run\_job('APEX\_210100.ORACLE\_APEX\_MAIL\_QUEUE') ;

end;

commit;

select mail\_subj, mail\_From, mail\_message\_created, mail\_send\_error , last\_updated\_on , mail\_message\_created from apex\_mail\_queue;

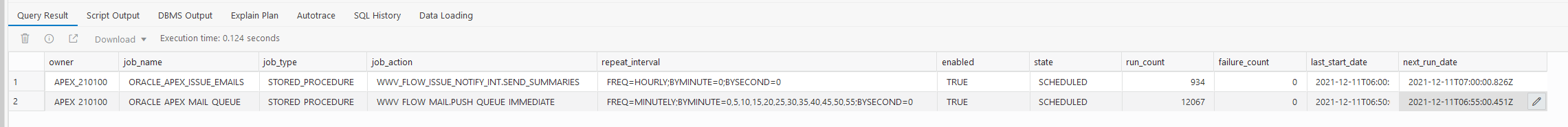
"OWNER" "JOB\_NAME" "JOB\_SUBNAME" "JOB\_STYLE" "JOB\_CREATOR" "CLIENT\_ID" "GLOBAL\_UID" "PROGRAM\_OWNER" "PROGRAM\_NAME" "JOB\_TYPE" "JOB\_ACTION" "NUMBER\_OF\_ARGUMENTS" "SCHEDULE\_OWNER" "SCHEDULE\_NAME" "SCHEDULE\_TYPE" "START\_DATE" "REPEAT\_INTERVAL" "EVENT\_QUEUE\_OWNER" "EVENT\_QUEUE\_NAME" "EVENT\_QUEUE\_AGENT" "EVENT\_CONDITION" "EVENT\_RULE" "FILE\_WATCHER\_OWNER" "FILE\_WATCHER\_NAME" "END\_DATE" "JOB\_CLASS" "ENABLED" "AUTO\_DROP" "RESTART\_ON\_RECOVERY" "RESTART\_ON\_FAILURE" "STATE" "JOB\_PRIORITY" "RUN\_COUNT" "UPTIME\_RUN\_COUNT" "MAX\_RUNS" "FAILURE\_COUNT" "UPTIME\_FAILURE\_COUNT" "MAX\_FAILURES" "RETRY\_COUNT" "LAST\_START\_DATE" "LAST\_RUN\_DURATION" "NEXT\_RUN\_DATE" "SCHEDULE\_LIMIT" "MAX\_RUN\_DURATION" "LOGGING\_LEVEL" "STORE\_OUTPUT" "STOP\_ON\_WINDOW\_CLOSE" "INSTANCE\_STICKINESS" "RAISE\_EVENTS" "SYSTEM" "JOB\_WEIGHT" "NLS\_ENV" "SOURCE" "NUMBER\_OF\_DESTINATIONS" "DESTINATION\_OWNER" "DESTINATION" "CREDENTIAL\_OWNER" "CREDENTIAL\_NAME" "INSTANCE\_ID" "DEFERRED\_DROP" "ALLOW\_RUNS\_IN\_RESTRICTED\_MODE" "COMMENTS" "FLAGS" "RESTARTABLE" "HAS\_CONSTRAINTS" "CONNECT\_CREDENTIAL\_OWNER" "CONNECT\_CREDENTIAL\_NAME" "FAIL\_ON\_SCRIPT\_ERROR"

"APEX\_210100" "ORACLE\_APEX\_ISSUE\_EMAILS" "" "REGULAR" "SYS" "" "" "" "" "STORED\_PROCEDURE" "WWV\_FLOW\_ISSUE\_NOTIFY\_INT.SEND\_SUMMARIES" 0 "" "" "CALENDAR" 2021-10-14 02:42:10.000000000 GMT "FREQ=HOURLY;BYMINUTE=0;BYSECOND=0" "" "" "" "" "" "" "" "DEFAULT\_JOB\_CLASS" "TRUE" "FALSE" "FALSE" "FALSE" "SCHEDULED" 3 929 0 0 2021-12-11 01:00:04.800409000 GMT +00 00:00:00.474206 2021-12-11 02:00:00.802536000 GMT "OFF" "TRUE" "FALSE" "TRUE" "" "FALSE" 1 "NLS\_LANGUAGE='AMERICAN' NLS\_TERRITORY='AMERICA' NLS\_CURRENCY='$' NLS\_ISO\_CURRENCY='AMERICA' NLS\_NUMERIC\_CHARACTERS='.,' NLS\_CALENDAR='GREGORIAN' NLS\_DATE\_FORMAT='DD-MON-RR' NLS\_DATE\_LANGUAGE='AMERICAN' NLS\_SORT='BINARY' NLS\_TIME\_FORMAT='HH.MI.SSXFF AM' NLS\_TIMESTAMP\_FORMAT='DD-MON-RR HH.MI.SSXFF AM' NLS\_TIME\_TZ\_FORMAT='HH.MI.SSXFF AM TZR' NLS\_TIMESTAMP\_TZ\_FORMAT='DD-MON-RR HH.MI.SSXFF AM TZR' NLS\_DUAL\_CURRENCY='$' NLS\_COMP='BINARY' NLS\_LENGTH\_SEMANTICS='BYTE' NLS\_NCHAR\_CONV\_EXCP='FALSE'" "" 1 "" "" "" "" "FALSE" "FALSE" "" 9007199255038000 "FALSE" "FALSE" "" "" "FALSE"

"APEX\_210100" "ORACLE\_APEX\_MAIL\_QUEUE" "" "REGULAR" "SYS" "" "" "" "" "STORED\_PROCEDURE" "WWV\_FLOW\_MAIL.PUSH\_QUEUE\_IMMEDIATE" 0 "" "" "CALENDAR" 2021-10-14 02:42:10.000000000 GMT "FREQ=MINUTELY;BYMINUTE=0,5,10,15,20,25,30,35,40,45,50,55;BYSECOND=0" "" "" "" "" "" "" "" "DEFAULT\_JOB\_CLASS" "TRUE" "FALSE" "FALSE" "FALSE" "SCHEDULED" 3 12002 0 0 2021-12-11 01:50:01.996345000 GMT +00 00:00:00.035619 2021-12-11 01:55:00.999081000 GMT "OFF" "TRUE" "FALSE" "TRUE" "" "FALSE" 1 "NLS\_LANGUAGE='AMERICAN' NLS\_TERRITORY='AMERICA' NLS\_CURRENCY='$' NLS\_ISO\_CURRENCY='AMERICA' NLS\_NUMERIC\_CHARACTERS='.,' NLS\_CALENDAR='GREGORIAN' NLS\_DATE\_FORMAT='DD-MON-RR' NLS\_DATE\_LANGUAGE='AMERICAN' NLS\_SORT='BINARY' NLS\_TIME\_FORMAT='HH.MI.SSXFF AM' NLS\_TIMESTAMP\_FORMAT='DD-MON-RR HH.MI.SSXFF AM' NLS\_TIME\_TZ\_FORMAT='HH.MI.SSXFF AM TZR' NLS\_TIMESTAMP\_TZ\_FORMAT='DD-MON-RR HH.MI.SSXFF AM TZR' NLS\_DUAL\_CURRENCY='$' NLS\_COMP='BINARY' NLS\_LENGTH\_SEMANTICS='BYTE' NLS\_NCHAR\_CONV\_EXCP='FALSE'" "" 1 "" "" "" "" "FALSE" "FALSE" "" 9007199255038000 "FALSE" "FALSE" "" "" "FALSE"

I have logged in as the admin user to look at the job queues if you look that the above plsql block, the data says in the mail queue and does not leave.

Please help with this.



select \*  from dba\_scheduler\_jobs where job\_name like '%MAIL%' and enabled = 'FALSE' order by job\_name;

zero

select owner, job\_name,job\_type,job\_action,repeat\_interval, enabled, state,run\_count, failure\_count,last\_start\_Date, next\_run\_Date from dba\_scheduler\_jobs where job\_name like '%MAIL%' and enabled <> 'FALSE' order by job\_name;

 declare

    l\_body      clob;

    l\_body\_html clob;

begin

    wwv\_flow\_api.set\_security\_group\_id;

    l\_body := 'To qqqqqq armando , view the content of this message, please use an HTML enabled mail client.' || utl\_tcp.crlf;

    l\_body\_html := '<html><body>

                   <p>Welcome qqqqqqarmando</p>

                   </body></html>';

    apex\_mail.send (

        p\_to        => 'rmander.ca@gmail.com',

        p\_from      => 'aplascencia@misorobotics.com',

        p\_body      => l\_body,

        p\_body\_html => l\_body\_html,

        p\_subj      => 'QQQQQQQQQQQWelcome Email' );

apex\_mail.push\_queue();

end;

/

commit;

A screenshot of a computer

Description automatically generated

begin

dbms\_network\_acl\_admin.append\_host\_ace(

host =>'smtp.email.us-phoenix-1.oci.oraclecloud.com', lower\_port => 587, upper\_port => 587,

ace => xs$ace\_type( privilege\_list => xs$name\_list('SMTP'), principal\_name => 'ADMIN',principal\_type => xs\_acl.ptype\_db));

Commit;

end;

/

begin

dbms\_network\_acl\_admin.append\_host\_ace(

host =>'smtp.email.us-phoenix-1.oci.oraclecloud.com', lower\_port => 587, upper\_port => 587,

ace => xs$ace\_type( privilege\_list => xs$name\_list('SMTP'), principal\_name => 'MISO',principal\_type => xs\_acl.ptype\_db));

Commit;

end;

/

begin

dbms\_network\_acl\_admin.append\_host\_ace(

host =>'smtp.email.us-phoenix-1.oci.oraclecloud.com', lower\_port => 587, upper\_port => 587,

ace => xs$ace\_type( privilege\_list => xs$name\_list('SMTP'), principal\_name => 'APEX\_210100',principal\_type => xs\_acl.ptype\_db));

Commit;

end;

/

[ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com](mailto:ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com)

https://docs.oracle.com/en-us/iaas/Content/Email/Concepts/overview.htm

smtp.email.us-phoenix-1.oci.oraclecloud.com

CREATE OR REPLACE PROCEDURE MISO\_AP\_SEND\_MAIL (

  msg\_to varchar2,

  msg\_subject varchar2,

  msg\_text varchar2 )

IS

  mail\_conn utl\_smtp.connection;

  username varchar2(1000) := 'ocid1.user.oc1..aaaaaaaahr7oxhekztltzlswuium6k6heb6dqsyoag2agf5cyztqtt2a7pea@ocid1.tenancy.oc1..aaaaaaaauzqaexjmy5i7nrrbttk4czh42lzvehrr4us3oayfa45e6vcnofga.hc.com';

  passwd   varchar2(500)  := 'ze0LkWvBU1)u;zqJ!xaJ';

  msg\_from varchar2(500)   := 'aplascencia@misorobotics.com';

  mailhost VARCHAR2(500)   := 'smtp.email.us-phoenix-1.oci.oraclecloud.com';

BEGIN

  mail\_conn := UTL\_smtp.open\_connection(mailhost, 587);

  utl\_smtp.starttls(mail\_conn);

  UTL\_SMTP.AUTH(mail\_conn, username, passwd, schemes => 'PLAIN');

  utl\_smtp.mail(mail\_conn, msg\_from);

  utl\_smtp.rcpt(mail\_conn, msg\_to);

  UTL\_smtp.open\_data(mail\_conn);

  UTL\_SMTP.write\_data(mail\_conn, 'Date: ' || TO\_CHAR(SYSDATE, 'DD-MON-YYYY HH24:MI:SS') || UTL\_TCP.crlf);

  UTL\_SMTP.write\_data(mail\_conn, 'To: ' || msg\_to || UTL\_TCP.crlf);

  UTL\_SMTP.write\_data(mail\_conn, 'From: ' || msg\_from || UTL\_TCP.crlf);

  UTL\_SMTP.write\_data(mail\_conn, 'Subject: ' || msg\_subject || UTL\_TCP.crlf);

  UTL\_SMTP.write\_data(mail\_conn, 'Reply-To: ' || msg\_to || UTL\_TCP.crlf || UTL\_TCP.crlf);

  UTL\_SMTP.write\_data(mail\_conn, msg\_text || UTL\_TCP.crlf || UTL\_TCP.crlf);

  UTL\_smtp.close\_data(mail\_conn);

  UTL\_smtp.quit(mail\_conn);

EXCEPTION

  WHEN UTL\_smtp.transient\_error OR UTL\_smtp.permanent\_error THEN

    UTL\_smtp.quit(mail\_conn);

    dbms\_output.put\_line(sqlerrm);

  WHEN OTHERS THEN

    UTL\_smtp.quit(mail\_conn);

    dbms\_output.put\_line(sqlerrm);

END;

/

begin

 MISO\_AP\_SEND\_MAIL ( 'rmander.ca@gmail.com','test email from utl\_smtp','body of email msg');

end;

begin

apex\_mail.send(

p\_to => 'rmander.ca@gmail.com',

p\_from => 'aplascencia@misorobotics.com',

p\_subj => 'apex mail frustration',

p\_body => 'Please review the email in html.',

p\_body\_html => 'Here it is ');

apex\_mail.push\_queue;

end;