

DBMS_CLOUD_AI

The DBMS_CLOUD_AI package enables you to develop applications that use large language models easily from SQL and PL/SQL. It supports configuring access to AI providers as well as tables and views from your database. Using natural language prompts, generate, run, explain, and narrate SQL queries. Also, chat directly with LLMs from SQL and PL/SQL.

DBMS_CLOUD_AI Overview

Describes the use of the DBMS_CLOUD_AI package.

Use the DBMS_CLOUD_AI package to create AI profiles and configure them for access to a Large Language Model (LLM). Set the AI profile in the current database user session to perform tasks such as generating, running, and explaining SQL.

Summary of DBMS_CLOUD_AI Subprograms

This section covers the DBMS_CLOUD_AI subprograms provided with Oracle Database.

Subprogram	Description
CREATE_PROFILE Procedure	This procedure creates a new AI profile for specifying your AI provider, large language model, and other attributes. See the set of AI Profile Attributes .
DISABLE_PROFILE Procedure	This procedure disables an AI profile in the current database.
DROP_PROFILE Procedure	This procedure drops an existing AI profile.
ENABLE_PROFILE Procedure	This procedure enables an AI profile to use in the current database.
GENERATE Function	This function sends the user prompt to the specified AI profile using the specified 'action' - enabling stateless invocation of Select AI.
GET_PROFILE Function	This function returns the profile name used in the current session.
GET_PROFILE Procedure	This procedure returns the profile name and the owner of the profile in the current session.
SET_ATTRIBUTE Procedure	This procedure sets AI profile attributes.

CREATE_PROFILE Procedure

This procedure creates a new AI profile for specifying your AI provider, large language model, and other attributes.

Syntax

```
DBMS_CLOUD_AI.CREATE_PROFILE
    profile_name          IN  VARCHAR2,
```

```

attributes      IN  CLOB      DEFAULT NULL,
status          IN  VARCHAR2  DEFAULT NULL,
description     IN  CLOB      DEFAULT NULL
);

```

Parameters

Parameter	Description
profile_name	A name for the AI profile. The profile name must follow the naming rules of Oracle SQL identifier. Maximum length of profile name is 125 characters. This is a mandatory parameter.
attributes	Profile attributes in JSON format. See AI Profile Attributes for more details. The default value is NULL.
status	Status of the profile. The default value is enabled.
description	Description for the AI profile. The default value is NULL.

Example

```

BEGIN
    DBMS_CLOUD_AI.CREATE_PROFILE(
        profile_name => 'OpenAI',
        attributes   => JSON_OBJECT('provider' value 'openai',
                                   'credential_name' value
'openai_cred'),
        status       => 'enabled',
        description  => 'AI profile to use OpenAI for SQL translation'
    );
END;
/

```

DROP_PROFILE Procedure

The procedure drops an existing AI profile. If the profile does not exist, then the procedure throws an error.

Syntax

```

DBMS_CLOUD_AI.DROP_PROFILE(
    profile_name  IN  VARCHAR2,
    force         IN  BOOLEAN DEFAULT FALSE
);

```

Parameters

Parameter	Description
profile_name	Name of the AI profile

Parameter	Description
<code>force</code>	If <code>TRUE</code> , then the procedure ignores errors if AI profile does not exist. The default value for this parameter is <code>FALSE</code> .

Example

```
BEGIN
    DBMS_CLOUD_AI.DROP_PROFILE(profile_name => 'OPENAI');
END;
/
```

Usage Notes

Use `force` to drop a profile and ignore errors if AI profile does not exist.

ENABLE_PROFILE Procedure

This procedure enables the AI profile that the user specifies. The procedure changes the status of the AI profile to `ENABLED`.

Syntax

```
DBMS_CLOUD_AI.ENABLE_PROFILE(
    profile_name          IN   VARCHAR2
);
```

Parameters

Parameter	Description
<code>profile_name</code>	Name for the AI profile to enable This parameter is mandatory.

Example to Enable AI Profile

```
BEGIN
    DBMS_CLOUD_AI.ENABLE_PROFILE(
        profile_name      => 'OPENAI'
    );
END;
/
```

DISABLE_PROFILE Procedure

This procedure disables the AI profile in the current database. The status of the AI profile is changed to `DISABLED` by this procedure.

Syntax

```
DBMS_CLOUD_AI.DISABLE_PROFILE(  
    profile_name IN VARCHAR2  
);
```

Parameters

Parameter	Description
profile_name	Name for the AI profile. This parameter is mandatory.

Example

```
BEGIN  
    DBMS_CLOUD_AI.DISABLE_PROFILE(  
        profile_name => 'OPENAI'  
    );  
END;  
/
```

GET_PROFILE Function

This function returns the AI profile name set in the current session.

Syntax

```
DBMS_CLOUD_AI.GET_PROFILE(  
    profile_name IN VARCHAR2  
);
```

Parameters

Parameter	Description
profile_name	A name for the AI profile in the current session. This parameter is mandatory.

Example

This example shows how you can display the name of the profile in the current session.

```
SELECT DBMS_CLOUD_AI.GET_PROFILE  
from DUAL;
```

GET_PROFILE Procedure

This procedure returns the AI profile name and the owner set in the current session.

Syntax

```
DBMS_CLOUD_AI.GET_PROFILE(

    profile_name OUT  VARCHAR2,
    profile_owner OUT  VARCHAR2
);
```

Parameters

Parameter	Description
profile_name	A name for the AI profile in the current session. This parameter is mandatory.
profile_owner	Identifies the owner of the AI profile in the current session.

Example

This example shows how you can display the name and owner of the profile in the current session.

```
DECLARE
    l_profile_name DBMS_ID;
    l_profile_owner DBMS_ID;
BEGIN
    DBMS_CLOUD_AI.GET_PROFILE(profile_name => l_profile_name,
                             profile_owner => l_profile_owner);
END;
```

SET_ATTRIBUTE Procedure

This procedure enables you to set AI profile attributes.

Syntax

```
DBMS_CLOUD_AI.SET_ATTRIBUTE(
    profile_name      IN  VARCHAR2,
    attribute_name     IN  VARCHAR2,
    attribute_value    IN  CLOB
);
```

Parameters

Only the owner can set or modify the attributes of the AI profile. For a list of supported attributes, see [Profile Attributes](#).

Parameter	Description
profile_name	Name of the AI profile for which you want to set the attributes. This parameter is mandatory.
attribute_name	Name of the AI profile attribute This parameter is mandatory.

Parameter	Description
attribute_value	Value of the profile attribute. The default value is NULL.

Example

```
BEGIN
  DBMS_CLOUD_AI.SET_ATTRIBUTE(
    profile_name    => 'OPENAI',
    attribute_name  => 'credential_name',
    attribute_value => 'OPENAI_CRED_NEW'
  );
END;
/
```

SET_PROFILE Procedure

This procedure sets the specified AI profile for current session.

After setting an AI profile using a stateful database session, any SQL statement with the prefix `SELECT AI` is considered a natural language prompt. Depending on the action specified with the `AI` prefix, a response is generated using AI. Optionally, it is possible to override the profile attributes or modify attributes by specifying them in JSON format. See [SET_ATTRIBUTE Procedure](#) for setting the attributes.

The AI profile can only be set for current session if the owner of the AI profile is the session user.

To set an AI profile for all sessions of a specific database user or all user sessions in the database, consider using a database event trigger for `AFTER LOGON` event on the specific user or the entire database.

Syntax

```
DBMS_CLOUD_AI.SET_PROFILE(
  profile_name      IN  VARCHAR2,
);
```

Parameters

Parameter	Description
profile_name	A name for the AI profile in the current session. This parameter is mandatory.

Example

```
BEGIN
  DBMS_CLOUD_AI.SET_PROFILE(
    profile_name    => 'OPENAI'
  );
END;
```

```
END;  
/
```

GENERATE Function

This function sends the user prompt to the specified AI profile using the specified 'action' - enabling stateless invocation of Select AI. With your existing AI profile, you can use this function to perform the supported actions `showsql`, `narrate`, or `chat`. The default action is `showsql`.



Note:

`runsql` and `explainsql` are not supported.

Overriding some or all of the profile attributes is also possible using this function.

Syntax

```
DBMS_CLOUD_AI.GENERATE(  
    prompt          IN  CLOB,  
    profile_name    IN  VARCHAR2 DEFAULT NULL,  
    action          IN  VARCHAR2 DEFAULT NULL,  
    attributes      IN  CLOB      DEFAULT NULL  
) RETURN CLOB;
```

Parameters

Parameter	Description
<code>prompt</code>	<p>Natural language prompt to translate using AI.</p> <p>The prompt can include <code>SELECT AI <action></code> as the prefix. The action can also be supplied separately as an "action" parameter. The <i>action</i> supplied in prompt overrides the "action" parameter. Default action is <code>showsql</code>.</p> <p>This parameter is mandatory.</p>

Parameter	Description
<code>profile_name</code>	<p>Name of the AI profile. This parameter is optional if an AI profile is already set in the session using <code>DBMS_CLOUD_AI.SET_PROFILE</code>.</p> <p>The default value is NULL.</p> <p>The following conditions apply:</p> <ul style="list-style-type: none"> • If a profile is set in the current session, the user may omit <code>profile_name</code> argument in the <code>DBMS_CLOUD_AI.GENERATE</code> function. • If the <code>profile_name</code> argument is supplied in the <code>DBMS_CLOUD_AI.GENERATE</code> function, it overrides any value set in the session using the <code>DBMS_CLOUD_AI.SET_PROFILE</code> procedure. • If there is no profile set in the session using the <code>DBMS_CLOUD_AI.SET_PROFILE</code> procedure, the <code>profile_name</code> argument must be supplied in the <code>DBMS_CLOUD_AI.GENERATE</code> function.



Note:

For Database Actions, you can either specify `profile_name` argument in `DBMS_CLOUD_AI.GENERATE` or you can run two steps as a PL/SQL script: `DBMS_CLOUD_AI.SET_PROFILE` and `DBMS_CLOUD_AI.GENERATE`.

```
EXEC
DBMS_CLOUD_AI.set_profile('OPENAI')
;
```

```
-----
-----
```

```
SELECT
DBMS_CLOUD_AI.GENERATE(prompt
=> 'how many customers',

profile_name => 'OPENAI',

action      => 'showsql')
FROM dual;
```

```
-----
-----
```


```
SELECT
DBMS_CLOUD_AI.GENERATE(prompt
=> 'how many customers',

profile_name => 'OPENAI',

action      => 'narrate')
FROM dual;
```

```
-----
-----
```

```
SELECT
```


Parameter	Description
	<pre>DBMS_CLOUD_AI.GENERATE(prompt => 'what is oracle autonomous database', profile_name => 'OPENAI', action => 'chat') FROM dual;</pre>
action	<p>Action for translating natural language prompt using AI. The supported actions include <code>showsql</code> (default), <code>narrate</code>, and <code>chat</code>.</p> <div>  Note: <p>This function does not support the <code>runsql</code> action. If you supply the <code>runsql</code> action, it returns the following error:</p> <pre>ORA-20000: runsql action is not supported by generate function ORA-06512: at "C##CLOUD\$SERVICE.DBMS_CLOUD", line xxxx ORA-06512: at "C##CLOUD\$SERVICE.DBMS_CLOUD_AI", line 2696 ORA-06512: at line x</pre> </div>
attributes	<p>Override specific AI profile attributes by supplying attributes in JSON format. See Profile Attributes for more details.</p>

Examples

The following examples illustrate `showsql`, `narrate`, and `chat` actions that can be used with the `DBMS_CLOUD_AI.GENERATE` function.

An example with `showsql` action is as follows:

```
SELECT DBMS_CLOUD_AI.GENERATE(prompt      => 'how many customers',
                             profile_name => 'OPENAI',
                             action      => 'showsql')
FROM dual;
```

An example with `narrate` action is as follows:

```
SELECT DBMS_CLOUD_AI.GENERATE(prompt      => 'how many customers',
                             profile_name => 'OPENAI',
```

```

                                action      => 'narrate')
FROM dual;

```

An example with `chat` action is as follows:

```

SELECT DBMS_CLOUD_AI.GENERATE(prompt      => 'what is oracle autonomous
database',
                                profile_name => 'OPENAI',
                                action      => 'chat')
FROM dual;

```

You can use `DBMS_CLOUD_AI.GENERATE` in a procedure and run the function. The following example takes an `ai_prompt`, `profile_name`, and `action` as input parameters and calls `DBMS_CLOUD_AI.GENERATE`

```

create or replace FUNCTION call_select_ai (ai_prompt  IN VARCHAR2,
                                           ai_profile IN VARCHAR2,
                                           ai_action  IN VARCHAR2) -- valid
for 'chat', 'narrate', 'showsql'
RETURN CLOB AS sai_resp clob;
BEGIN
    sai_resp := DBMS_CLOUD_AI.GENERATE(prompt      => ai_prompt,
                                       profile_name => ai_profile,
                                       action      => ai_action);

    return(sai_resp);
END call_select_ai;


```

Profile Attributes

Attributes of an AI profile help to manage and configure the behavior of the AI profile. Some attributes are optional and have a default value.

Attributes

Attribute Name	Description
<code>azure_deployment_name</code>	Name of the Azure OpenAI Service deployed model. The name can only include alphanumeric characters, underscore character (<code>_</code>) and a hyphen (<code>-</code>) character. The name cannot end with an underscore (<code>_</code>) or a hyphen (<code>-</code>). To know how to get the <code>azure_deployment_name</code> , see Create and deploy an Azure OpenAI Service resource .
<code>azure_embedding_deployment_name</code>	Name of the Azure OpenAI deployed embedding model. The name can only include alphanumeric characters, underscore, and hyphen. The name can't start or end with a hyphen or underscore.
<code>azure_resource_name</code>	Name of the Azure OpenAI Service resource. The resource name can only include alphanumeric characters and hyphens, and can't start or end with a hyphen. To know how to get the <code>azure_resource_name</code> , see Create and deploy an Azure OpenAI Service resource .

Attribute Name	Description
comments	Include column comments in the metadata used for translating natural language prompts using AI. <code>BOOLEAN</code> datatype is supported. The valid values are <code>TRUE</code> or <code>FALSE</code> for a string with <code>VARCHAR2</code> datatype.
	<div> Note: Boolean values are not applicable in the <code>DBMS_CLOUD_AI.SET_ATTRIBUTE</code> procedure when setting a single attribute because <code>attribute_value</code> parameter is of <code>CLOB</code> data type.</div>
conversation	A <code>VARCHAR2</code> attribute that indicates if conversation history is enabled for a profile. Valid values are <code>true</code> or <code>false</code> . The default value is <code>false</code> . The values are not case sensitive.
credential_name	The name of the credential to access the AI provider APIs. Credential using bearer tokens can be created by using the provider name as the user name and bearer token as the password. Vault Secret credentials are also supported. Principal authentication, for example, Azure service principal, is also supported. This is a mandatory attribute. See CREATE_CREDENTIAL Procedure .
max_tokens	Denotes the number of tokens to predict per generation. Default is 1024. See Tokens and Tokenizers for more details.

Attribute Name	Description
model	The name of the AI model being used to generate responses.


**Note:**


- **Cohere:** [Custom models](#) can be supplied with their full ID.
- **OCI Generative AI:** The Chat Models are supported for all Select AI actions such as `runsql`, `showsql`, `explainsql`, `narrate`, and `chat`.

Select AI supports pretrained models for OCI Generative AI. Custom models can also be supplied with their full OCIDs. If you are supplying OCID or `oci_endpoint_id`, be sure to provide either `oci_runtime_type` or `oci_api_format` depending on the OCI Chat or Generate Text models.

To know more about supported models in OCI Generative AI, see [Pretrained Foundational Models in Generative AI](#).

- This parameter is not used for Azure as the model is determined when you create your deployment in the Azure OpenAI Service portal.

Attribute Name	Description
object_list	<p>Array of JSON objects specifying the owner and object names that are eligible for natural language translation to SQL. To include all objects of a given user, omit the "name" and only specify the "owner" key in the JSON object.</p> <p>The following types of objects can be used:</p> <ul style="list-style-type: none">• tables• views• materialized views• global temporary tables• external tables• synonyms on the above object types <p>For translation natural language to SQL, the object name, object owner, object columns and comments are sent to the AI provider using HTTPS requests. Avoid specifying objects with sensitive object name, column names or comments in the object list.</p> <p>AI providers may have limit on the size of metadata allowed in translation requests. Consider limiting the list of objects suitable for the natural language prompts by your application users.</p> <p>Format:</p> <pre>[{"owner": "SH", "name": "SALES", {"owner": "TEST_USER"}]</pre> <p>External tables created using sync of OCI Data Catalog or AWS Glue can also be used the object list. This helps in managing metadata in central Data Catalogs and use the metadata directly for translating natural language prompts using AI.</p>
oci_apiformat	<p>Specifies the format in which the API expects data to be sent and received. Use this attribute to generate text responses. This attribute applies to OCI Generative AI Chat Models in a dedicated AI cluster. Specify this attribute when you specify a model OCID in the <code>model</code> attribute or provide an endpoint in the <code>oci_endpoint_id</code> attribute.</p> <p>Supported values are:</p> <ul style="list-style-type: none">• COHERE• GENERIC
<div> Note: Use this attribute for OCI Generative AI Chat Models</div>	
oci_compartment_id	<p>Specifies the OCID of the compartment you are permitted to access when calling the OCI Generative AI service. The compartment ID can contain alphanumeric characters, hyphens and dots.</p> <p>The default is the compartment ID of the PDB.</p>

Attribute Name	Description
oci_endpoint_id	<p>This attributes indicates the endpoint OCID of the Oracle dedicated AI hosting cluster. The endpoint ID can contain alphanumeric characters, hyphens and dots. To find the endpoint OCID, see Getting an Endpoint's Details in Generative AI.</p> <p>When you want to use the Oracle dedicated AI cluster, you must provide the endpoint OCID of the hosting cluster.</p> <p>By default, the endpoint ID is empty and the model is on-demand on a shared infrastructure.</p>
oci_runtime_type	<p>This attribute indicates the runtime type of the provided model. This attribute is applicable to OCI Generate Text models in a dedicated AI cluster. Specify this attribute when you specify a model OCID in the <code>model</code> attribute or provide an endpoint in the <code>oci_endpoint_id</code> attribute.</p> <p>All permitted values can be found in OCI Generative AI runtimeType. See LlmInferenceRequest Reference.</p> <p>The supported values are:</p> <ul style="list-style-type: none">• COHERE• LLAMA
provider	<p>AI provider for the AI profile.</p> <p>Supported providers:</p> <ul style="list-style-type: none">• openai• cohere• azure• oci <p>This is a mandatory attribute.</p>
region	<p>This attribute indicates the location of the Generative AI cluster that you want to use. The region can contain alphanumeric characters and hyphen characters.</p>
<div> Note: The Oracle Generative AI cluster is available in Chicago, Frankfurt, and London regions. See Pretrained Foundational Models in Generative AI.</div>	
The default region is <code>us-chicago-1</code> .	
stop_tokens	<p>The generated text will be terminated at the beginning of the earliest stop sequence. Sequence will be incorporated into the text. The attribute value must be a valid array of string values in JSON format. <code>stop_tokens</code> takes a JSON array as input. To learn more about stop tokens or stop sequences, see Cohere documentation.</p>
temperature	<p>Sampling from Generate Text models incorporates randomness, so that the same prompt may yield different outputs each time you hit "generate". Temperature is a non-negative float number used to tune the degree of randomness. Lower temperatures mean less random generations. See Temperature for more details. This parameter is applicable to all the supported service providers.</p>

The following example is using Cohere as the provider and displays custom profile attributes:

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'COHERE',
    attributes =>
      '{"provider": "cohere",
       "credential_name": "COHERE_CRED",
       "object_list": [{"owner": "ADB_USER"}],
       "max_tokens": 512,
       "stop_tokens": [";"],
       "model": "command-nightly",
       "temperature": 0.5,
       "comments": true
      }');
END;
/
```

The following example shows custom profile attributes using OCI Generative AI:

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name =>
      'GENAI',
    attributes => '{"provider":
      "oci",
       "credential_name": "GENAI_CRED",
       "object_list": [{"owner": "SH", "name": "customers"},
                       {"owner": "SH", "name": "countries"},
                       {"owner": "SH", "name": "supplementary_demographics"},
                       {"owner": "SH", "name": "profits"},
                       {"owner": "SH", "name": "promotions"},
                       {"owner": "SH", "name": "products"}],
       "oci_compartment_id": "ocidl.compartment.oc1...",
       "oci_endpoint_id": "ocidl.generativeaiendpoint.oc1.us-chicago-1....",
       "region": "us-chicago-1",
       "model": "cohere.command-light",
       "oci_runtime_type": "COHERE"
      }');
END;
/
```