ORACLE

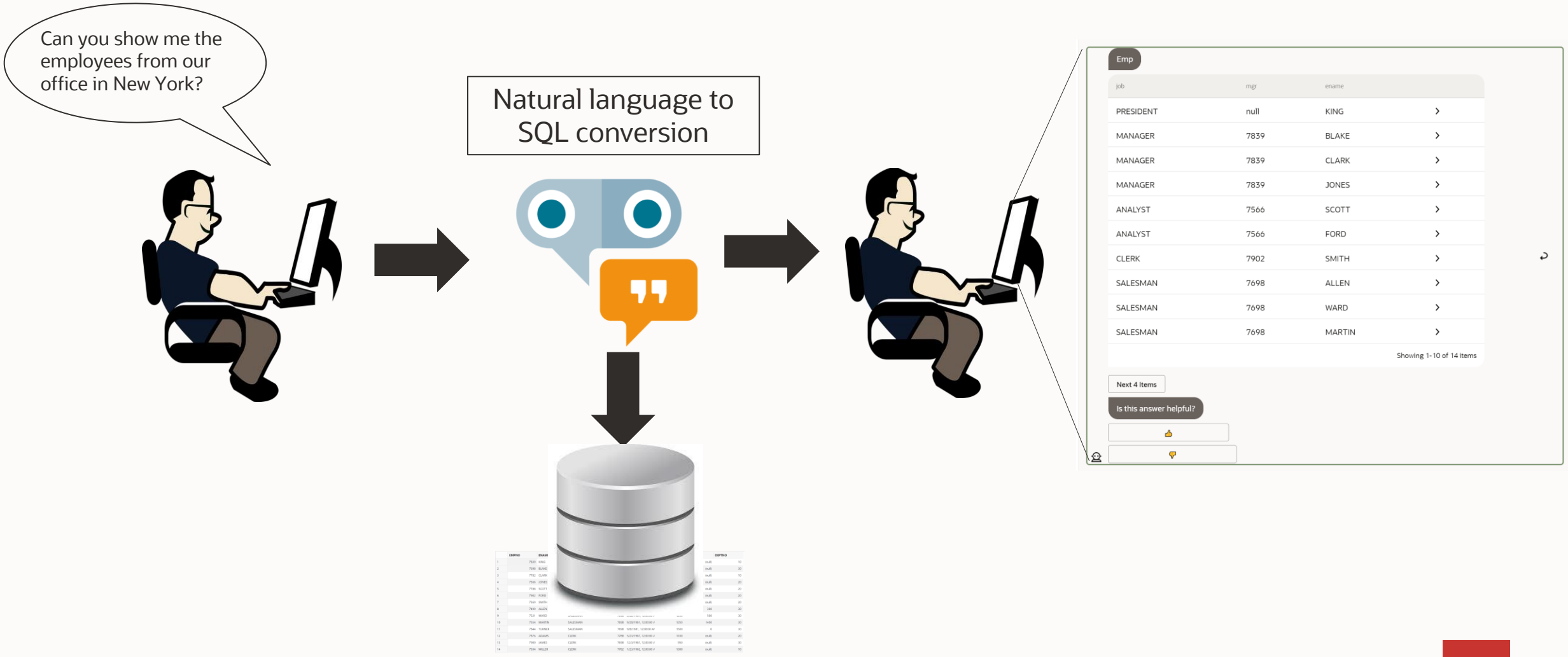
SQL Dialog

Program agenda

1. **What is SQL Dialog?**
2. Steps to build an SQL Dialog
3. SQL Dialog training data
4. SQL Dialog presentation
5. SQL Dialog queries

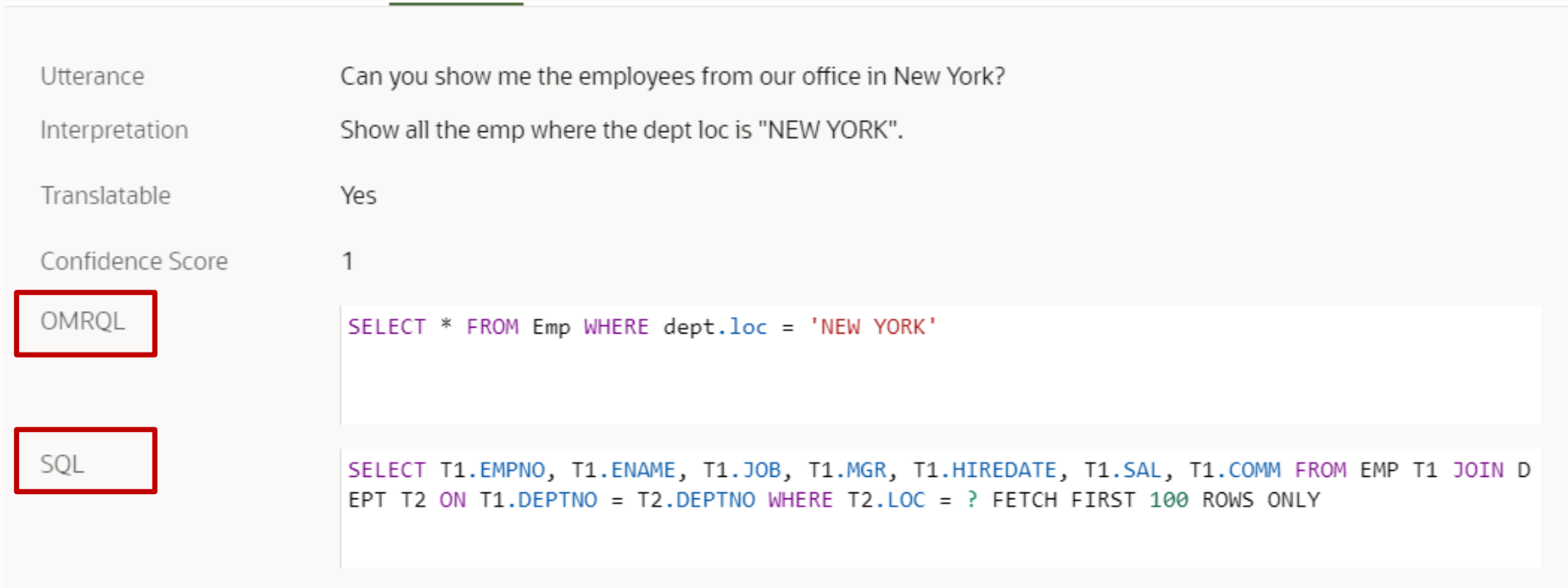
SQL Dialog provides
a **natural language conversational interface to databases**

How does it work?



How does it work?

Oracle meaning representation query language (**OMRQL**) is a query based on the canonical names of the database table attributes.



OMRQL is then converted into the **SQL Query** which is used to query the database

Where do we define the intents?

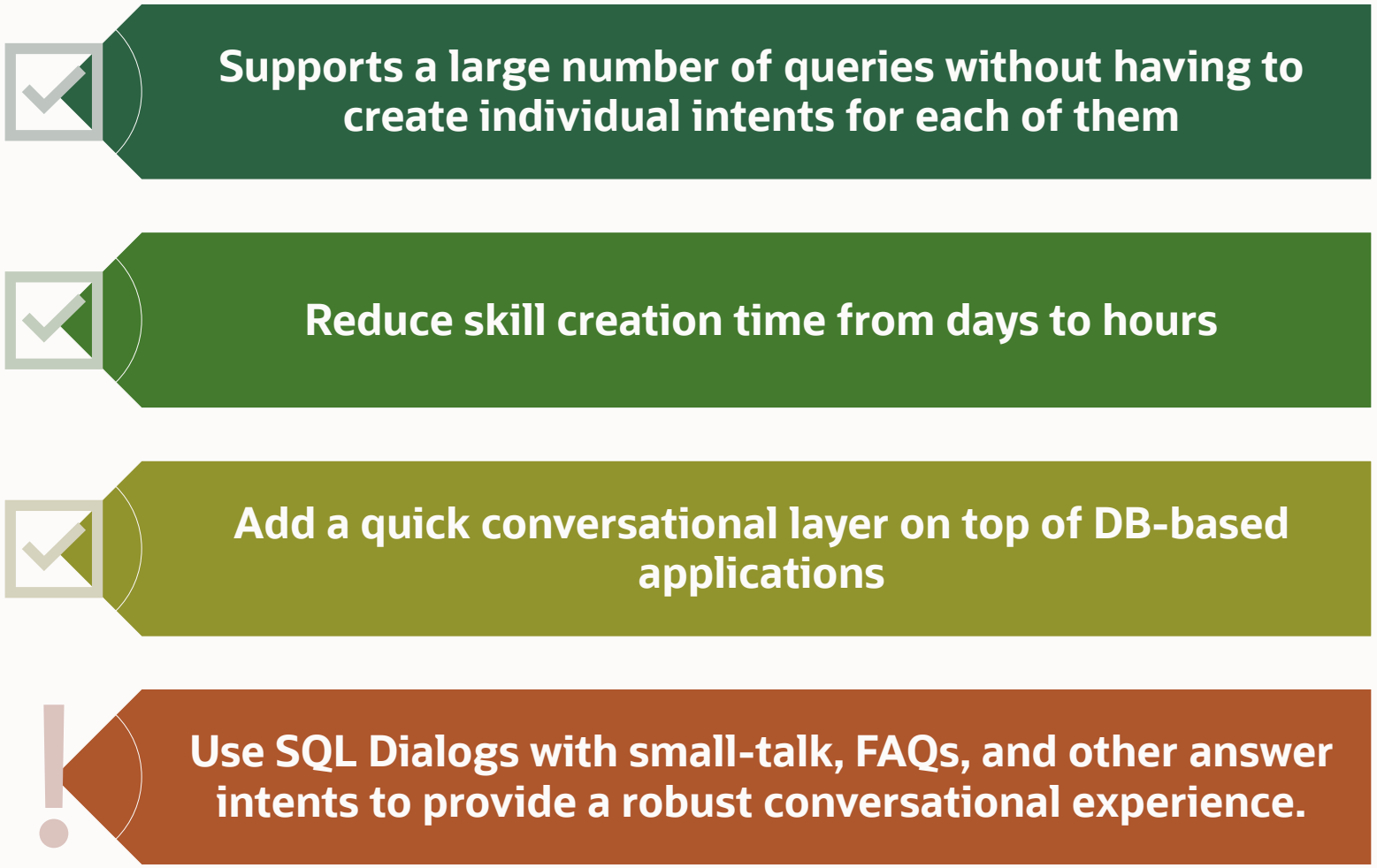
> SQL Dialog does not require Intents!

It uses a special entity (**query entity)** that is mapped to the database schema

The **query entity** has attributes mapped to the table columns

No need to identify all potential queries ahead of time - that means no intents, no utterances and custom components!

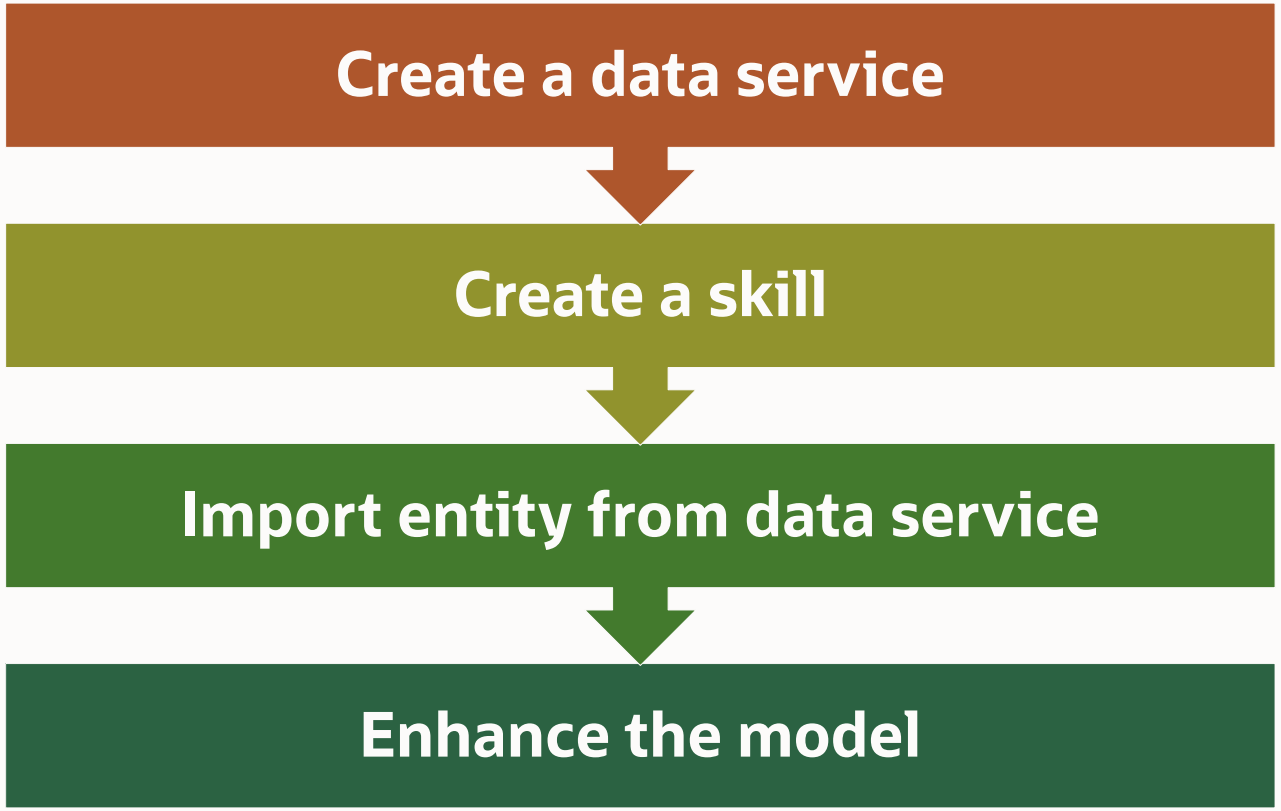
SQL Dialog Benefits



Program agenda

1. What is SQL Dialog?
2. **Steps to build an SQL Dialog**
3. SQL Dialog training data
4. SQL Dialog presentation
5. SQL Dialog queries

Steps to build a SQL Dialog



Create a data service

The first step is to configure a data service

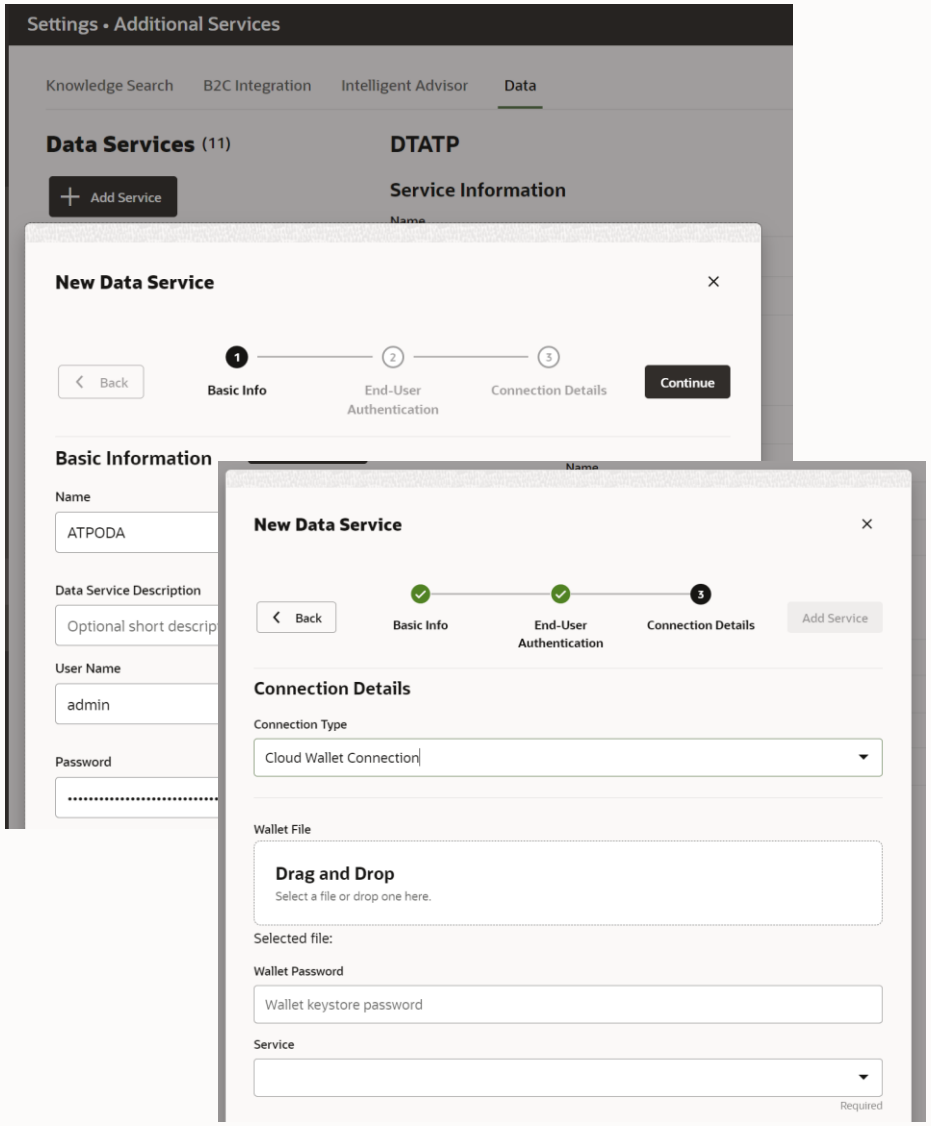
- You should have a database ready to use

Go to **Settings>Additional Services** and choose the data tab

The same data service can be used by multiple skills

Currently only the Oracle DBCS and ADB are supported

You can use a **Cloud Wallet** Connection Type or a **Basic** one



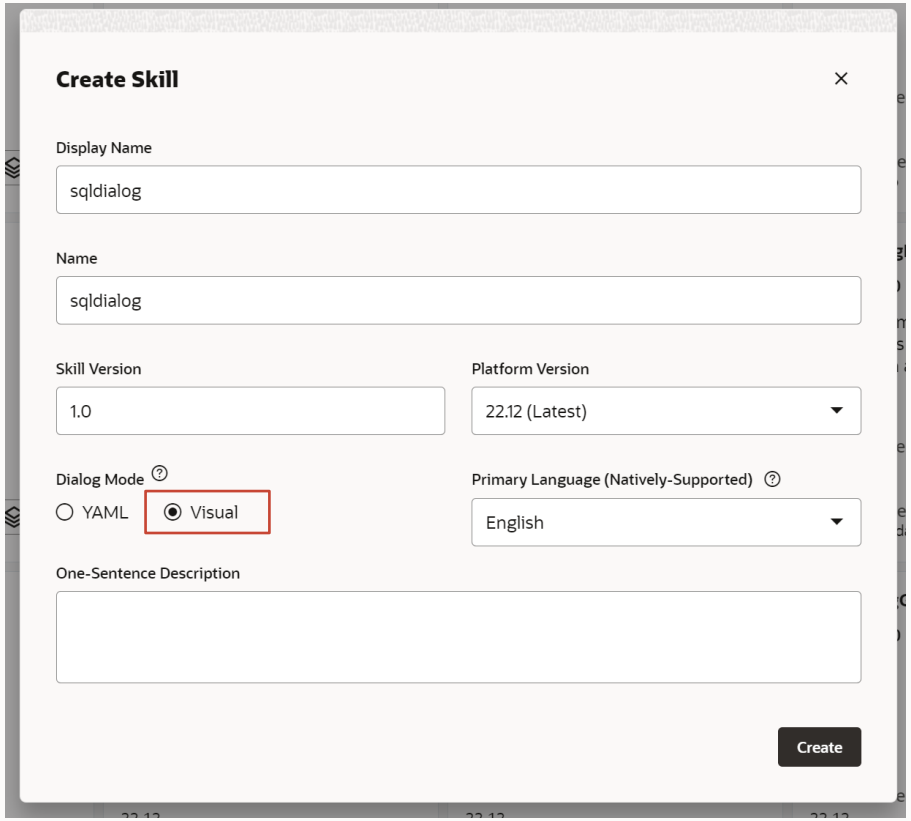
Create a skill

The skill must be using the **Visual Dialog Mode**

- YAML is not supported

The **Primary** Language needs to be English

- This version doesn't support multi-language SQL Dialog skills or skills where the primary language is not English

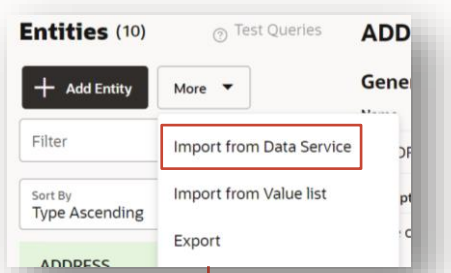


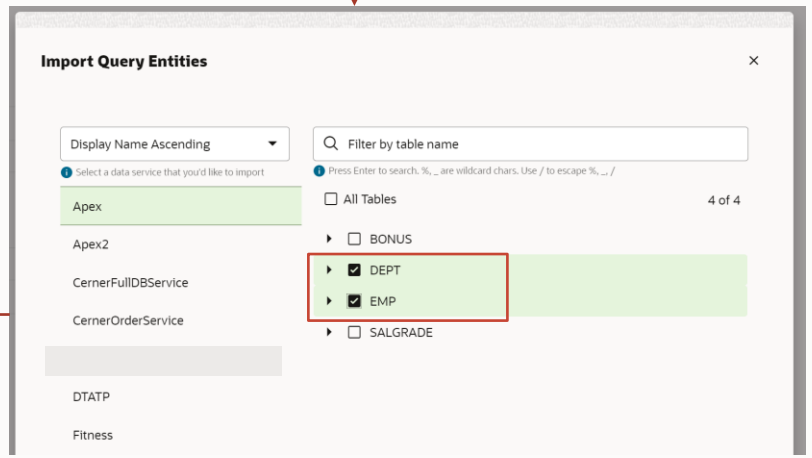
Import entity from data service

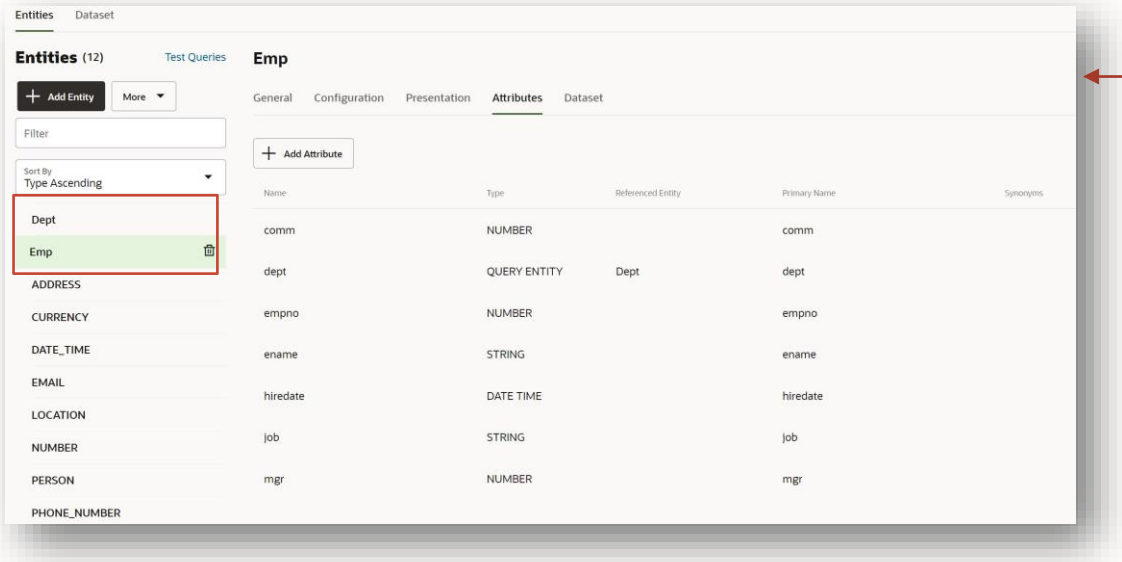
In the entities tab, use the **Import From Data Service** to automatically create **Query Entities**

You can choose the required tables and attributes

Each table will create a **query entity** with the selected attributes

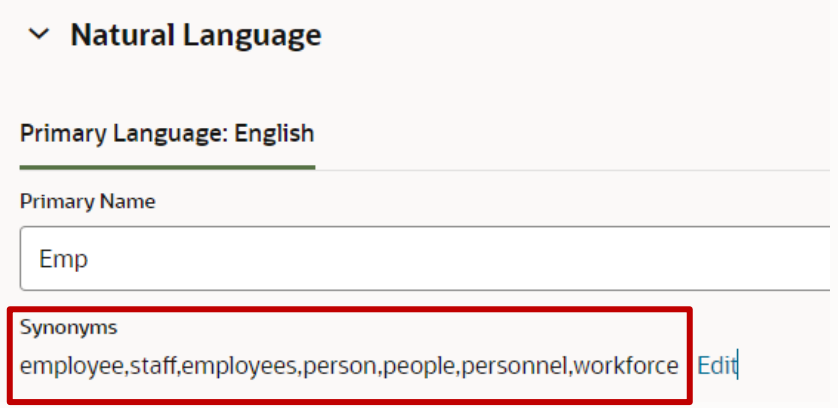






Enhance the model

To improve the model we should add **synonyms**, **value lists** and **utterances**, to help the skill associate natural language phrases, with the physical model's tables and columns



Emp

General | Configuration | Presentation | **Attributes** | Dataset

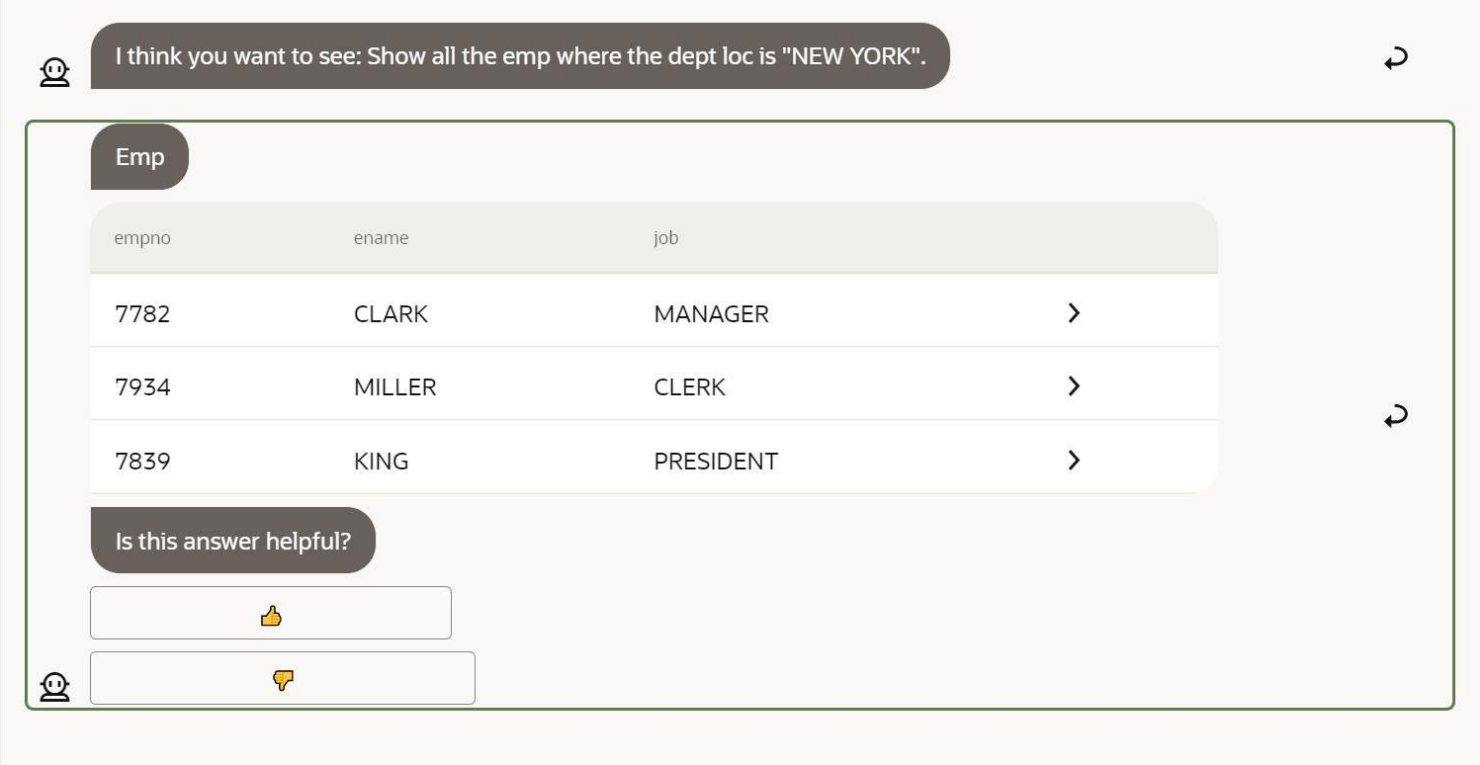
+ Add Attribute

Name	Type	Referenced Entity	Primary Name	Synonyms
comm	NUMBER		comm	commission,bonus
dept	QUERY ENTITY	Dept	dept	department
empno	NUMBER		empno	number,employee number
ename	STRING		ename	name,employee name

Enhance the model

With an enhanced model, the better the natural language queries are associated with an SQL

“can you show me all the employees in the big apple?”



Program agenda

1. What is SQL Dialog?
2. Steps to build an SQL Dialog
3. **SQL Dialog training data**
4. SQL Dialog presentation
5. SQL Dialog queries

SQL Dialog training data

> We should add training data to improve the skill ability to interpret and route queries

If your skill has intents, or it is in a DA, then, just like with intents, your skill needs utterances to help it route SQL queries to the SQL Dialogs conversation

The routing mechanism uses the autocomplete suggestions, training data, generated routing utterances, and handcrafted routing utterances to learn how to recognize SQL queries.

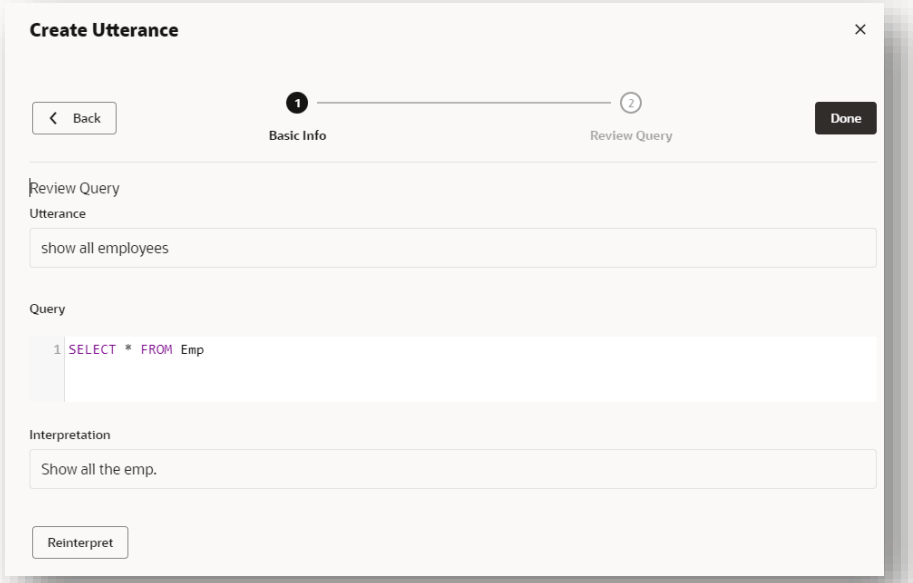
You can see each type of utterance in the separate tabs on the **Query Entities Dataset** page.

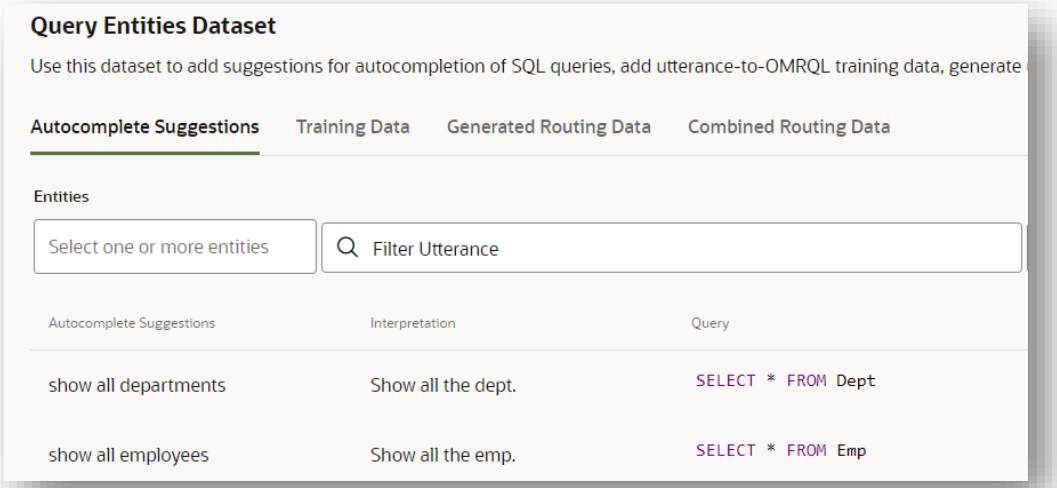
Autocomplete

You can help users learn about the database queries that they can make by providing autocomplete suggestions

These suggestions provide hints about what kinds of questions the logical model can answer

The utterances also help the skill with routing



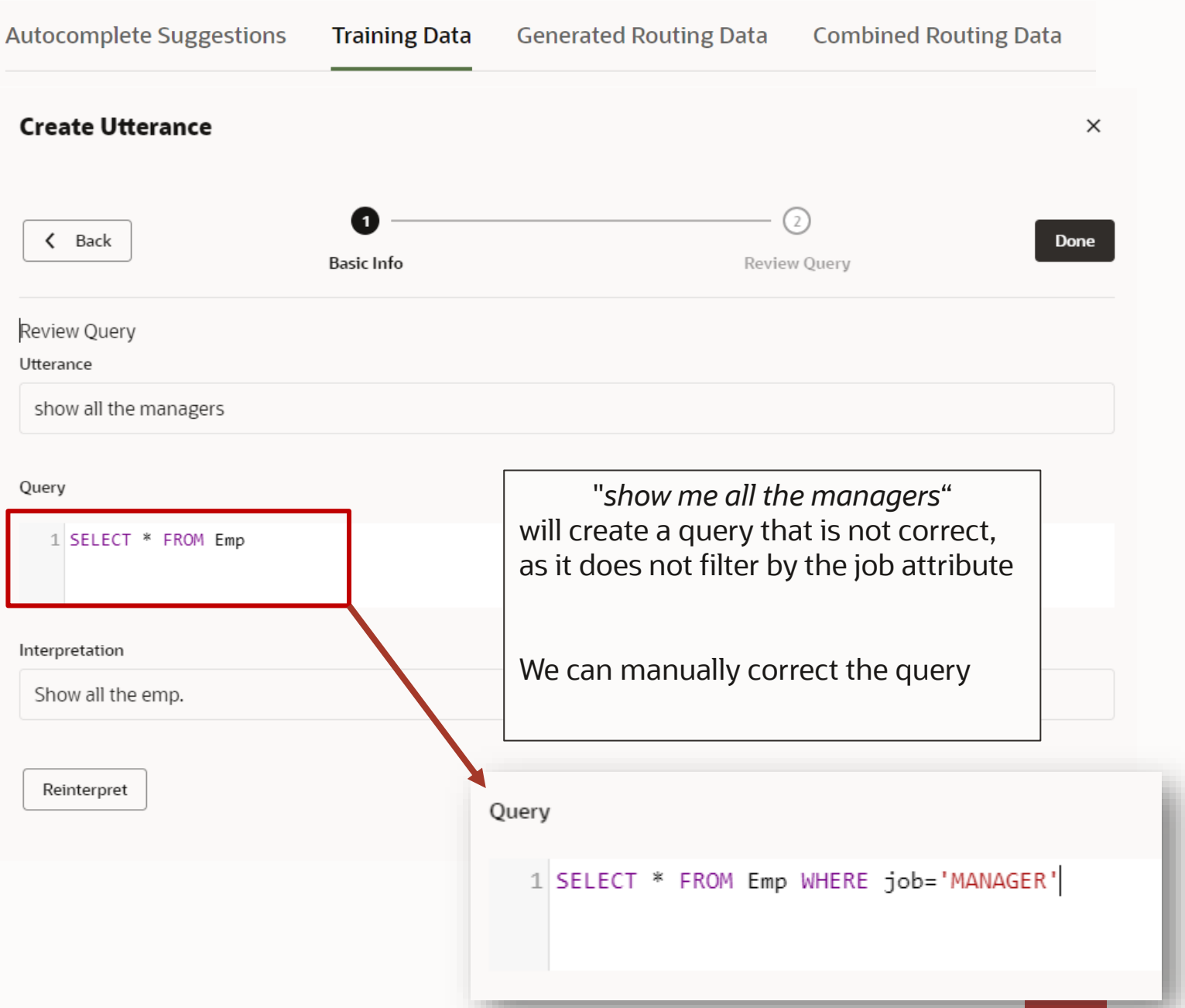


Training data

Where the skill is not interpreting natural language queries correctly, you can use custom training data to teach the correct interpretation

Creating utterances follows the same guidelines as for the *standard* utterances

- Balance the number of utterances
- Grammatically diversity
- Diversify Values
- Balance the training of values that match primary names or synonyms

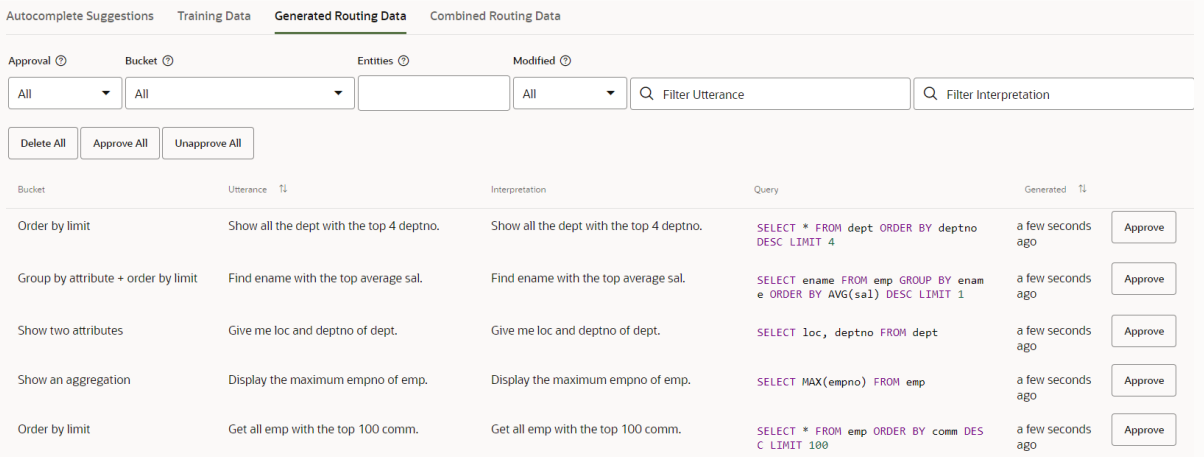


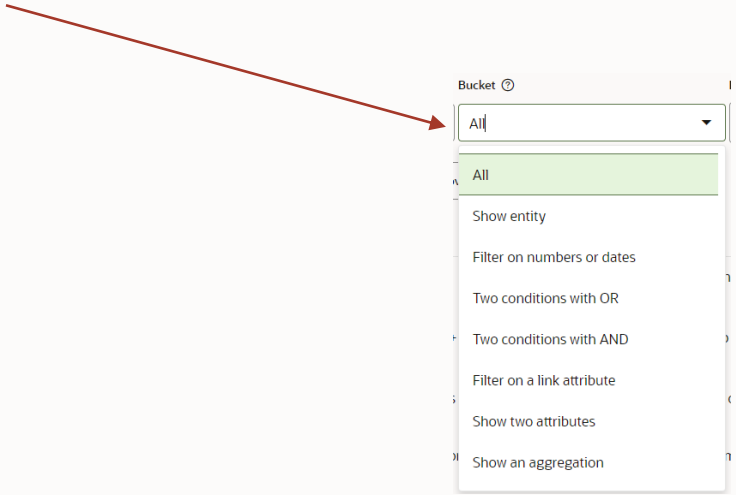
Generated routing data

Accurate routing of utterances to the SQL conversation requires a lot of sample utterances with the translated query. On the **Generated Routing Data** tab, you can quickly generate 100 utterances that reflect questions the logical model can answer

Those generated utterances are based on templates (buckets) and can be used to filter the data

Must be reviewed before use. Those that you approve are added to the **Combined Routing Data** tab and are marked as either *synthetic* or, if you edited them, *refined*



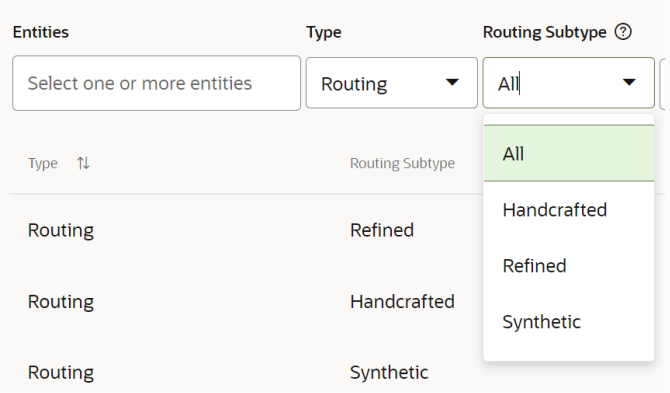


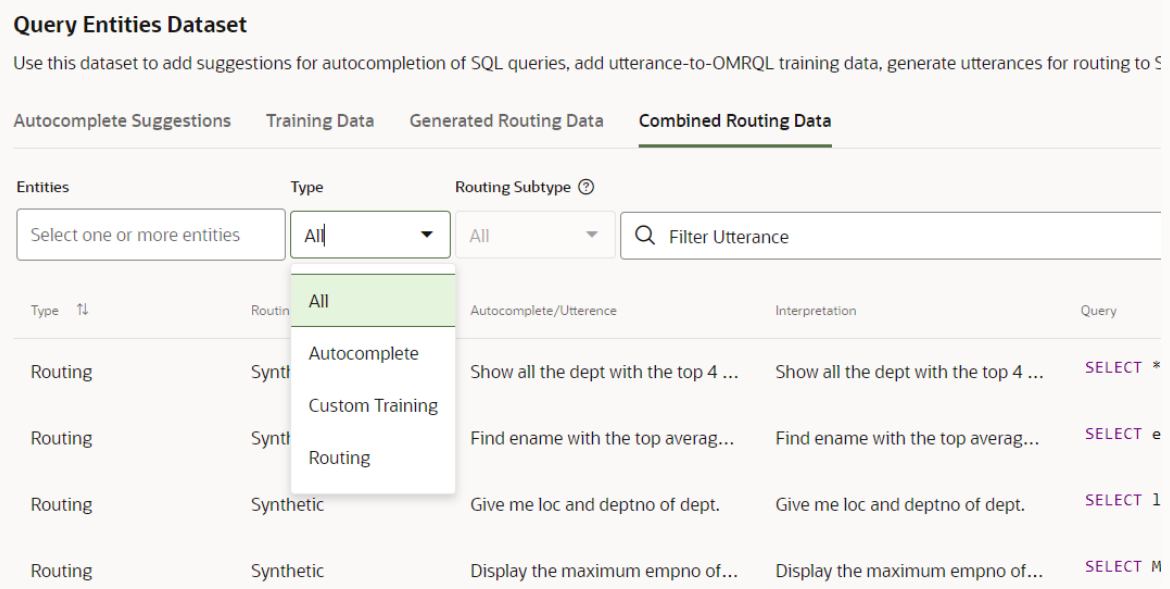
Combined routing data

The combined routing data is an aggregation of all autocomplete suggestions, custom training data, and generated and hand crafted routing data

Hand crafted data is manually defined routing data

All of these help the skill route SQL queries to the SQL conversation





Program agenda

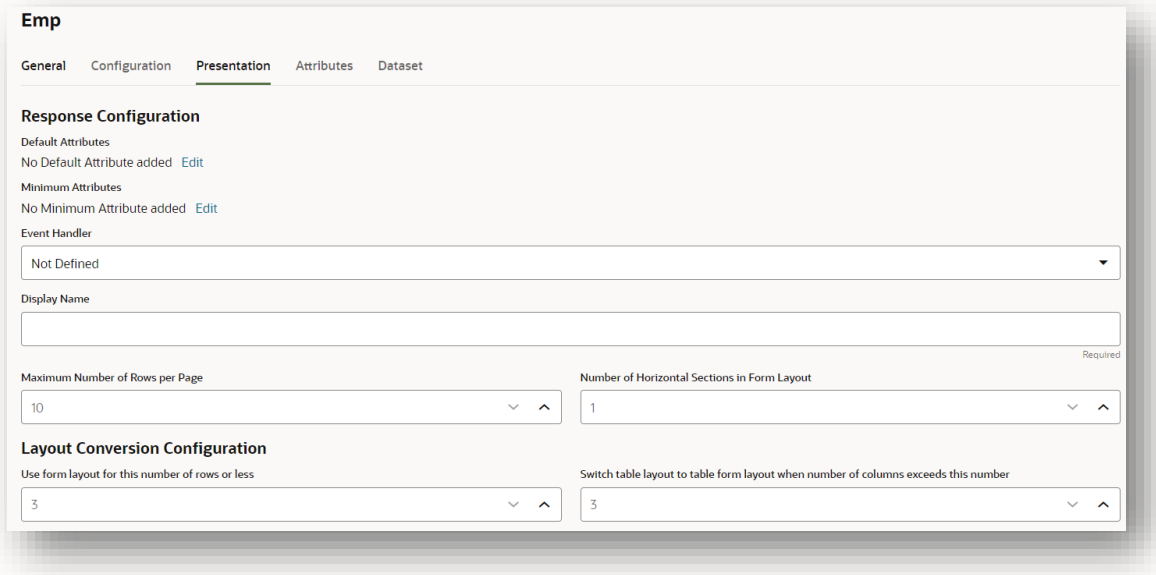
1. What is SQL Dialog?
2. Steps to build an SQL Dialog
3. SQL Dialog training data
4. **SQL Dialog presentation**
5. SQL Dialog queries

SQL Dialog presentation

You can control how to present the data to the user

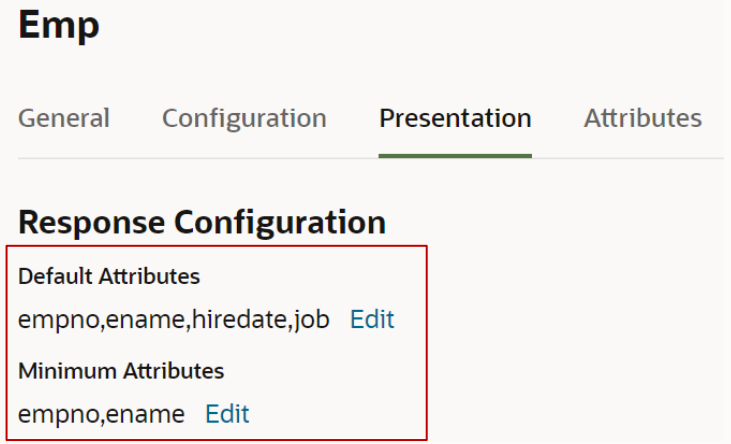
Some of the things it can do:

- Configure Whether to Display Form or Table
- Show One or Two Horizontal Sections in Form
- Set the Title for the Results
- Define an Entity's Default Sort Order
- Define Which Attributes to Include When Not Specified by the Utterance
- Define Which Attributes to Always Include in the Results
- Configure the Results Page Size
- Add Buttons and Links to Results
- Add a Custom Attribute
- Use Event Handlers for more customization

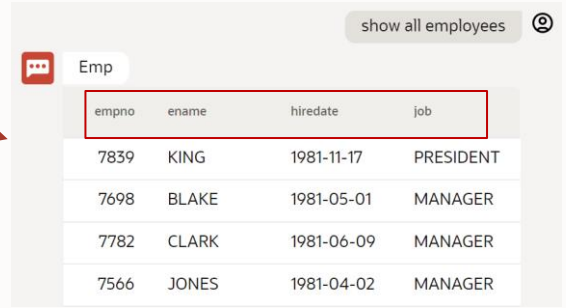


SQL Dialog presentation

When it comes to the response layout you should start by defining Default and Minimum attributes.



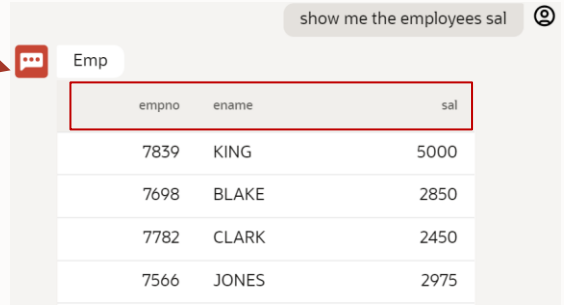
If the utterance doesn't name any attributes, then you probably want the results to include **Default Attributes.**



When an utterance identifies specific attributes, you might want the result to include not only the requested attributes, but also **Minimum Attributes**.

"show me the employees salary"

Here we only ask for the salary, but the minimum list will define the extra attributes to show



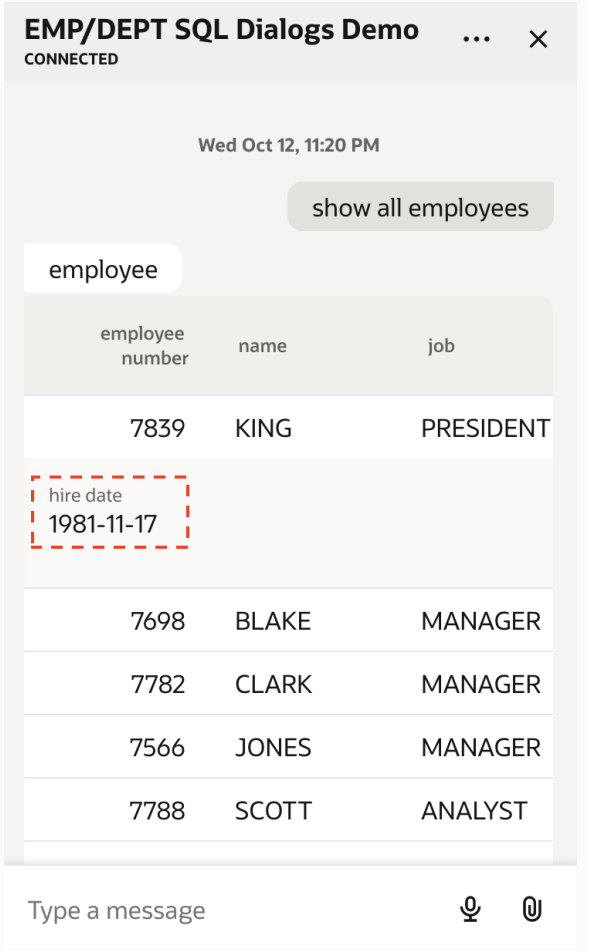
SQL Dialog presentation

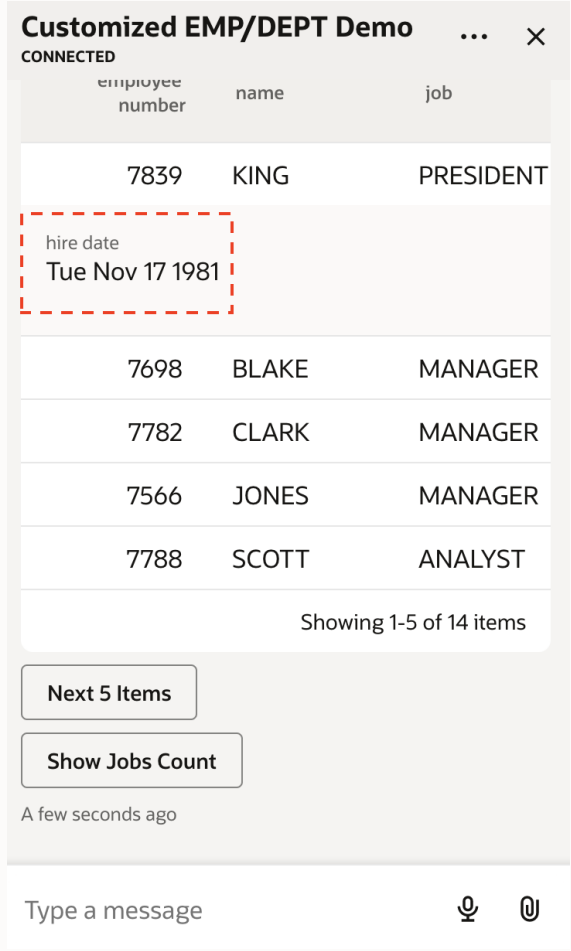
Formatting an attribute

Attributes can be formatted on an individual basis with a series of predefined parameters

- Date/Time/Number have specific **format masks**

For further formatting capabilities, entity event handlers have access to the attributes and allow further customization



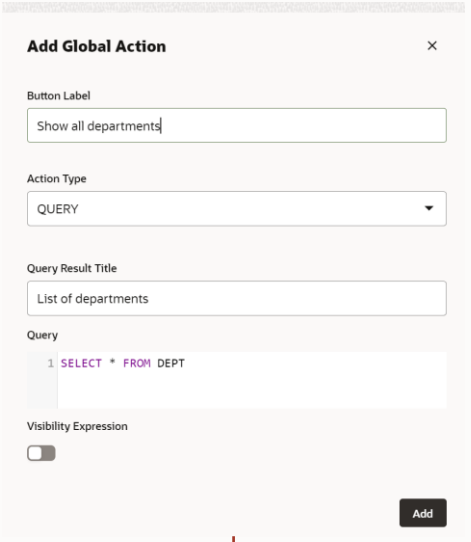


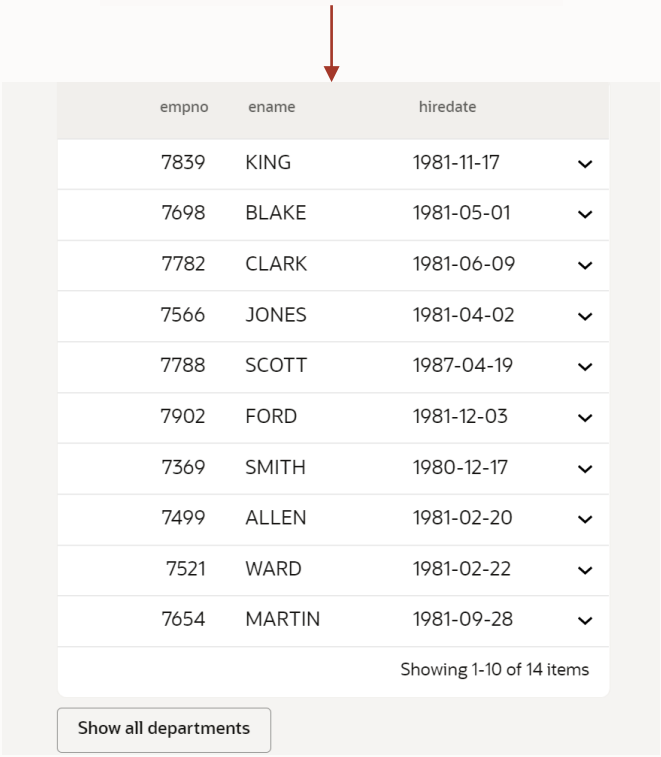
SQL Dialog presentation

Global action – Show all departments

You can add buttons and links to a query entity's results at both the global level and the row level

We can optionally restrict when the action appears. with **Visibility Expression** where we can provide a FreeMarker expression, such as ${row.job = 'MANAGER'}.





Follow-Up Actions Configuration

Add Global Action

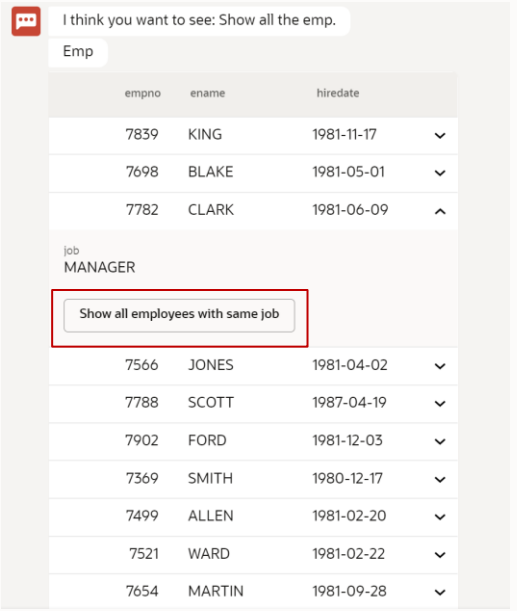
Sequence	Type	Button Label	Query/URL
1	QUERY	Show all departments	SELECT * FROM DEPT

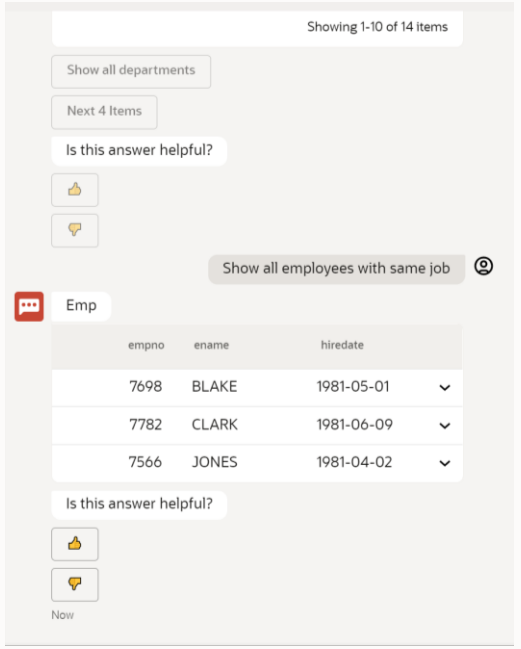
SQL Dialog presentation

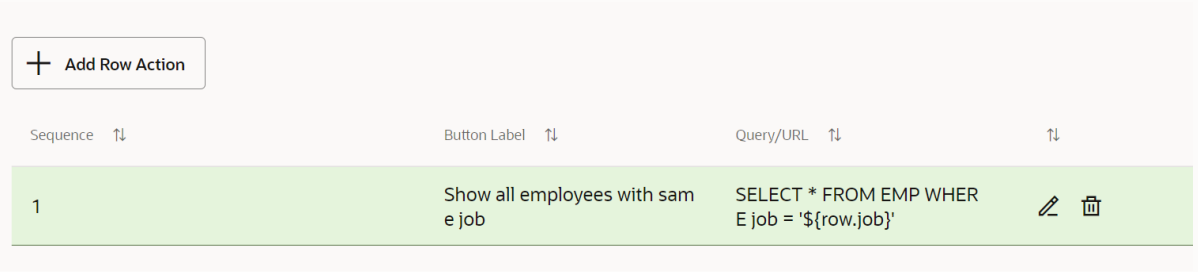
Row action – show all employees with same job

For row actions you need to create a query that uses data from the row to invoke a new query

- ${row.<attribute_name>} allows to dynamically do this







SQL Dialog presentation

Event Handlers – SQL Query Event Handler

An SQL Query Event Handler (SQEH) enables you to customize the SQL Dialogs query results.

The SQEH is deployed as part of a component service. You use below command to add an event handler to an existing package.

```
bots-node-sdk init component myEventHandler sqlQueryEventHandler
```

This example creates a component of type **sqlQueryEventHandler** that is named **myEventHandler**.

https://github.com/oracle/bots-node-sdk/blob/master/DATA_QUERY_EVENT_HANDLER.md

SQL Dialog presentation

Event Handlers – SQL Query Event Handler

You can have **Entity** level and **Attribute** level events

Entity

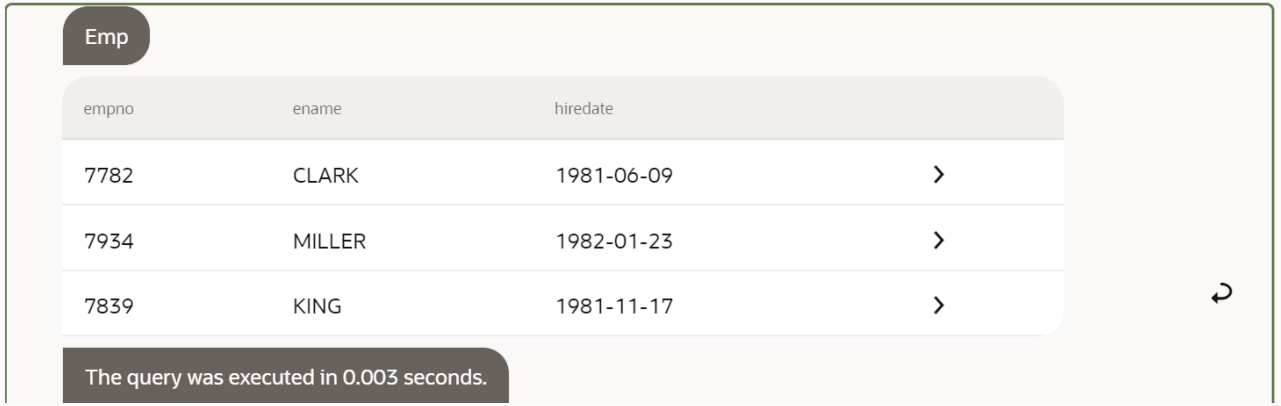
- changeUISettings
- changeResponseData
- changeBotMessages

Attribute

- changeUISettings
- format

```
changeBotMessages: async (event, context) => {
  let message = event.messages[event.messages.length - 1];
  let millis = context.getQueryExecutionTime();
  let minutes = Math.floor(millis / 60000);
  let seconds = ((millis % 60000) / 1000).toFixed(3);
  message.footerText = `The query was executed in ${minutes > 0 ? minutes +
      ' minutes and ' : ''}${seconds} seconds.\n\n${message.footerText}`;
  return event.messages;
}
```

Example that adds a footer text that displays the execution time of the query



Program agenda

1. What is SQL Dialog?
2. Steps to build an SQL Dialog
3. SQL Dialog training data
4. SQL Dialog presentation
5. **SQL Dialog queries**

SQL Dialog **supported queries**

The SQL Dialogs natural language processing model supports queries that translate to the basic SQL clauses:

SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY and **LIMIT**.

SQL Dialog supported queries

Examples of supported queries

Display

- show me all the employees
- name, salary and department name of employees
- what are the unique jobs that an employee can have?
- how many employees
- return the highest salary of all employees

Filters

- show the names of employees whose job title is clerk
- return the salary of all clerks
- employees whose names starts with Jo
- view employees who joined in 2020 and earn above 7000

Filters with dates

- Absolute example: who were the employees hired on 10 Dec 2020
- Relative interval example: who were the employees hired last year
- Relative Date example: who were the employees hired today
- Date + time example: packages delivered yesterday at 6 pm
- Date + time interval example: Packages delivered between 5th Jan 7 pm and 5th Feb 10 am

Ordering and limiting the number of rows

- show employees sorted by their department names
- return the name and salary of all employees in descending order of salary
- what are the 10 highest salaries of all employees
- which employee has the lowest salary
- show the top 5 employees

Group by

- the average salary of each job
- what is the highest salary per department?
- show the name, location and number of employees per department
- show all jobs with an average salary above 3000

https://docs.oracle.com/en/cloud/paas/digital-assistant/use-chatbot/sql-dialog-skills.html#GUID-D094DD0E-207A-4901-BE6C-168AD4D51C60

SQL Dialog un-supported queries

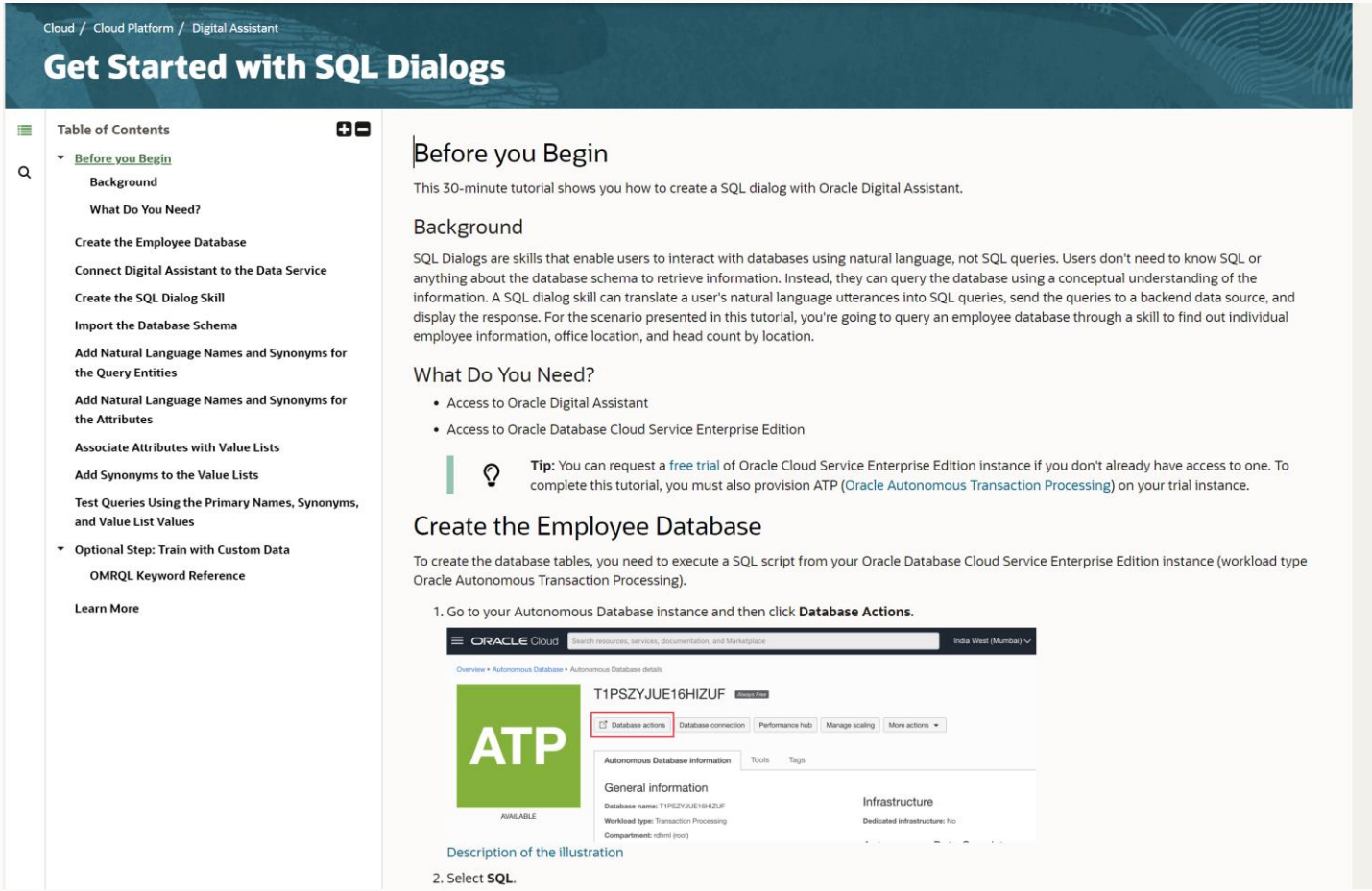
SQL Dialogs doesn't support the more complex queries that involve sub-queries and SET operators (INTERSECT, UNION, EXCEPT, and NONE)

Other un-supported queries (no conversion to OMRQL)

- Non-English query
- Use of pronouns -> *“What is **my** salary?”*
- Yes and no questions -> *“is John a clerk?”*
- Negation -> *“Which employee is **not** on location Amsterdam?”*
- Selecting more than one entity -> *“Show me all employees and departments”*
- SQL queries that involve sub-queries and SET operators (INTERSECT, UNION, EXCEPT, and NONE)
  - *“what is the total remuneration earned by each employee?”*

Check the documentation for a full list

More resources - Tutorial



https://docs.oracle.com/en/cloud/paas/digital-assistant/tutorial-sql-dialogs/index.html


ORACLE