IN OAUG
XPLORE

UNLEASHING THE POTENTIAL OF ORACLE APEX:

MASTERING REST API FOR MODERN APPS

5th May 2023, 5 PM, IST

Rutvik Prajapati
Programmer Analyst
Loblaw Companies Ltd.

**Rutvik Prajapati , @rutveek*

About Me

- 9+ years of experience as Oracle Database Developer
- Started my career as Oracle FORMS and Reports Developer
- Working on Oracle APEX since 2014
- Programmer Analyst, Loblaw Companies Limited, Winnipeg, Canada

Rutvik Prajapati

@rutveek

bit.ly/rutvik_linkedin

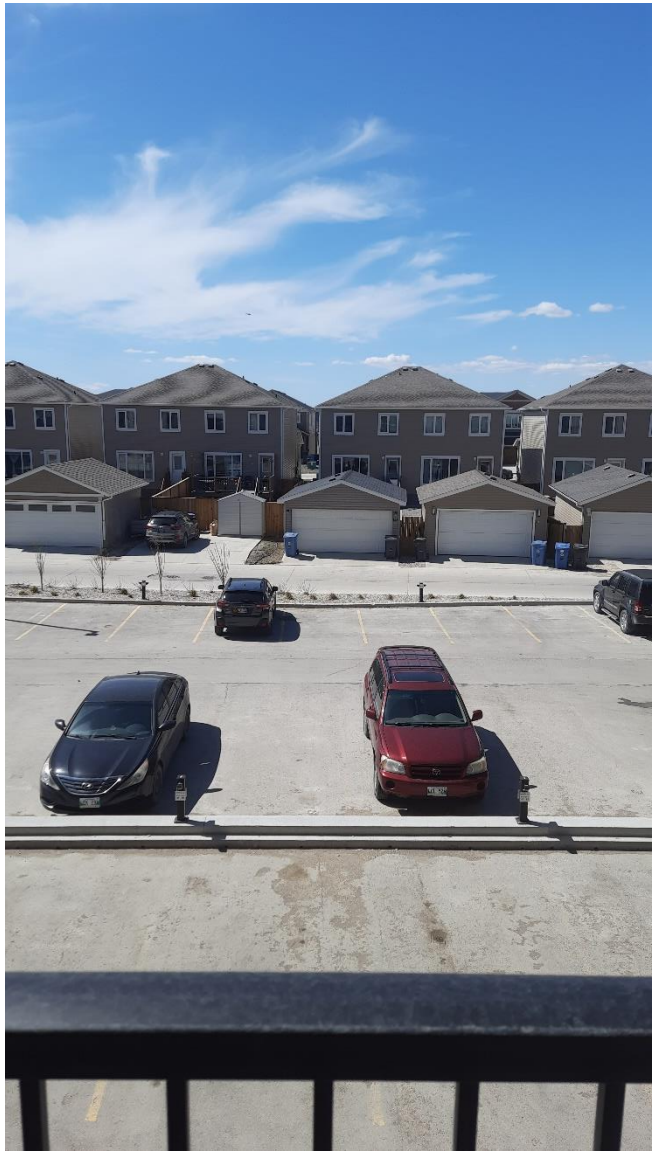
bit.ly/rutvik_blog

Oracle ACE Associate

About me

*Rutvik Prajapati , @rutveek

About me



*Rutvik Prajapati , @rutveek

Canada
Oracle APEX
Special Interest Group

@canapexsig

https://www.linkedin.com/company/canapexsig

canapexsig@gmail.com

*Rutvik Prajapati , @rutveek

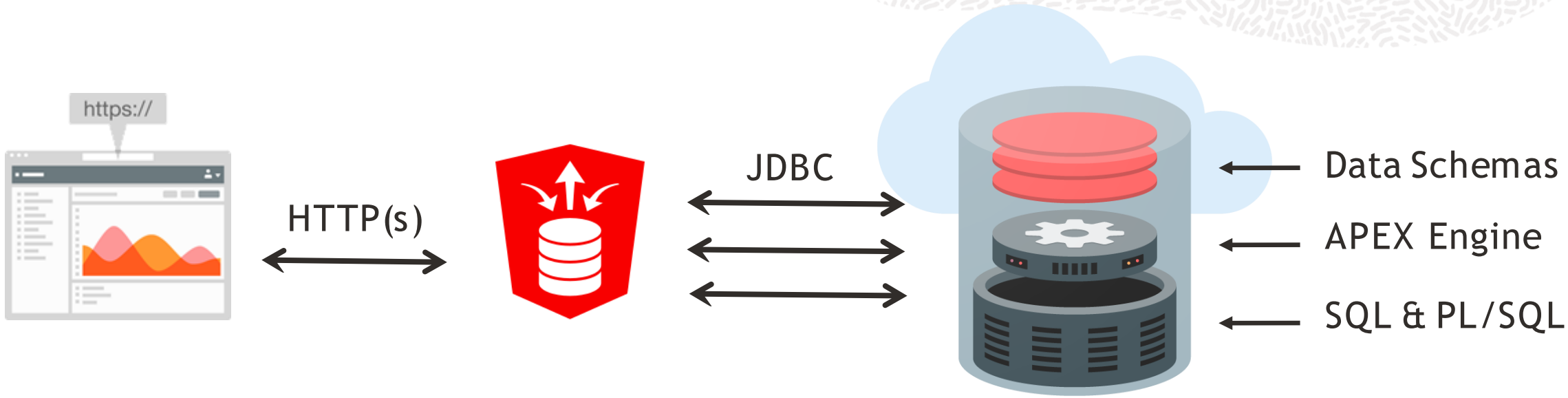

MOST THINGS WORTH HAVING
LIVE OUTSIDE OUR CONFORT ZONE
ON THE OTHER SIDE OF FEAR


MASH
PROGRAM
Mentor and Speaker Hub
Our goal is to *connect* speakers with mentors to assist in *preparing* technical sessions and *improving* presentation skills
Interested? Read more and get in touch
https://mashprogram.wordpress.com

Agenda

- Oracle APEX Architecture
- What is ORDS?
- What is REST API?
- Auto Rest Feature of ORDS
- Demo on Auto REST on a Table and Package Procedure
- Manually Create REST API on a Table
- Demo on Manually Creating REST API
- How to Secure REST API?
- Demo: Secure REST API

Oracle APEX Architecture



Oracle REST Data Services
(Weblogic, Jetty, Tomcat)

No Application Logic
Converts HTTP to database API calls

Oracle Database
(Pluggable or Dedicated, 12c *or Above*)

Zero latency database data access
Dynamically driven by APEX metadata

Browser | **Mid Tier** | **Database Tier**

*Rutvik Prajapati, @rutveek

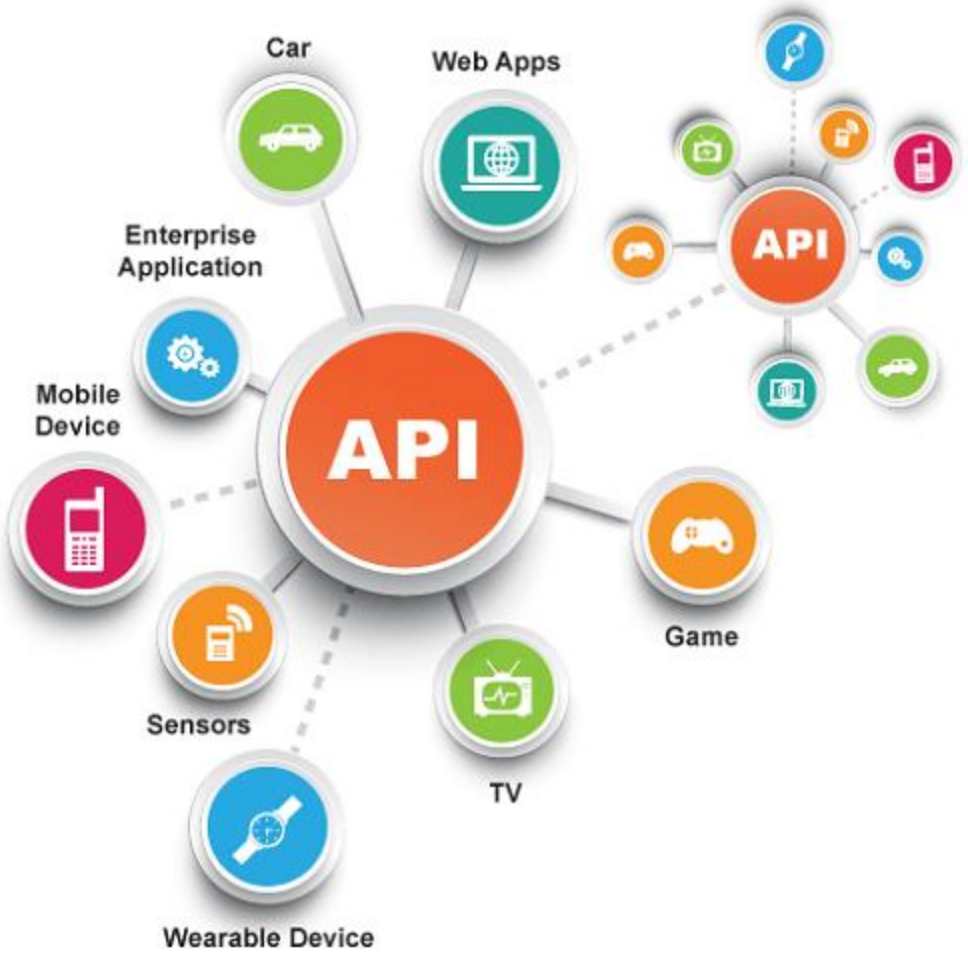
What is ORDS?

Oracle REST Data Services

- A mid-tier Java application
- Java EE-based alternative for Oracle HTTP serve which runs in Java application server like WebLogic or Tomcat.
- Maps standard http(s) RESTful requests to database transactions.
- Allows you to expose and capture data in your Oracle Database using the REST Protocol
- Access to Relational data over HTTP(s) without installing JDBC/ODBC drivers
- Automatically returns SQL queries in JSON.
- Automatically pages response for GET method services.
- Supports OAuth2.0 authentication

What is API?

- API stands for Application Programming Interface
- Application can interact with each other using API.
- An API is an interface that computer programs use to interact with an application.
- APIS are directly used by programs, not humans.
- API let computers talk to each other.

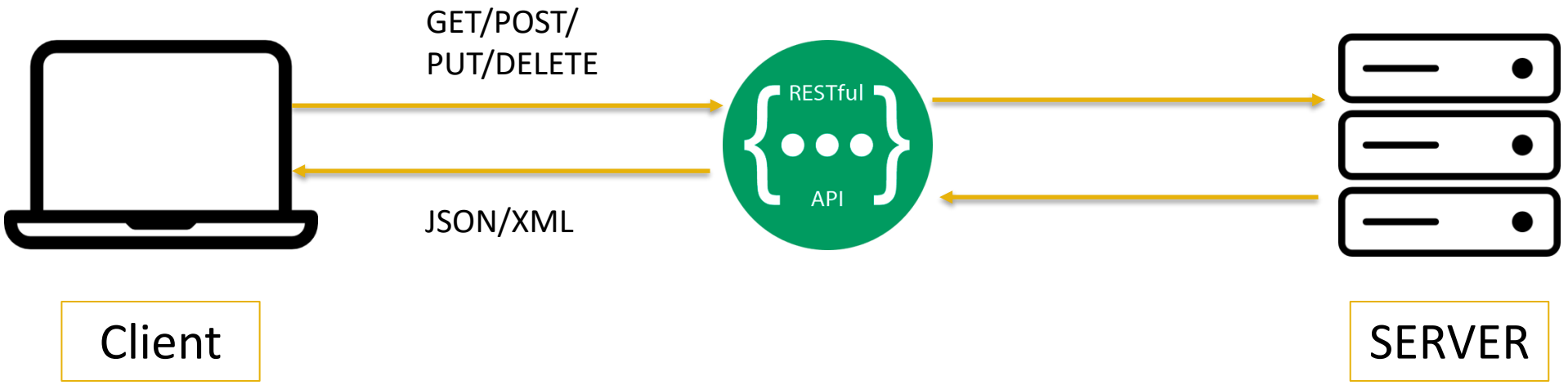


*Rutvik Prajapati , @rutveek

What is REST API?

Representational State Transfer

- An architecture that provides interoperability between two systems.
- Use HTTP(s) protocol for communication

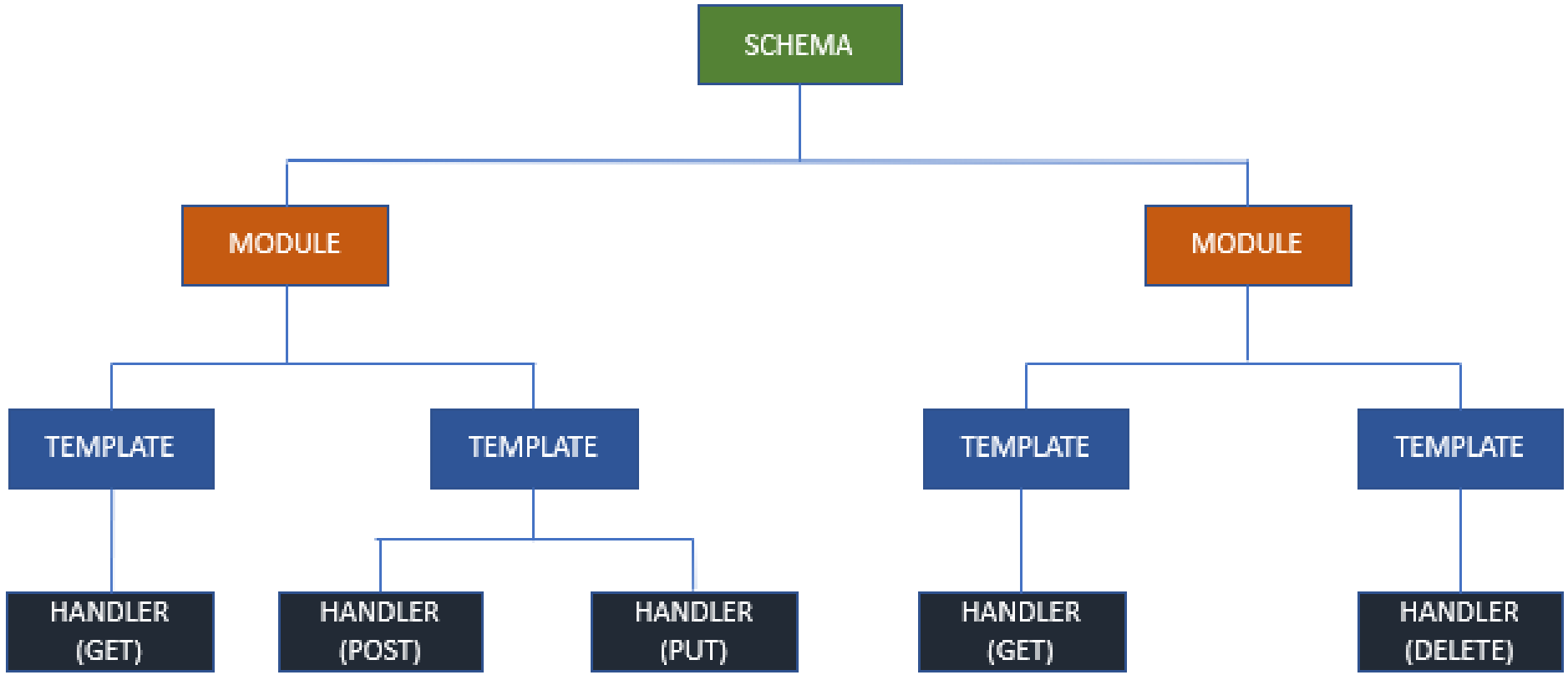


HTTPS Methods

- REST communicates over HTTP, therefore uses standard HTTP methods

Operation	HTTP Method	SQL Equivalent
CREATE	POST	INSERT
READ	GET	SELECT
UPDATE	PUT	UPDATE
DELETE	DELETE	DELETE

Structure of a ORDS Webservice



Understanding ORDS URI Terminology

https://server.com/ords/myschema/hr/employees/:id

server.com	Server URL	
ords	ORDS Alias	Alias for the ORDS engine (Defined at the web server level
myschema	Schema Alias	Defines the path used to reach the ORDS Enables Schema
hr	Module Base Path	Defines the BASE_PATH for a group of related services
employees	URI Template	Defines the path that is used to access a specific resource
:id	Bind Variable	Defines the bind variable used in the handler

Creating REST API in ORDS

ORDS provides two different ways to expose data/functionality via REST:

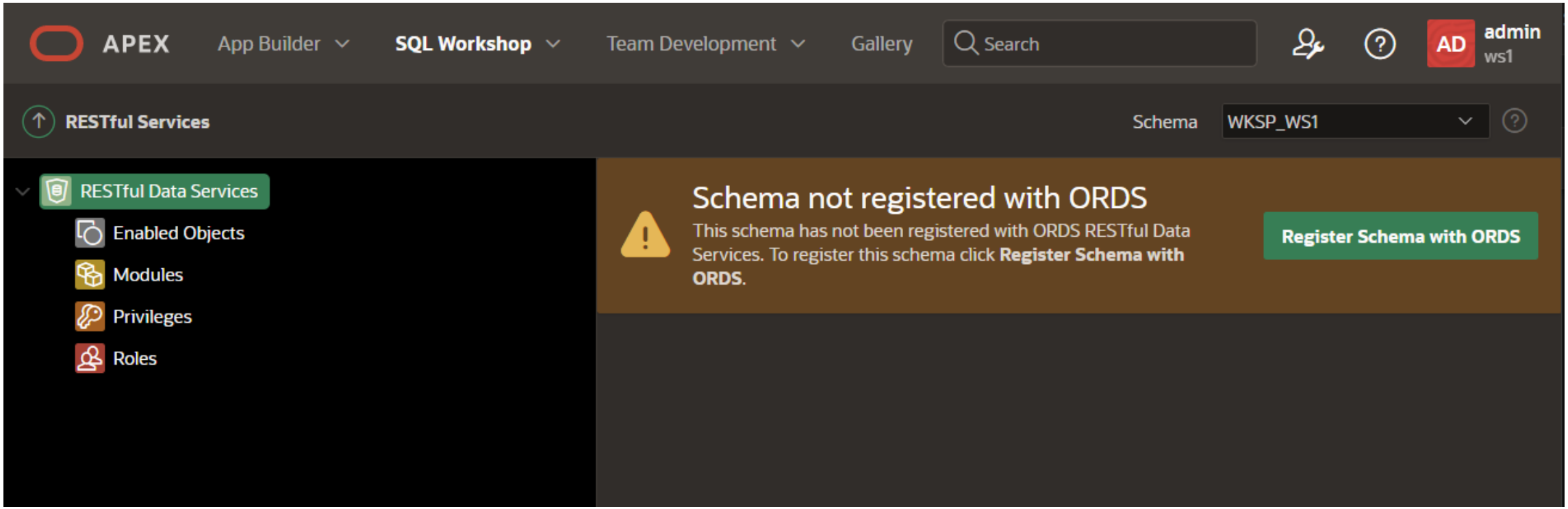
- **AutoREST Enablement**

- **Manual REST Service Creation**

AutoREST Enablement

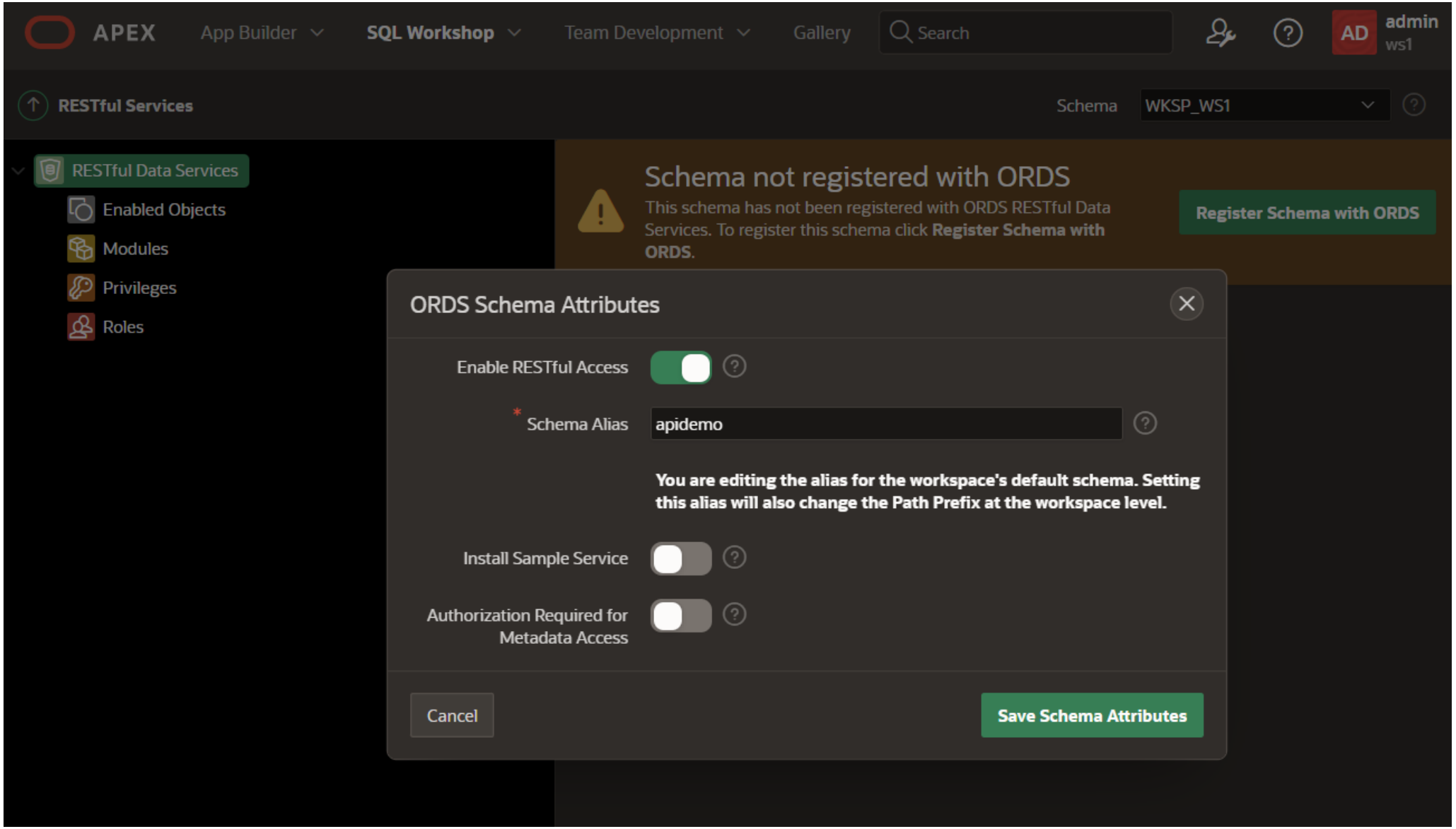
- Concept of making database resource available via ORDS with Zero Code
- Can expose Tables, Views, Packages, Procedures, and Functions
  - Tables (GET, POST, PUT, DELETE)
  - Views (GET)
  - Procedures, Functions, Packages (POST)
- You sacrifice flexibility and customizability for ease of creation
- No customization of included columns or data format
- No way to introduce extra validation or logic

How to enable schema for ORDS?



*Rutvik Prajapati , @rutveek

How to enable schema for ORDS?



*Rutvik Prajapati , @rutveek

How to enable schema for ORDS?

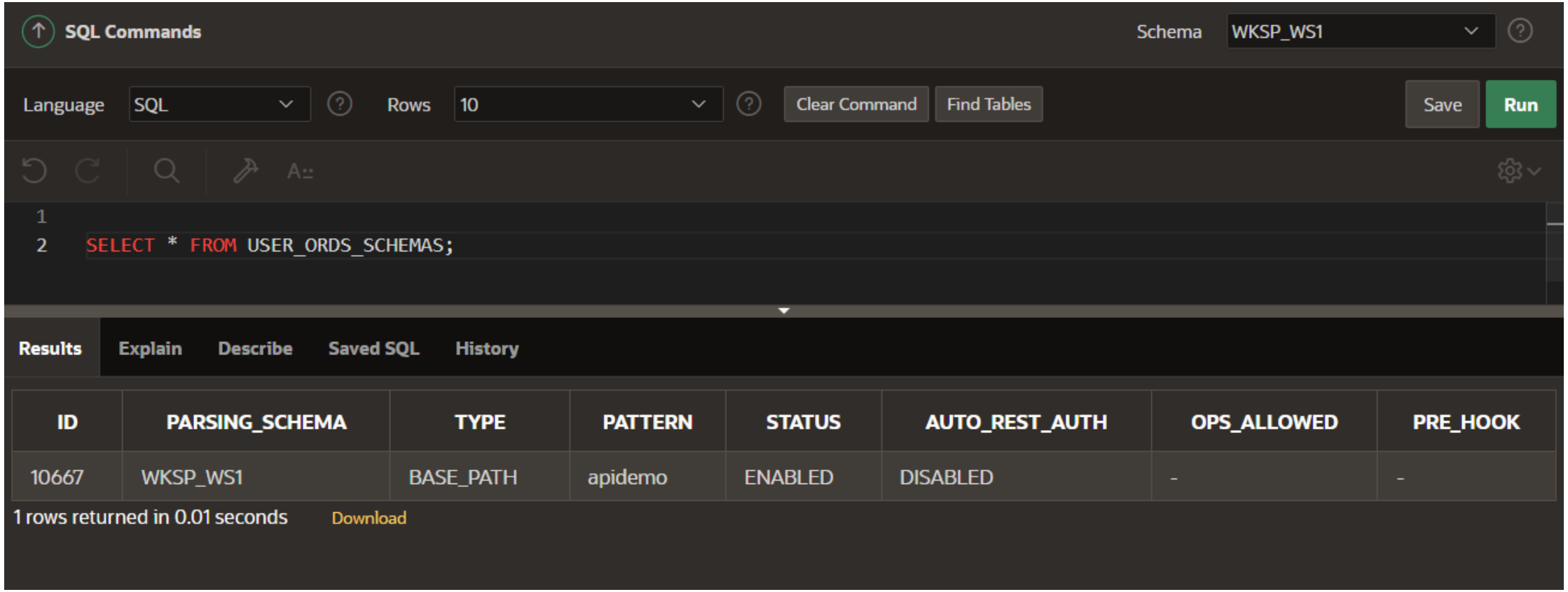
- Using ENABLE_SCHEMA procedure of ORDS API, we can enable or disable the schema.
- We only need to perform this action once.

```
ORDS.ENABLE_SCHEMA (
                        p_enabled                 => TRUE,
                        p_schema                  => ‘WKSP_RUTVIKWS’,
                       p_url_mapping_type         => 'BASE_PATH',
                       p_url_mapping_pattern      => ‘apidemo',   --- Schema Alias which will appear in the URL
                       p_auto_rest_auth           => FALSE
                  );
```

- **Note**: Only database users with the DBA role can enable or disable a schema other than their own.

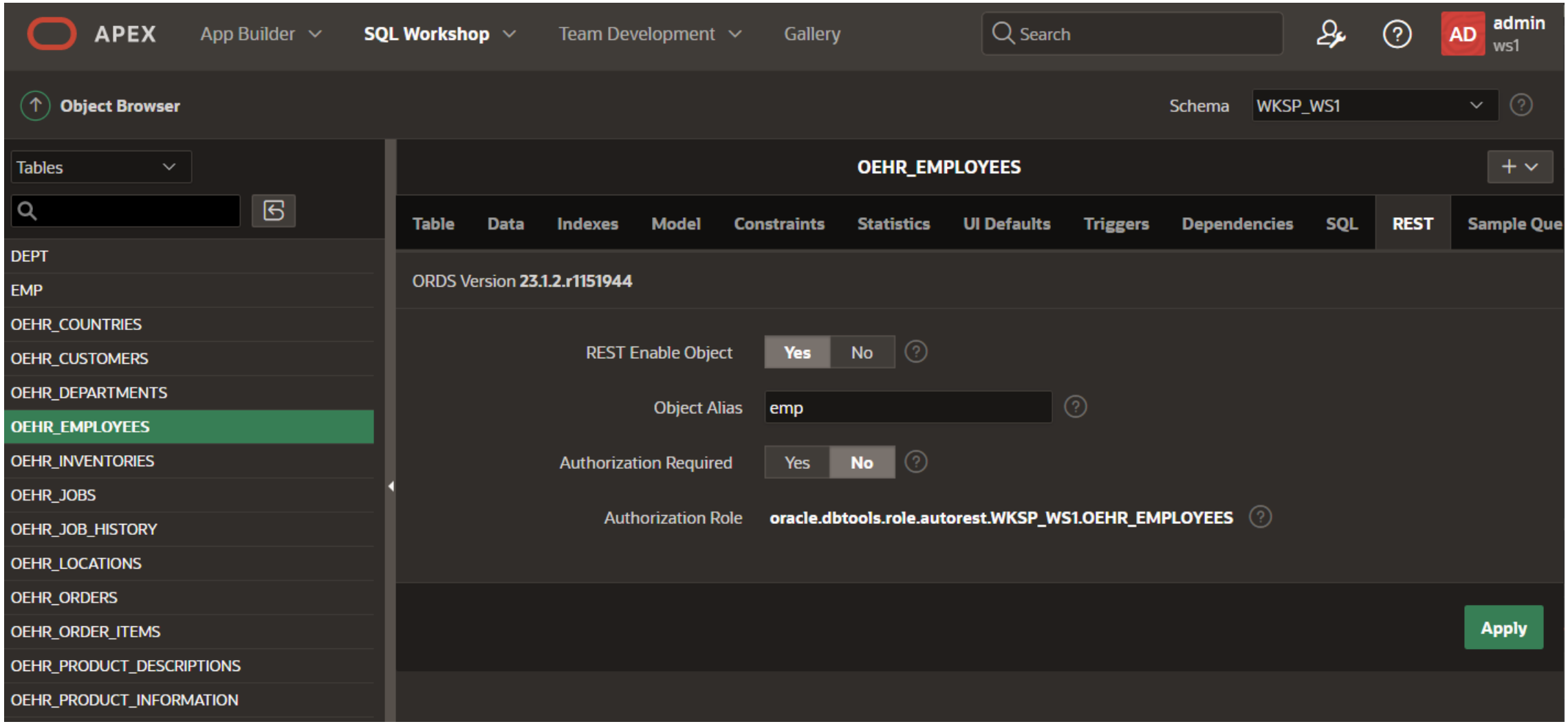
How to check if Schema is ORDS Enabled or not?

- By querying USER_ORDS_SCHEMAS and ORDS_METADATA.ORDS_SCHEMAS we can check the status of schema.



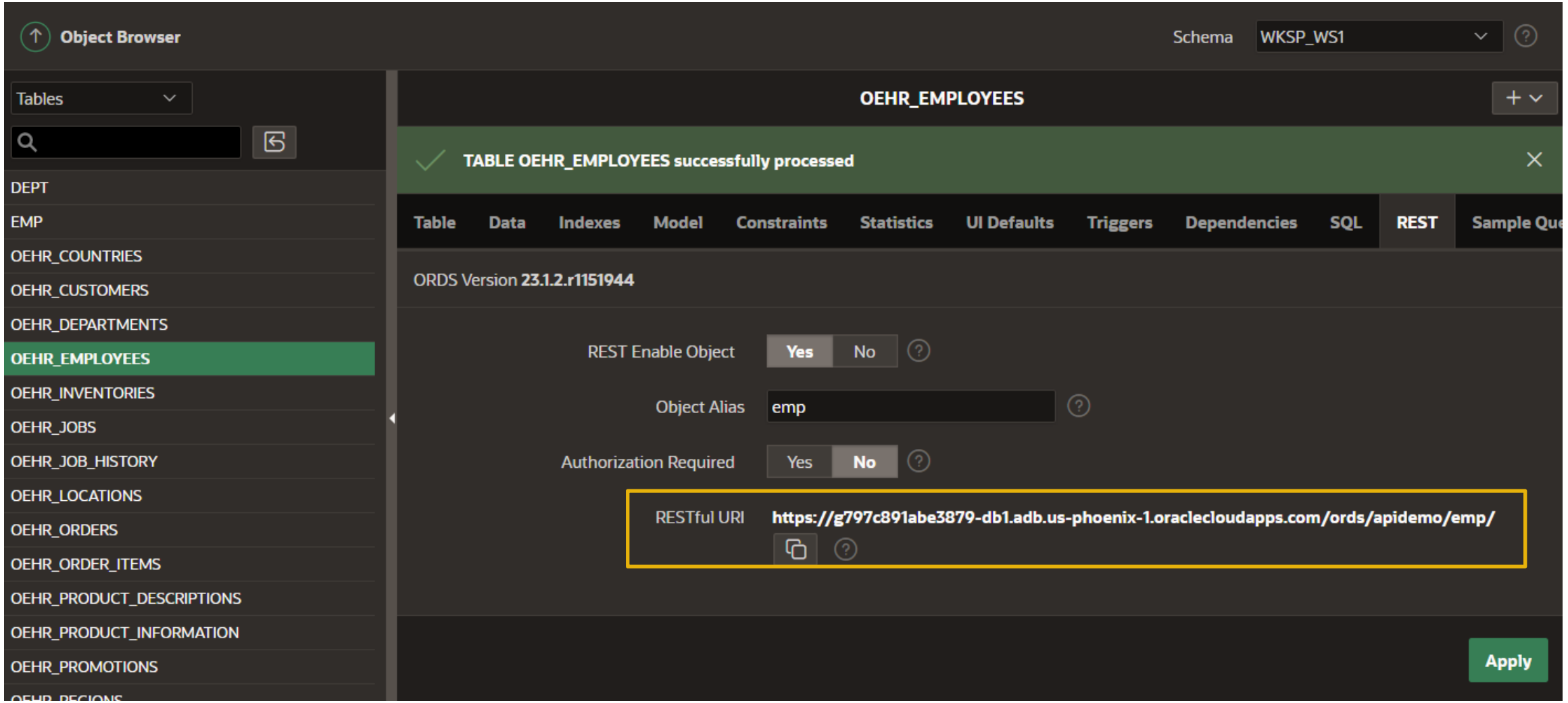
Demo

AUTOREST Using APEX



*Rutvik Prajapati , @rutveek

AUTOREST Using APEX



*Rutvik Prajapati , @rutveek

AUTOREST Using ORDS API

```
ORDS.ENABLE_OBJECT (
                    p_enabled          => TRUE,  -- Is AutoREST Access enabled
                    p_schema           => 'WKSP_WS1',     -- Schema owning the object
                    p_object           => 'OEHR_EMPLOYEES'  -- Object Name
                    p_object_type      => 'TABLE', -- Object Type
                   p_object_alias      => 'emp',   --- Object Alias
                   p_auto_rest_auth    => FALSE
                );
```

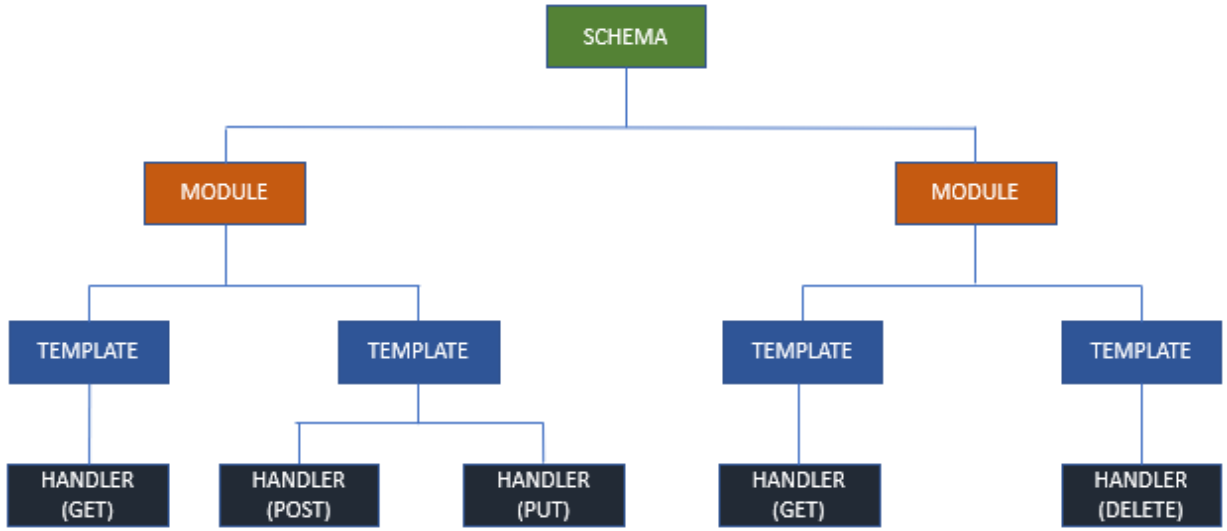
Demo

Manual REST Service Creation

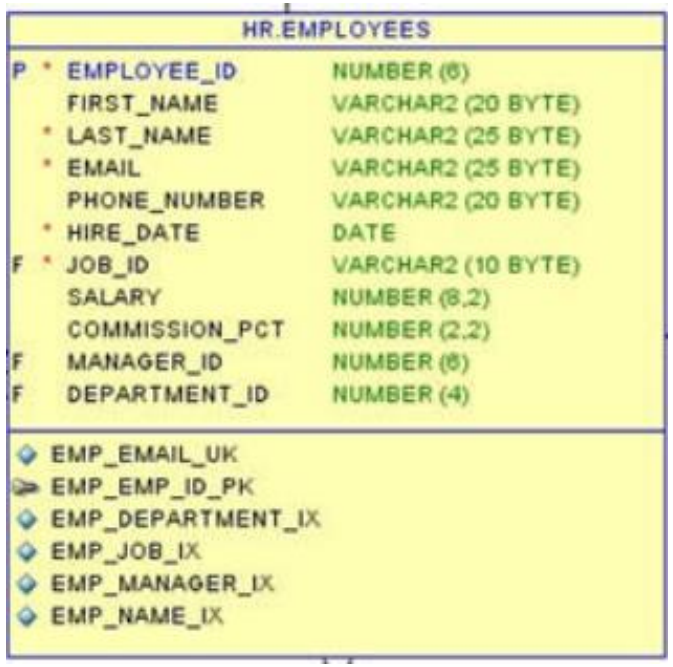
- Requires you to specify the SQL or PL/SQL to support required actions
- More effort but more flexibility
- Can customize columns, join across multiple tables, etc.
- Validate incoming data using PL/SQL
- Include complex logic to decide what taction to take

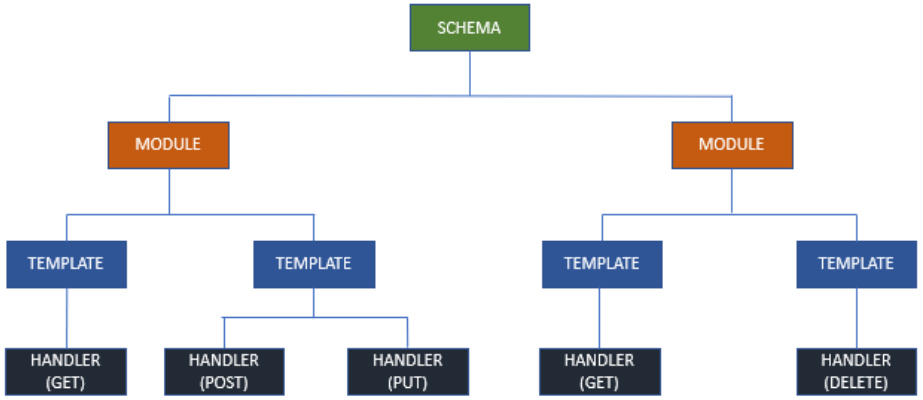
Manual REST Service Creation

- Create Module
- Create Template
- Create Handler
  - For each action we need to create a new handler
    - GET
    - POST
    - PUT
    - DELETE



Manual REST Service Creation





Module : hr/v1/

Template : employees/:id?

Handler:

GET : Get all emps or single emps

POST : Create single emp record

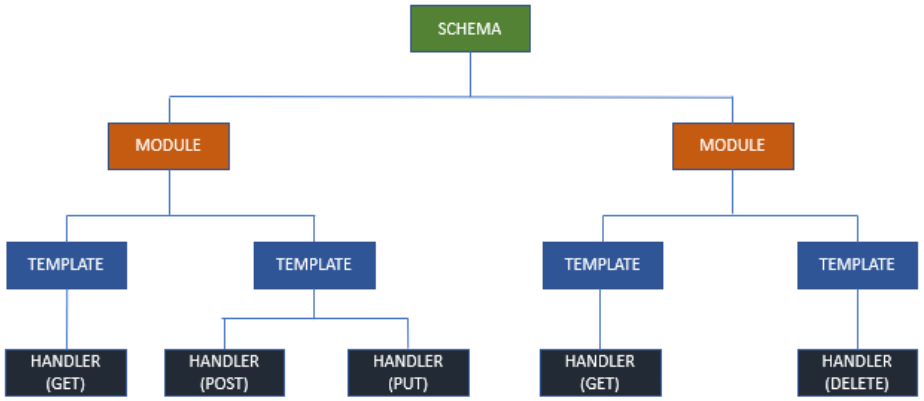
PUT : Update single emp record

DELETE : Delete single emp record

Manual REST Service Creation

Added 3 columns to Employees table

MIME_TYPE	VARCHAR2(256)	Yes
FILE_NAME	VARCHAR2(256)	Yes
CONTENT	BLOB	Yes



Module : hr/v2/

Template : employees/

Handler:

GET : Get all emps

POST : Create single emp record

Template : employees/:id

Handler:

GET : Get single emp record

PUT : Update single emp record

DELETE : Delete single emp record

Template : employees/:id/photo/

Handler:

GET : Get photo of emp

PUT : Update emp photo

Manual REST Service Creation

ORDS Parameters

- Implicit
  - :body, :body_text, :status_code, :content_type, and more…
- Explicit
  - User defined

Find out more about implicit parameters:

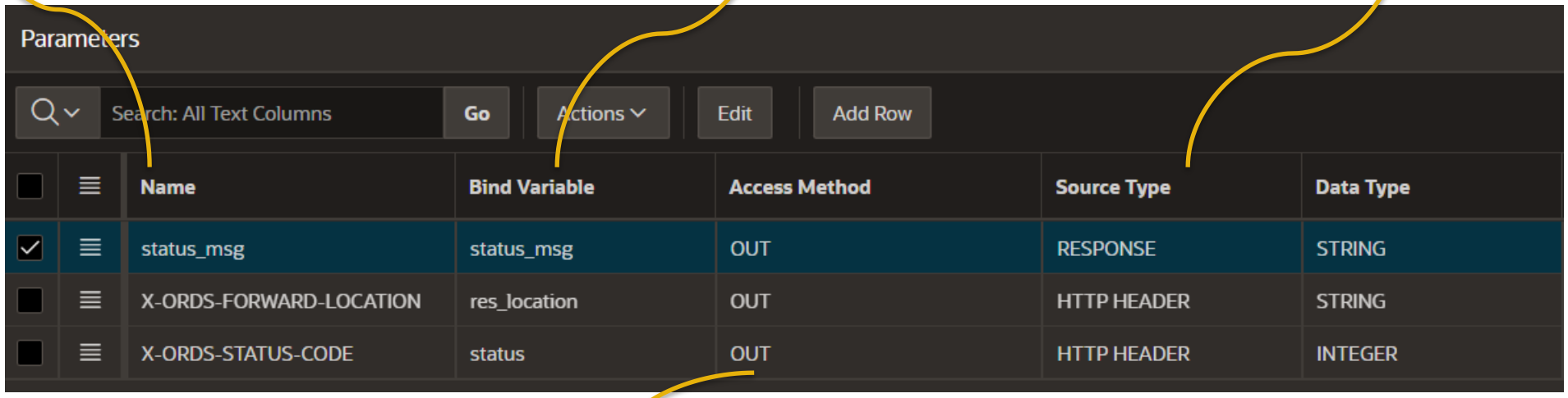
https://docs.oracle.com/en/database/oracle/oracle-rest-data-services/22.1/orddg/implicit-parameters.html#GUID-B0BB1694-715C-4948-84A5-307EA3868063

Manual REST Service Creation

Parameter name that you can see in request of response.

Bind variable name that you can reference in SQL or PL/SQL code.

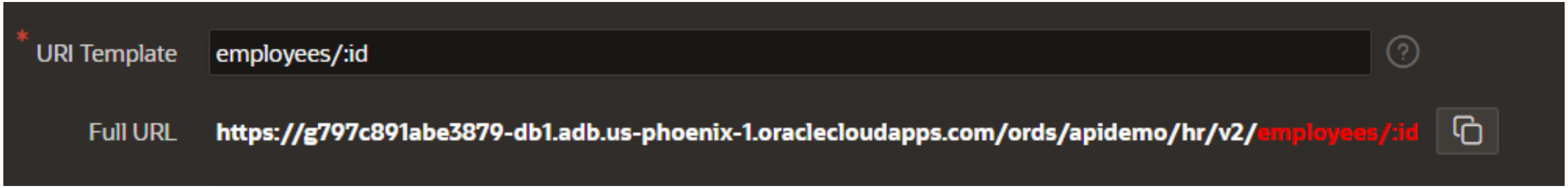
Indicates the source type of parameters.
IN: URI, HTTP Header
OUT: Response, HTTP Header



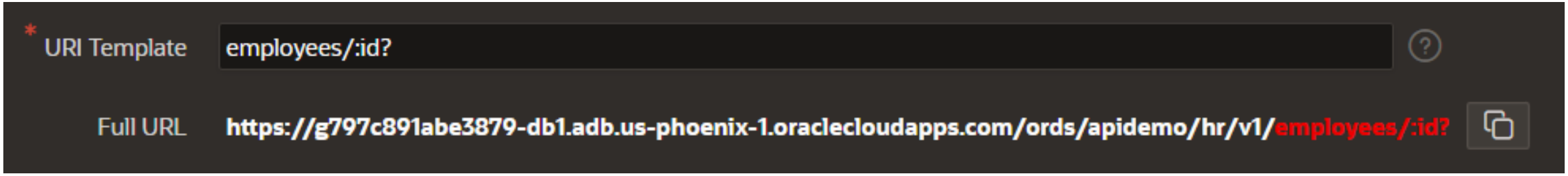
Access Method: IN or OUT
IN : to receive the value from the client during the request
OUT : to send the value to the client during response

Manual REST Service Creation

:id is a mandatory bind variable



:id is an optional bind variable

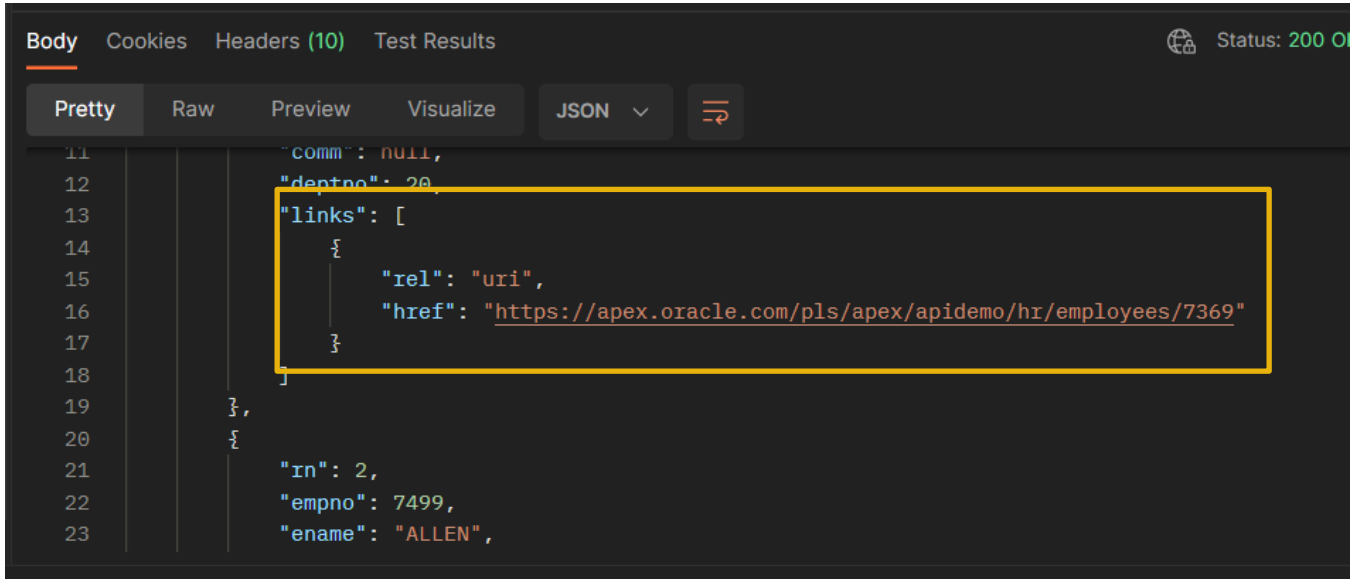


Manual REST Service Creation

Column that starts with $: ORDS will generate a link for that JSON attribute.

Source

```
select empno "$uri",
       rn,
       empno,
       ename,
       job,
       hiredate,
       mgr,
       sal,
       comm,
       deptno
  from (
       select emp.*
            , row_number() over (order by empno) rn
         from emp
       ) tmp
```



Manual REST Service Creation

Parameter that starts with $: ORDS will generate a link for that JSON attribute.

```
1  BEGIN
2     HR_API_V1.CREATE_EMP(
3        P_EMPLOYEE_ID     => :employee_id,
4        P_FIRST_NAME      => :first_name,
5        P_LAST_NAME       => :last_name,
6        P_EMAIL           => :email,
7        P_PHONE_NUMBER    => :phone,
8        P_HIRE_DATE       => :hire_date,
9        P_JOB_ID          => :job_id,
10       P_SALARY          => :salary,
11       P_COMMISSION_PCT  => :comm,
12       P_MANAGER_ID      => :manager_id,
13       P_DEPARTMENT_ID   => :department_id,
14       P_RES_LOC         => :res_location,
15       P_STATUS_CD       => :status,
16       P_STATUS_MSG      => :status_msg
17    );
18
19 END;
```

Parameters

Search: All Text Columns | Go | Actions | Edit | Add Row

Name	Bind Variable	Access Method	Source Type	Data Type
$self	res_location	OUT	RESPONSE	STRING
status_msg	status_msg	OUT	RESPONSE	STRING
X-ORDS-STATUS-CODE	status	OUT	HTTP HEADER	INTEGER

Securing REST Service

Mainly two types of authentications

- Basic Auth
  - Protect a service using username and password
  - Database Users
  - APEX Workspace User

- OAUTH2
  - More Secure
  - Involves Provider (Server) and Consumer (Client) of the service
  - Owner of the service creates a “Client”, assigns it privileges and provides details to the consumer

Securing REST Service using OAuth2

Server Side

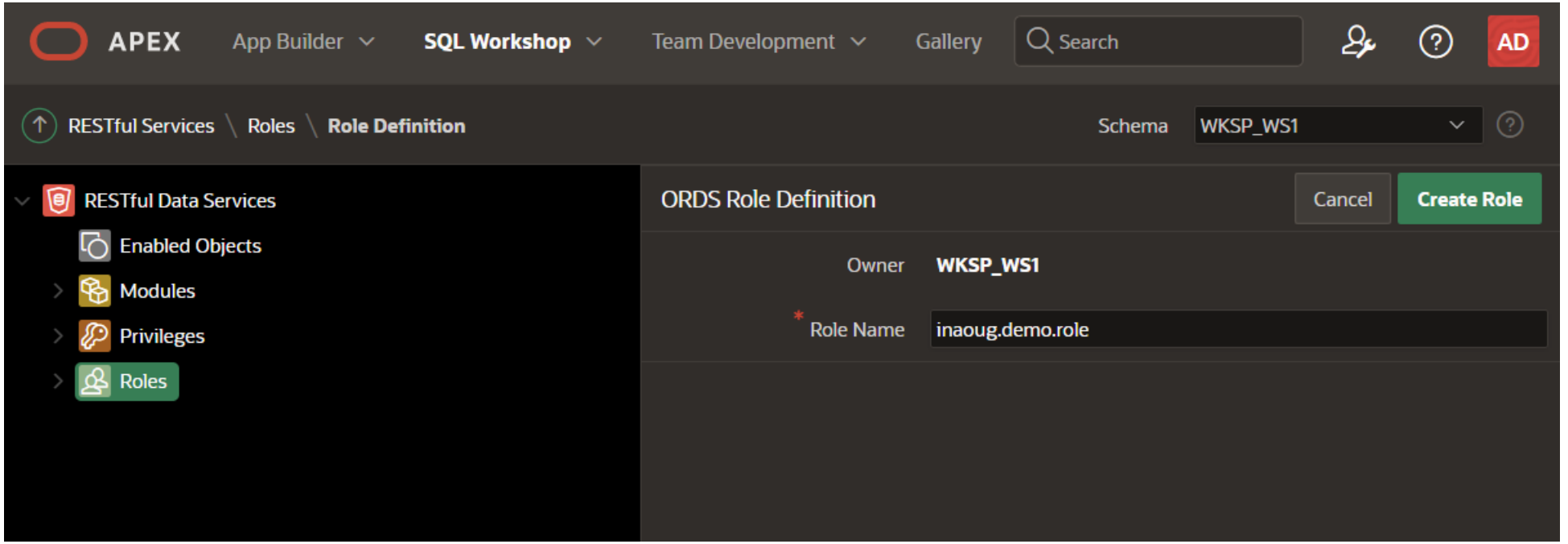
- Define the Module
- Create a Role
- Create a Privilege
- Create Client Credentials
- Link Module, Role, Privilege and Client

Client Side

- Authenticate as Client
- Validate/Retrieve Token
- Use token to access REST Service

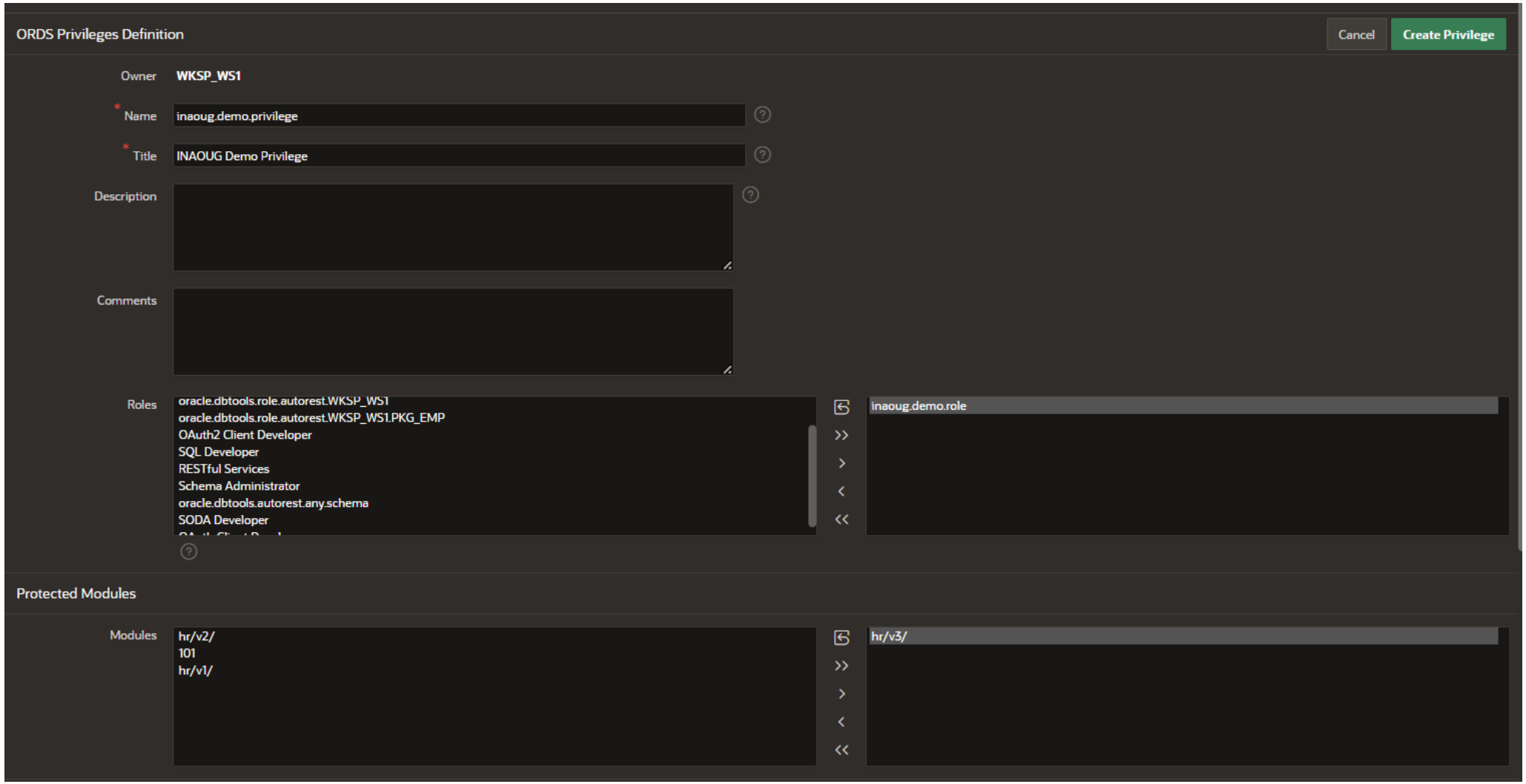
Securing REST Service using OAuth2

Create ORDS Role



Securing REST Service using OAuth2

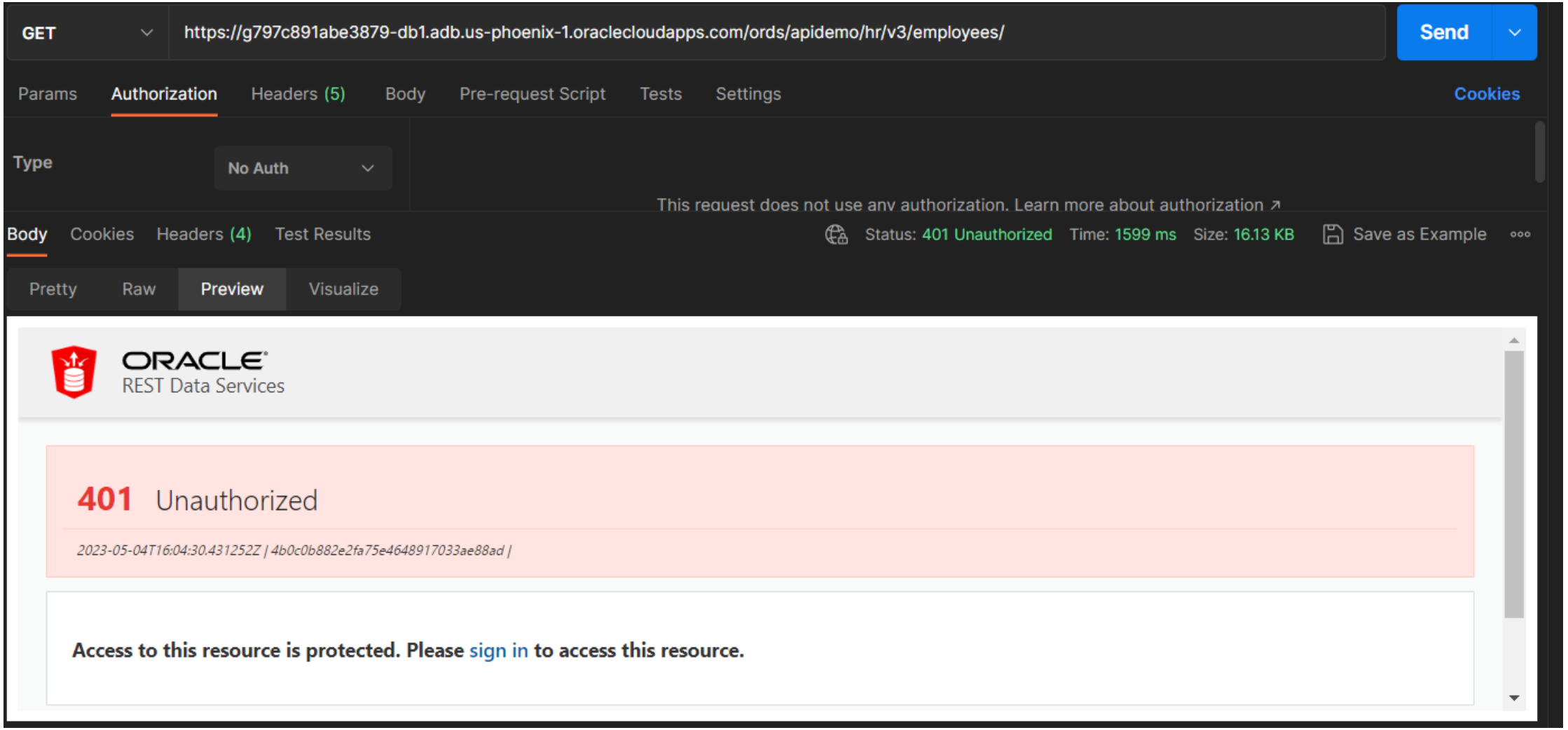
Create Privilege, Assign Role and Protect Modules



*Rutvik Prajapati , @rutveek

Securing REST Service using OAuth2

After this step, our webservice is protected and we cannot directly access it



*Rutvik Prajapati , @rutveek

Securing REST Service using OAuth2

Create client using PL/SQL API

```
BEGIN
    OAUTH.create_client(
        p_name          => 'rutvik.prajapati',
        p_grant_type     => 'client_credentials',
        p_owner         => 'WKSP_WS1',
        p_description    => 'Internal client to access API',
        p_support_email   => 'rutveek@gmail.com',
        p_privilege_names => 'inaoug.demo.privilege'
    );

    COMMIT;
END;
```

Assign Role to client

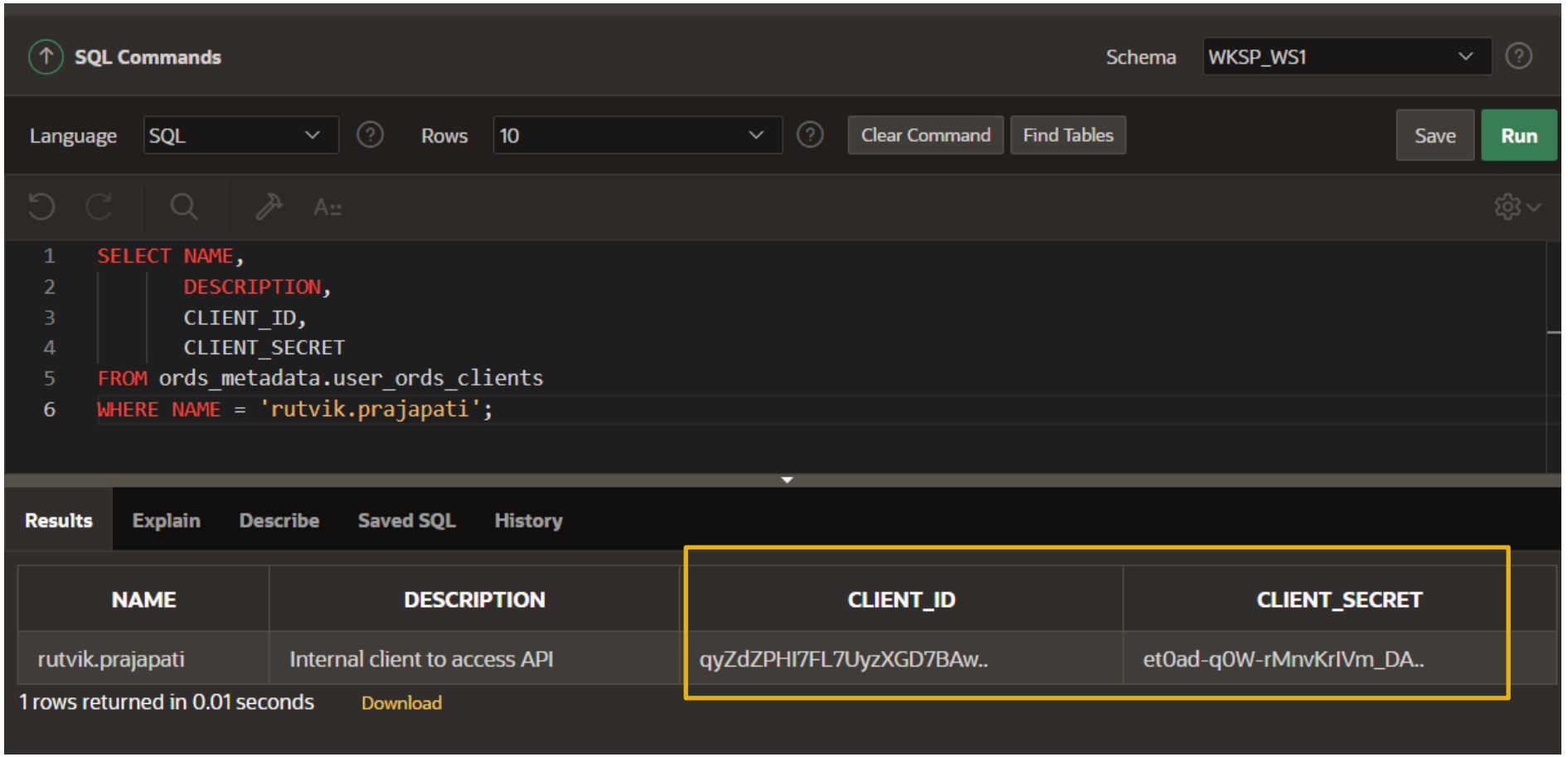
```
BEGIN

  OAUTH.grant_client_role(
   p_client_name => 'rutvik.prajapati',
   p_role_name   => 'inaoug.demo.role'
  );

  COMMIT;
END;
```

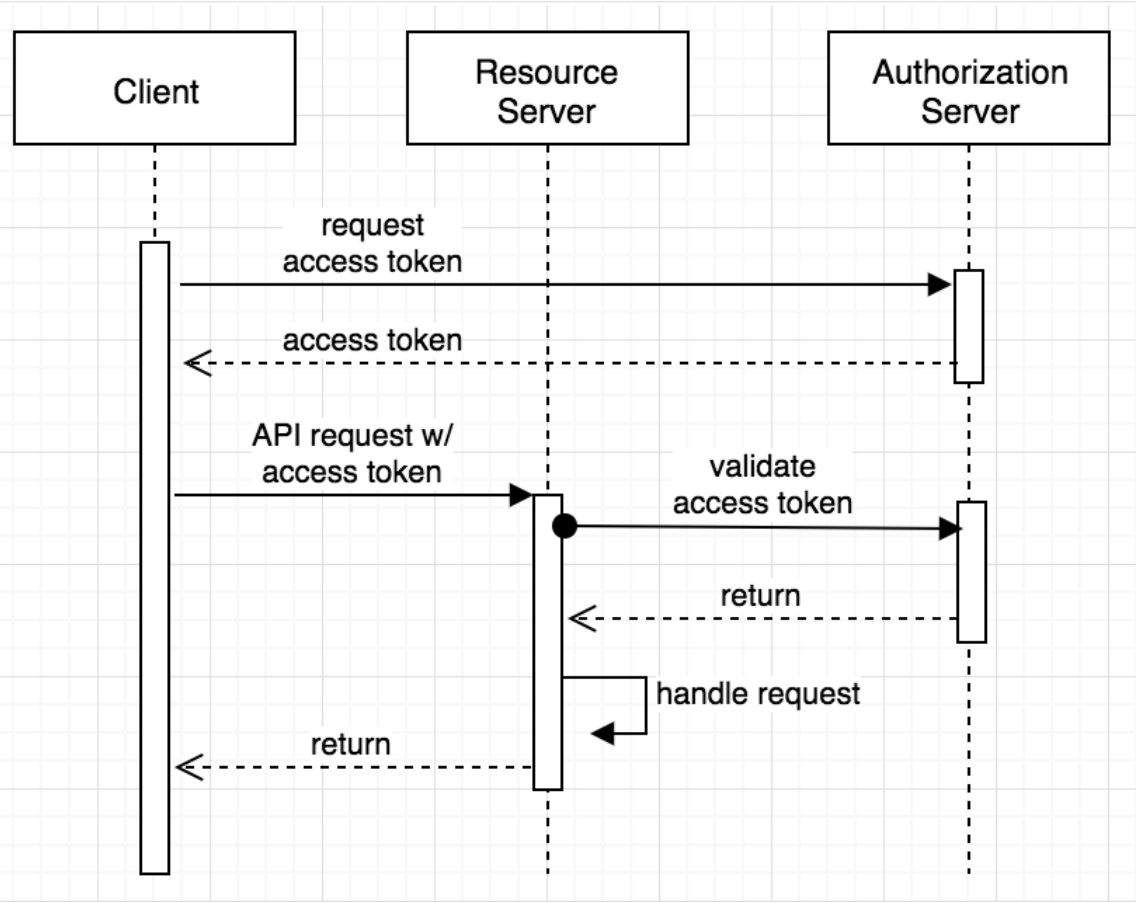
Securing REST Service using OAuth2

- We created client and assign a role to that client. Next, we need to provide client id and client secret to the consumer(client) of this API.
- For that query ords_metadata.user_ords_clients view



Securing REST Service using OAuth2

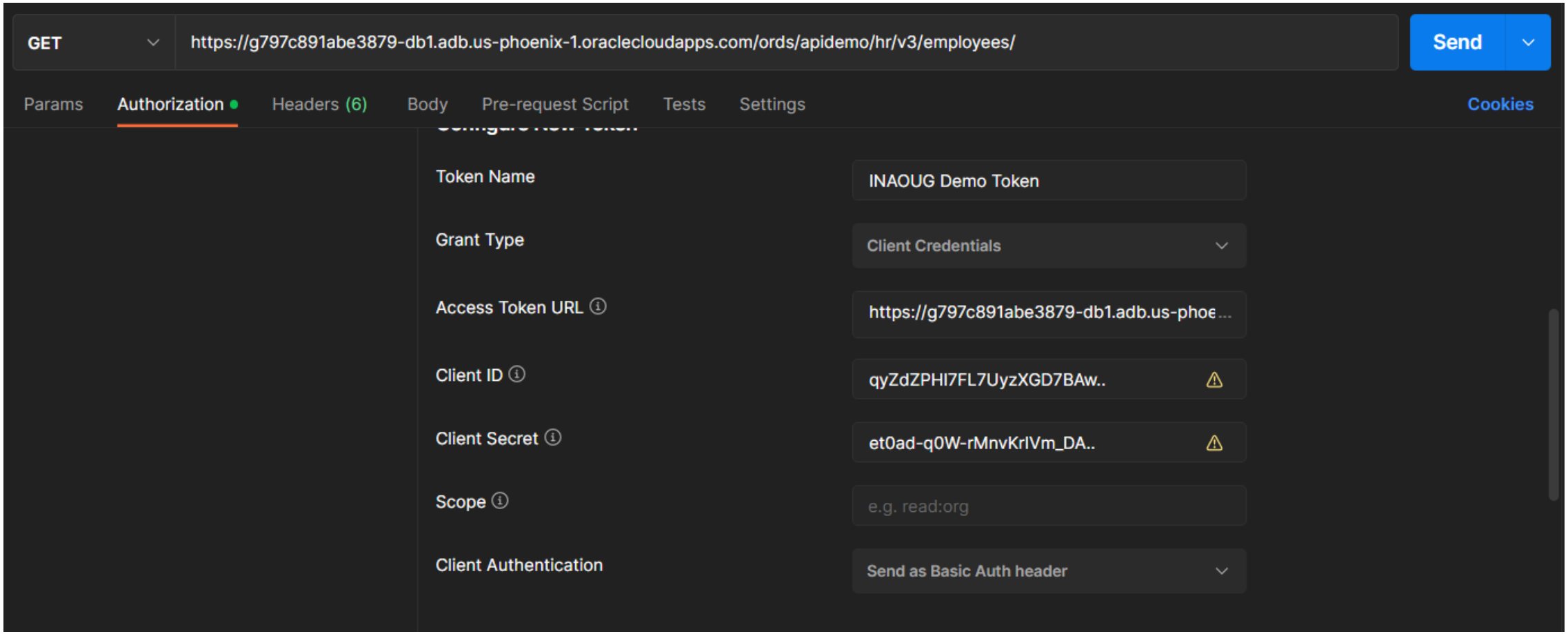
- OAuth2 Client Credential Flow



Access Token URL

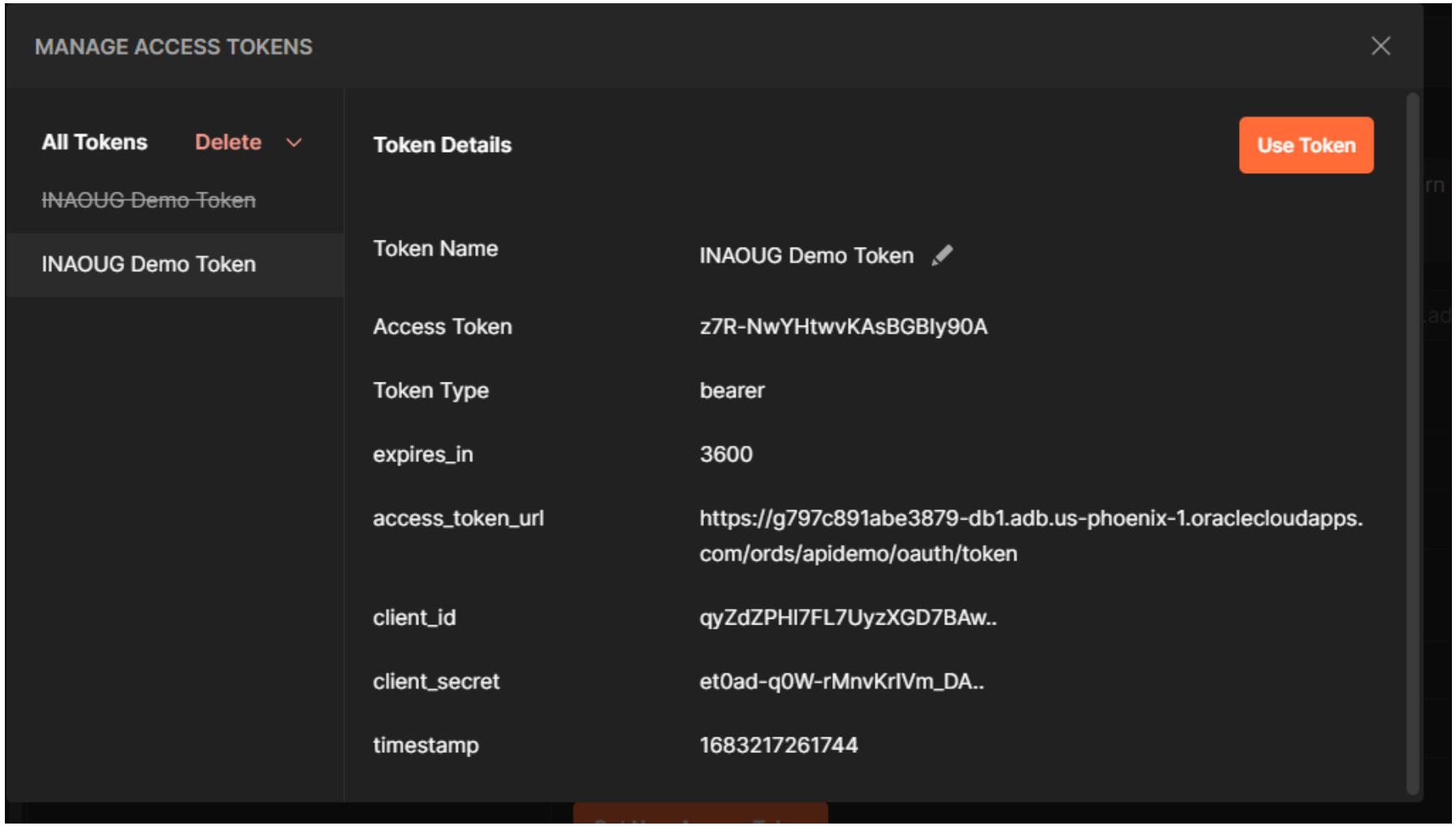
http(s)://<server>/<ords_alias>/<schema_alias>/oauth/token

Securing REST Service using OAuth2

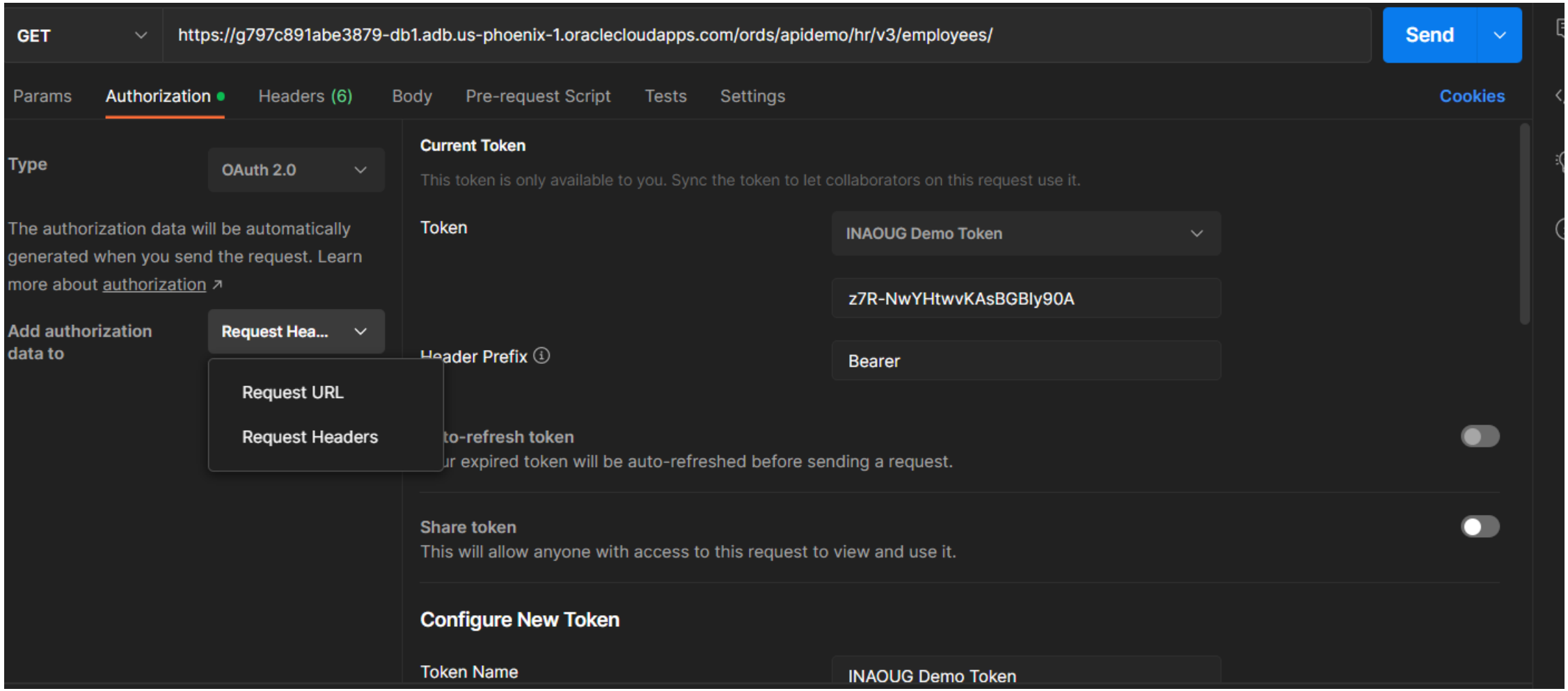


*Rutvik Prajapati , @rutveek

Securing REST Service using OAuth2

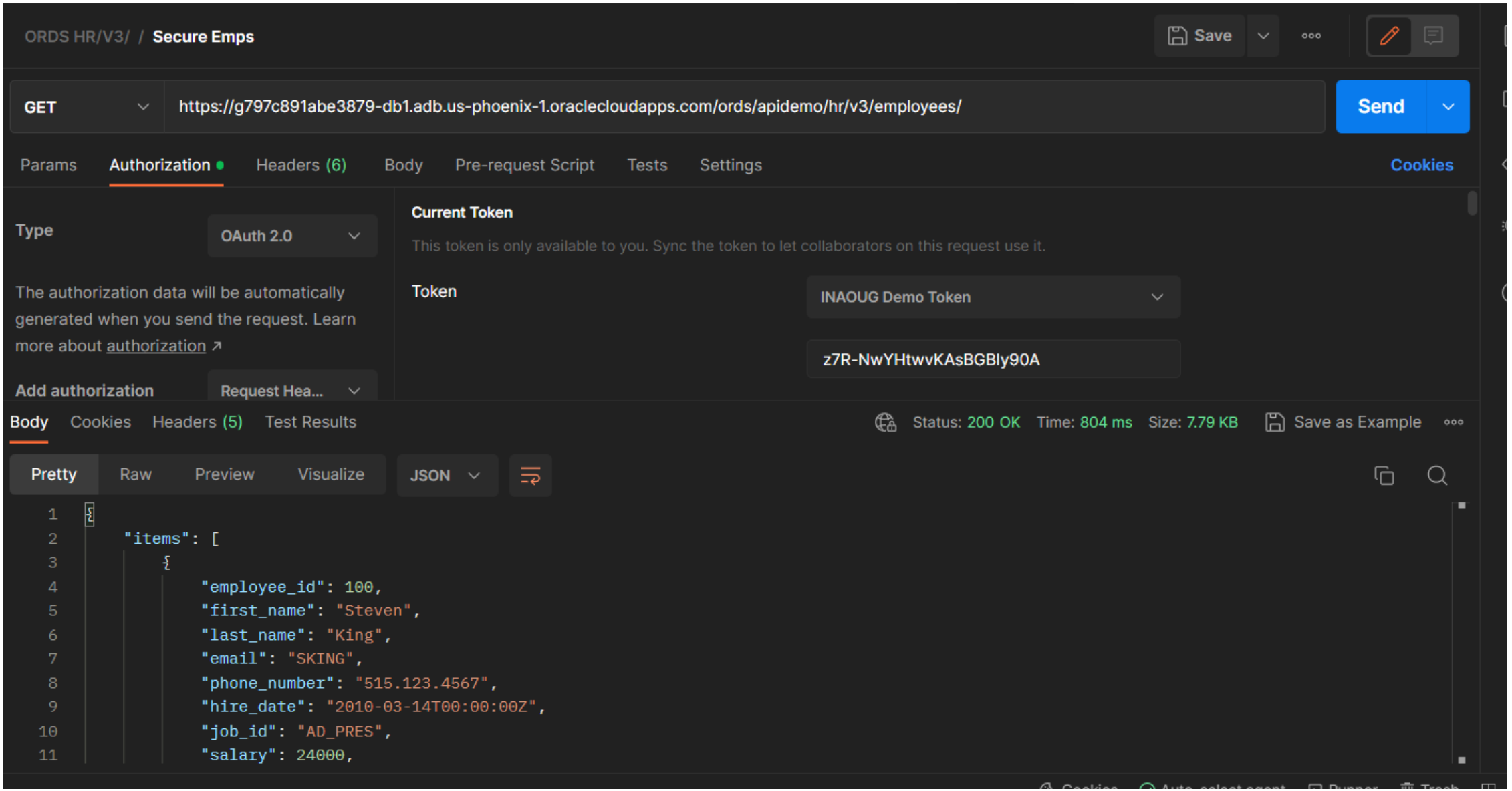


Securing REST Service using OAuth2



*Rutvik Prajapati , @rutveek

Securing REST Service using OAuth2



Demo

Learn More

https://www.thatjeffsmith.com/archive/2020/06/ords-101-working-with-pl-sql/

https://www.thatjeffsmith.com/archive/2019/03/calling-a-pl-sql-function-via-ords/

https://www.thatjeffsmith.com/oracle-database-rest-apis/

THANK YOU

*Rutvik Prajapati , @rutveek

Q & A