

Creando una REST API en menos de 1 hora con Oracle Database Actions

Pre-requisitos

- Una instancia de ADB-S (Puede ser tipo "Always Free")
- Un navegador de internet
- Una terminal tipo UNIX (Puede ser "Git Bash" en Windows)
- cURL debe estar instalado en la terminal tipo UNIX que se pretenda utilizar
- Opcionalmente tener instalado **jq**, que es una herramienta de línea de comandos para formatear y filtrar JSON

Propósito

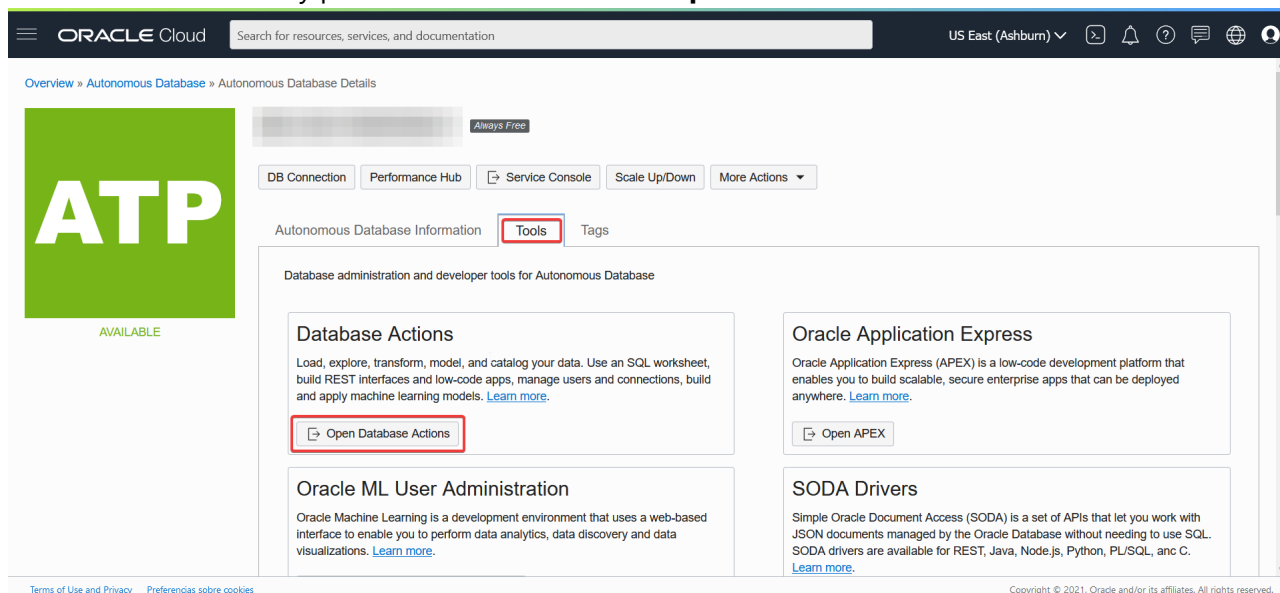
Este documento te guiará paso a paso por la creación de una REST API para una aplicación ficticia de recordatorios (TODOs en inglés) que tiene dos tablas:

- **USERS**: Donde almacenaremos nuestros usuarios
- **TODOS**: Donde almacenaremos nuestras tareas

En este documento proporcionaremos dos archivos con datos de ejemplo para generar las tablas que necesitamos y posteriormente procederemos a habilitar las tablas para REST usando Oracle Database Actions.

Crear un nuevo usuario (Paso a Paso)

1. Accede a la consola de servicio de tu instancia de Autonomous Database
2. Selecciona la tab **Tools** y posteriormente haz click en **Open Database Actions**



3. Ingresa **ADMIN** y haz click en **Siguiente**

ORACLE Database Actions

Nombre de usuario

1 ADMIN

2 Siguiente

Acerca de Oracle | Contacto | Avisos legales | Términos de uso | Derechos de Privacidad

Copyright © 2009, 2021 Oracle y/o sus filiales. Todos los derechos reservados.

4. Ingresa tu password y haz click en **Conectar**

ORACLE Database Actions

Nombre de usuario

ADMIN

Contraseña

1

2 Conectar

Acerca de Oracle | Contacto | Avisos legales | Términos de uso | Derechos de Privacidad

Copyright © 2009, 2021 Oracle y/o sus filiales. Todos los derechos reservados.

5. Abre **Usuarios de Base de Dato**

ORACLE Database Actions

ADMIN

Desarrollo

- SQL: Ejecutar consultas y scripts, y crear objetos de base de datos
- MODELADOR DE DATOS: Crear diagramas relacionales para objetos de base de datos
- REST: Desplegar API de REST para la base de datos
- JSON: Gestionar la base de datos de documentos JSON
- APEX: Crear aplicaciones web rápidamente

Administración

- USUARIOS DE BASE DE DATOS**: Crear, editar privilegios y otros parámetros y activar usuarios de base de datos para REST

Herramientas de datos

- CARGA DE DATOS: Cargar datos de archivos locales o bases de datos remotas o acceder a ellos
- CATÁLOGO: Aprenda sobre la dependencia de los datos y el impacto de los cambios
- ESTADÍSTICAS DE DATOS: Descubrir anomalías, valores atípicos y patrones ocultos en los datos
- MODELOS DE NEGOCIO: Crear modelos de negocio de rendimiento y análisis

Supervisión

- HUB DE RENDIMIENTO: Ver datos de rendimiento consolidados para la base de datos

Introducción

Servicios Web RESTful
Despliegue API de REST para la base de datos de Oracle - GET, PUT, POST y DELETE de forma segura usando HTTPS con los procedimientos almacenados y datos de Oracle.

Cargue datos
Rellene las tablas existentes o compile tablas nuevas a partir de archivos locales (Avro, JSON, XML, CSV o Excel) usando nuestro asistente de carga de datos.

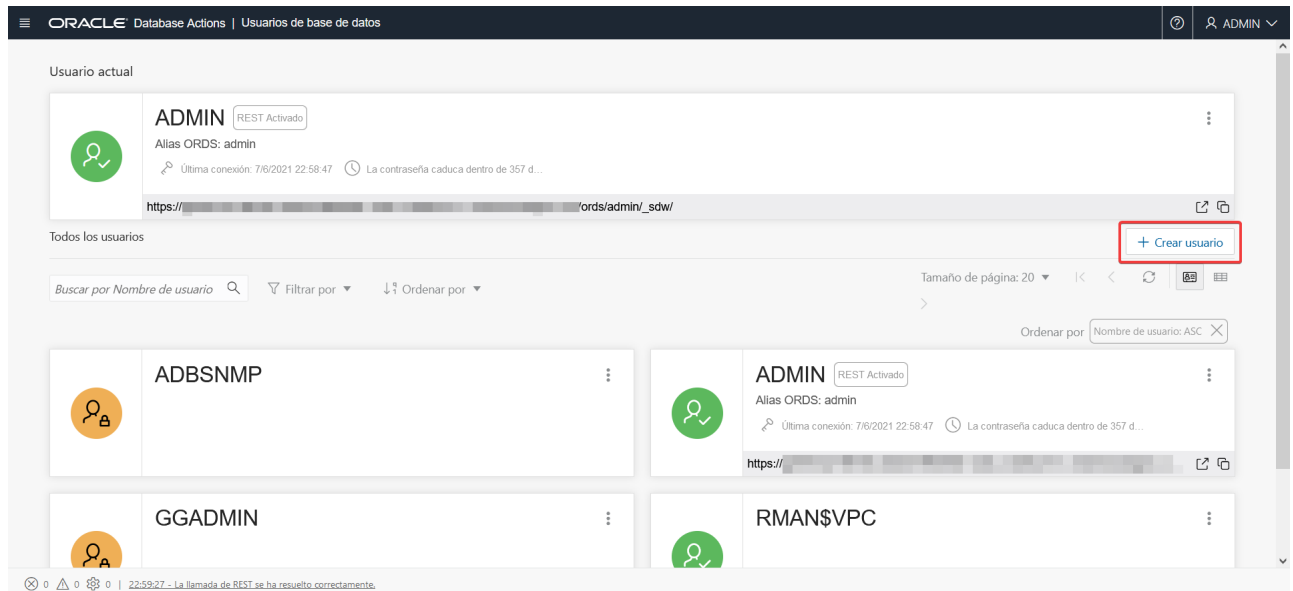
JSON
Cree recopilaciones y documentos, agregue, edite, suprima y examine sus documentos, y vea las guías de datos de JSON.

Disponible de forma local
Ahora SQL Developer Web también está disponible de forma local en las bases de datos Oracle.

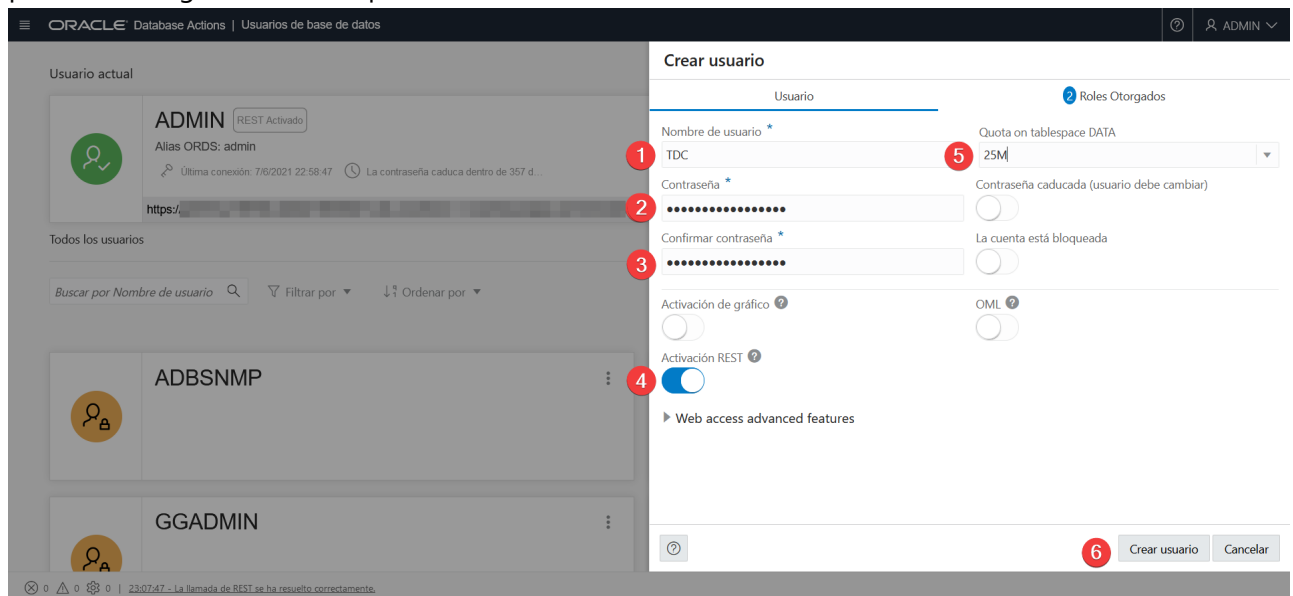
¿Necesita Ayuda?
Documentación
Foro de la comunidad de SQL Developer
Twitter de SQL Developer

22:58:49 - La llamada de REST se ha resuelto correctamente.

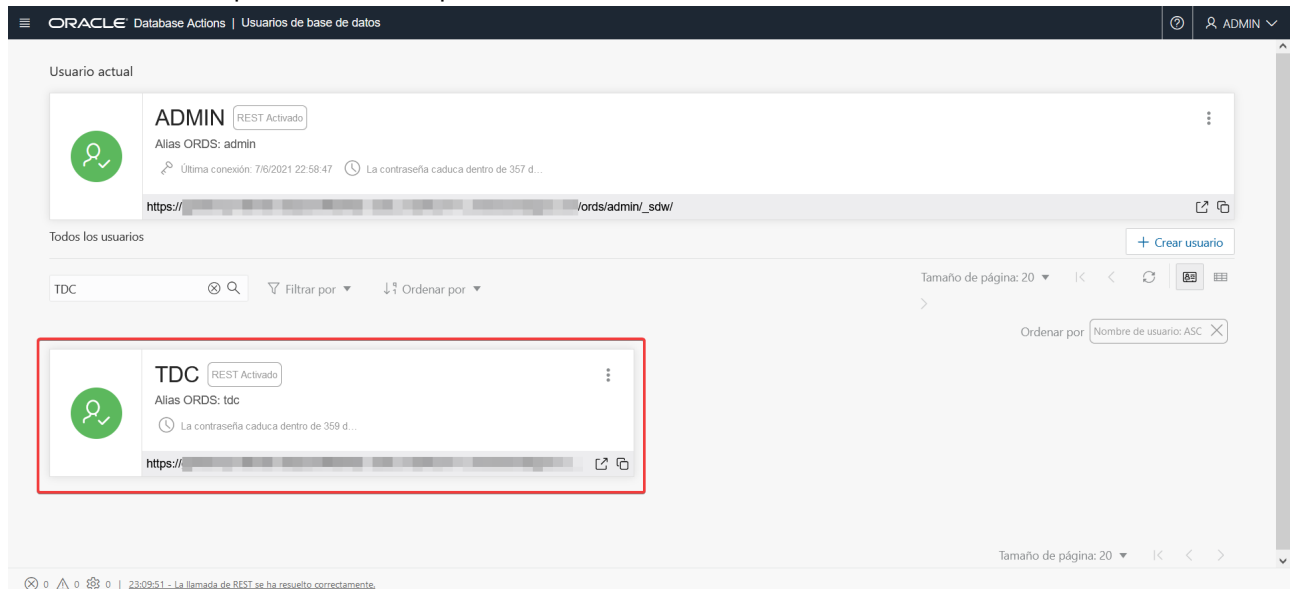
6. Haz click en **Crear usuario**



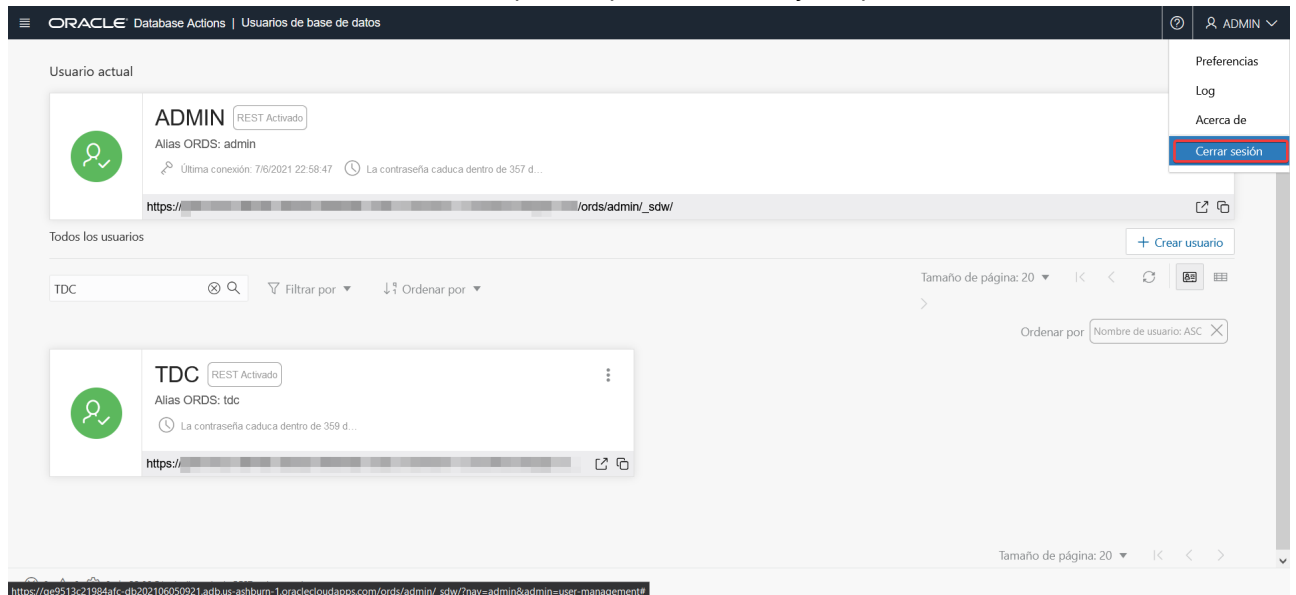
7. Llena los detalles del usuario y haz click en **Crear Usuario**. Es muy importante habilitar la opción de **Activación REST** y asignarle una cuota diferente a la predeterminada al usuario, de lo contrario no podremos cargar datos al esquema



8. El nuevo usuario aparecerá en el reporte una vez creado correctamente

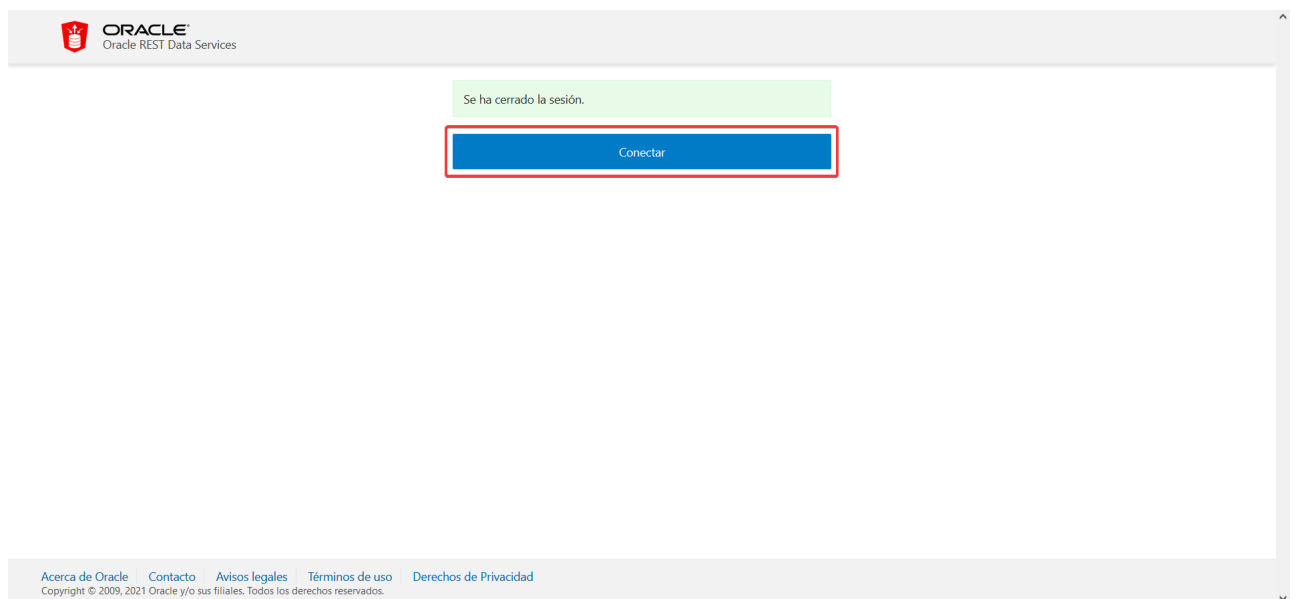


9. Haz click en el botón de usuario de la esquina superior derecha y después haz click en **Cerrar sesión**



Carga de Datos (Paso a paso)

1. Descarga el archivo [users.csv](#) del repositorio
2. Descarga el archivo [todos.csv](#) del repositorio
3. Haz click en **Conectar**



4. Ingresa el nuevo nombre de usuario que creaste anteriormente y haz click en **Siguiente**

ORACLE Database Actions

Nombre de usuario

1 TDC

2 Siguiente

[Acerca de Oracle](#) [Contacto](#) [Avisos legales](#) [Términos de uso](#) [Derechos de Privacidad](#)

Copyright © 2009, 2021 Oracle y/o sus filiales. Todos los derechos reservados.

5. Ingresa el password y haz click en **Conectar**

ORACLE Database Actions

Nombre de usuario

TDC

Contraseña

1

2 Conectar

[Acerca de Oracle](#) [Contacto](#) [Avisos legales](#) [Términos de uso](#) [Derechos de Privacidad](#)

Copyright © 2009, 2021 Oracle y/o sus filiales. Todos los derechos reservados.

6. Haz click en **SQL**

ORACLE Database Actions

Desarrollo

SQL
Ejecutar consultas y scripts, y crear objetos de base de datos

MODELADOR DE DATOS
Crear diagramas relacionales para objetos de base de datos

REST
Desplegar API de REST para la base de datos

JSON
Gestionar la base de datos de documentos JSON

Administración

USUARIOS DE BASE DE DATOS
Crear, editar privilegios y otros parámetros y activar usuarios de base de datos para REST

Supervisión

SUPERVISIÓN SQL EN TIEMPO REAL
Supervisar consultas SQL ejecutadas en tiempo real

Introducción

Servicios Web RESTful
Despliegue API de REST para la base de datos de Oracle - GET, PUT, POST y DELETE de forma segura usando HTTPS con los procedimientos almacenados y datos de Oracle.

Cargue datos
Rellene las tablas existentes o compile tablas nuevas a partir de archivos locales (Avro, JSON, XML, CSV o Excel) usando nuestro asistente de carga de datos.

JSON
Cree recopilaciones y documentos, agregue, edite, suprima y examine sus documentos, y vea las guías de datos de JSON.

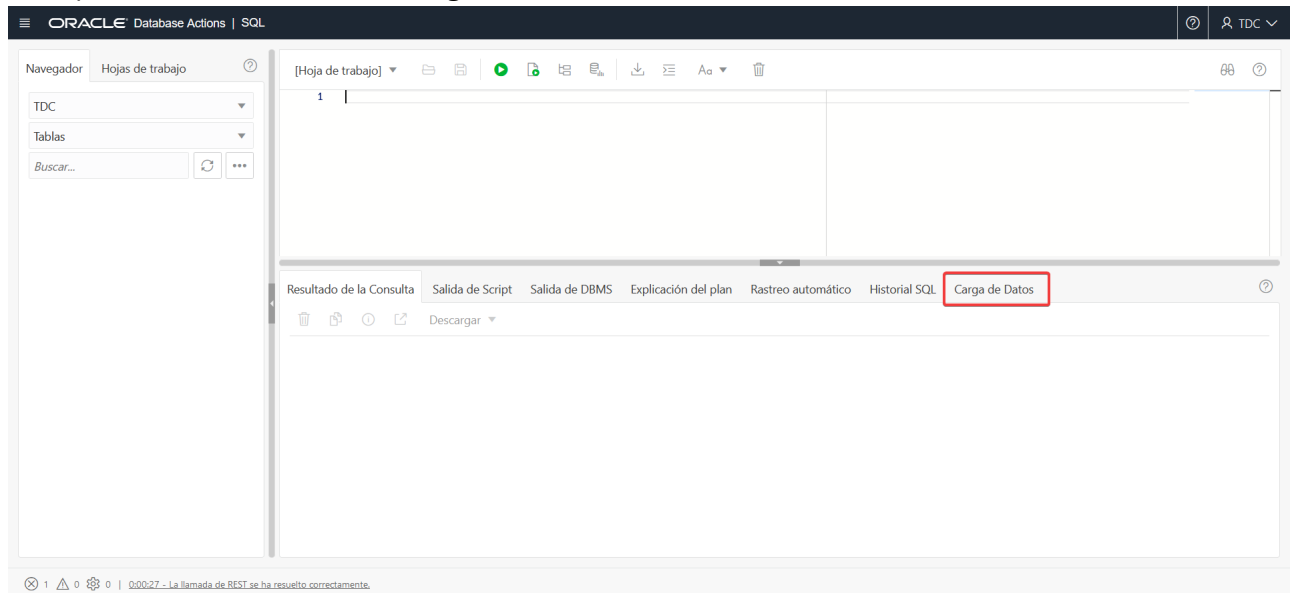
Disponible de forma local
Ahora SQL Developer Web también está disponible de forma local en las bases de datos Oracle.

¿Necesita Ayuda?

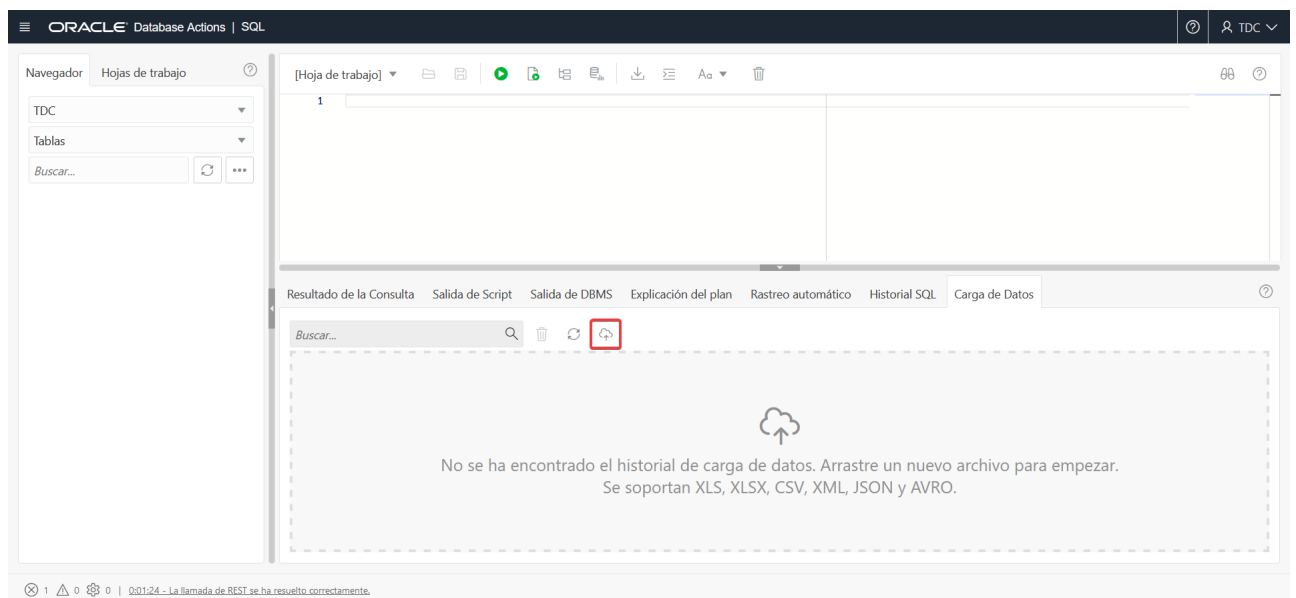
Documentación
Foro de la comunidad de SQL Developer
Twitter de SQL Developer

23:31:38 - La llamada de REST se ha resuelto correctamente.

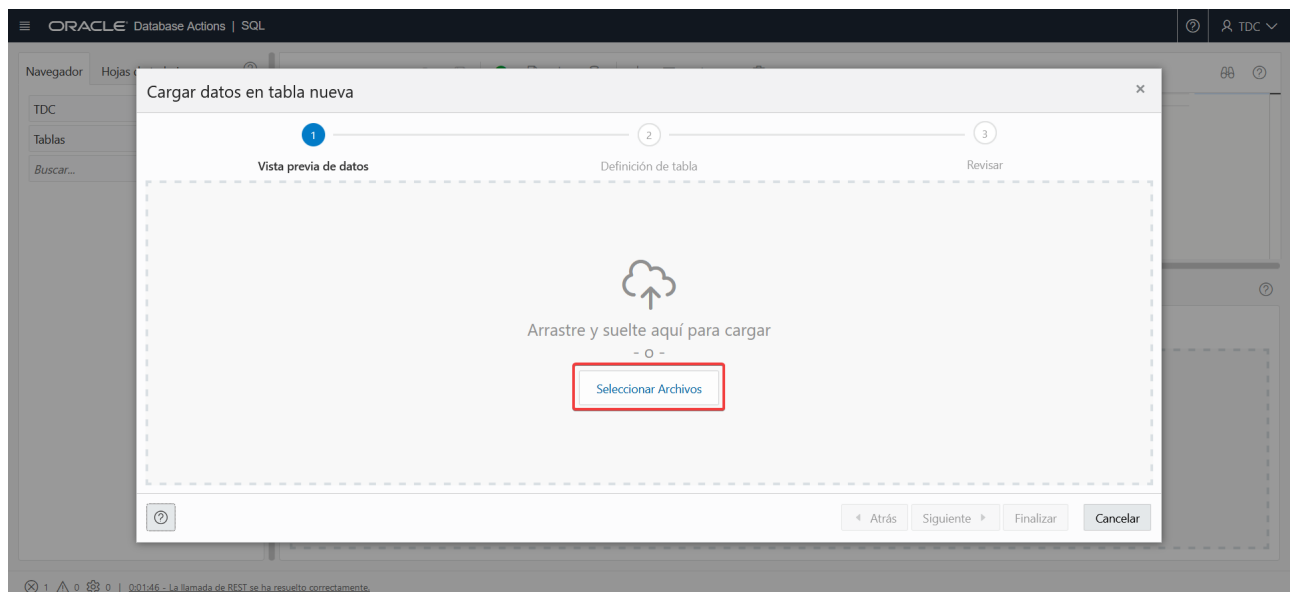
7. En el panel inferior, haz click en **Carga de datos**



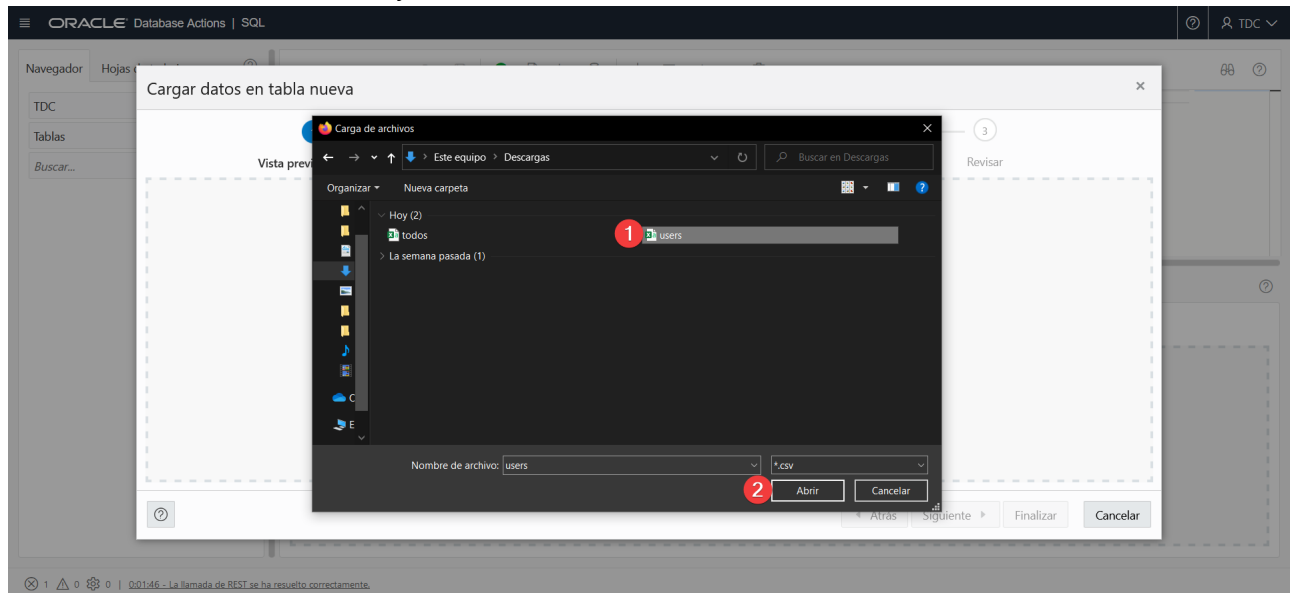
8. Haz click en el botón con el ícono de nube



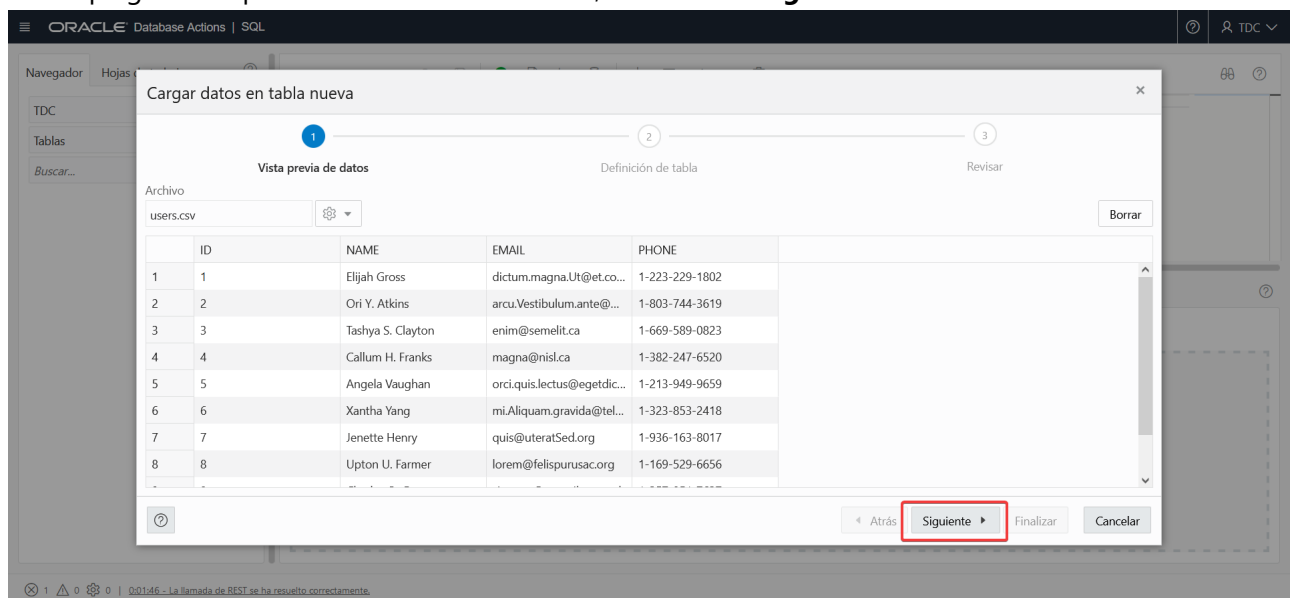
9. Haz click en **Seleccionar Archivos**



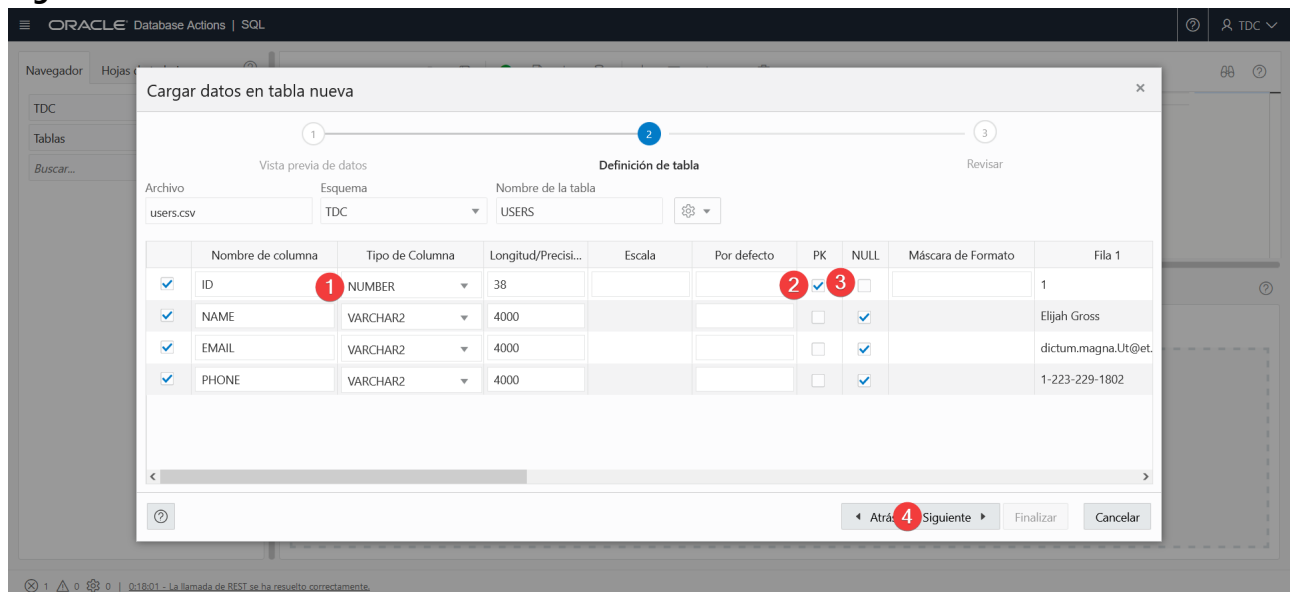
10. Selecciona el archivo **users.csv** y haz click en **Abrir**



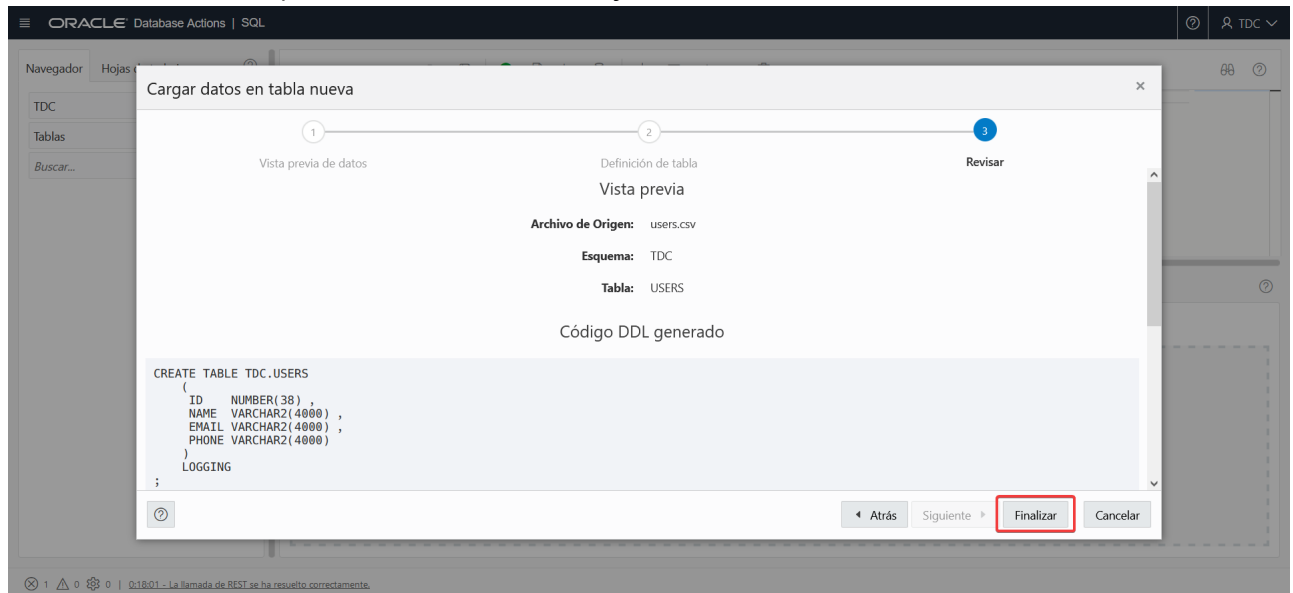
11. Se desplegará una previsualización de los datos, haz click en **Siguiente**



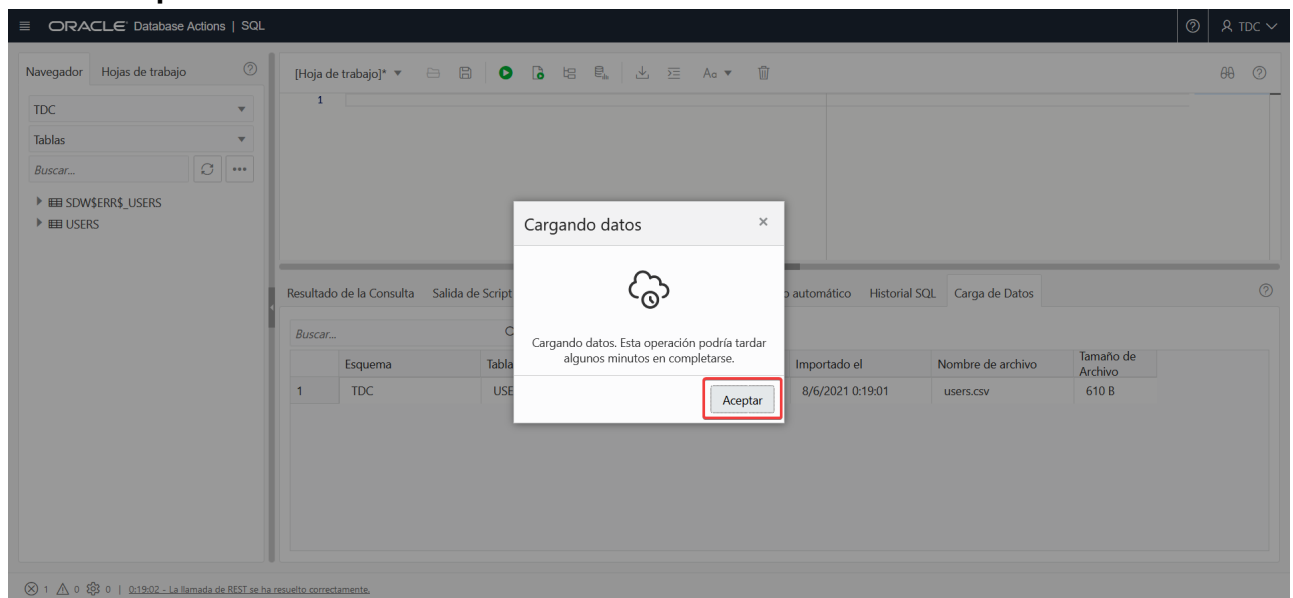
12. Cambia el **ID** a tipo **NUMBER**, selecciona la opción **PK** y deselecciona la opción **NULL**, luego haz click en **Siguiente**



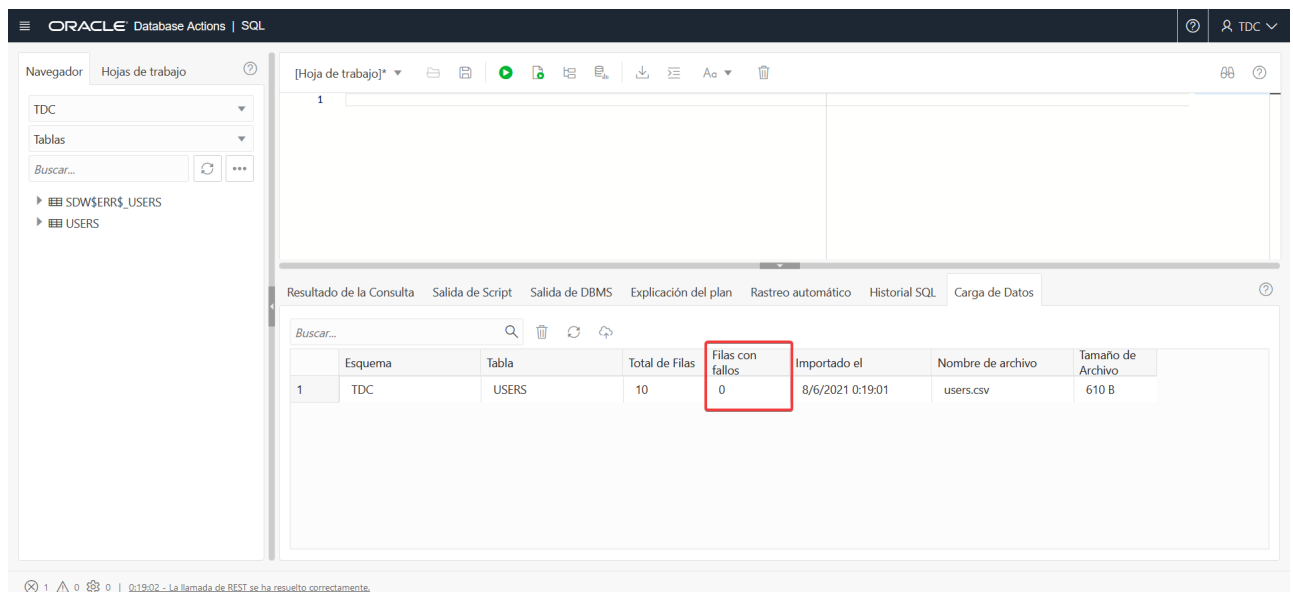
13. Se mostrará una vista previa de los comandos a ejecutarse. Haz click en **Finalizar**



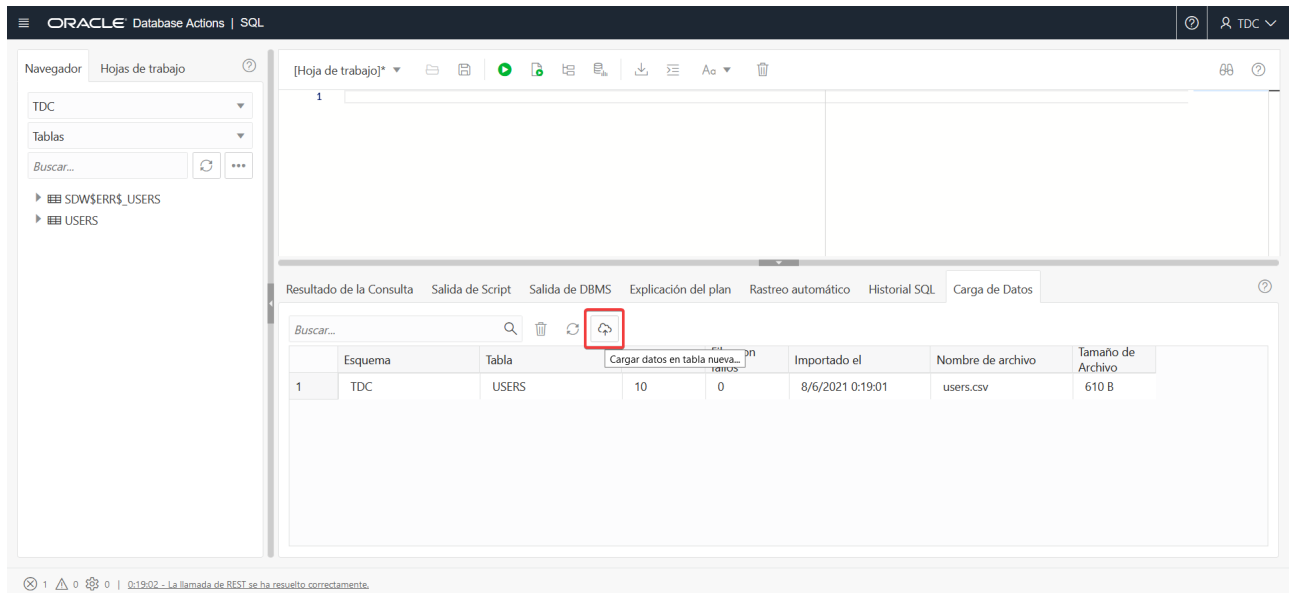
14. Aparecerá un diálogo indicándonos que la carga de datos continuará ejecutándose en el fondo. Haz click en **Aceptar**



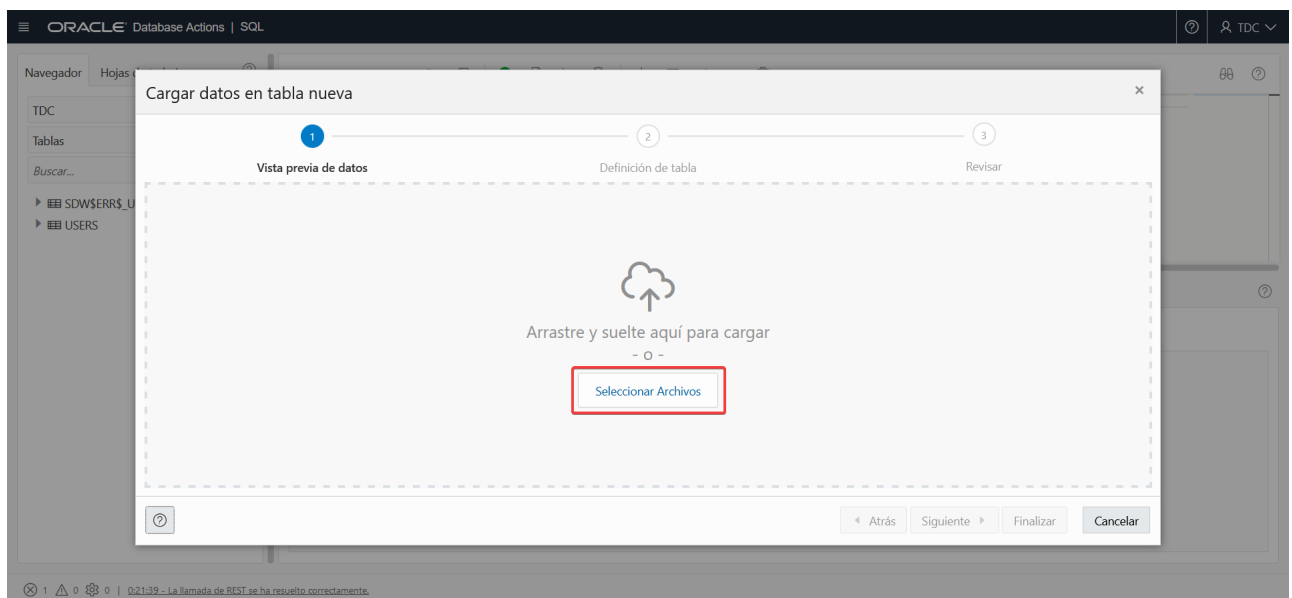
15. En el panel de abajo, debe aparecernos nuestra carga reciente en el reporte, indicando que no hubo fallos



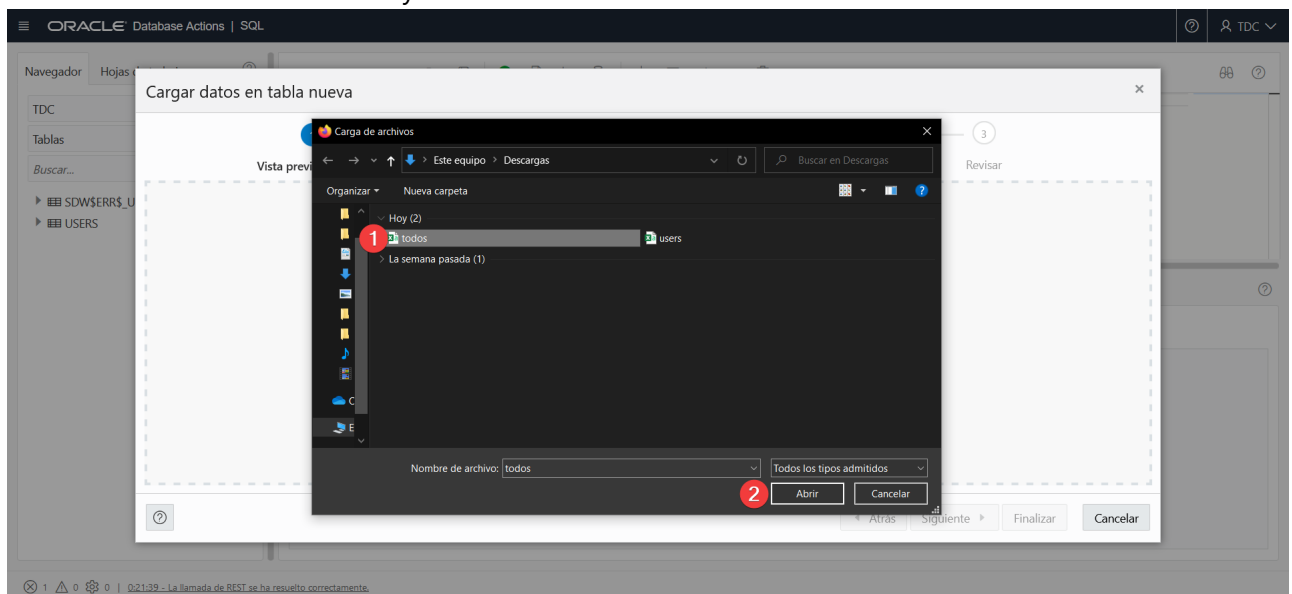
16. Haz click nuevamente en el botón con el ícono de nube



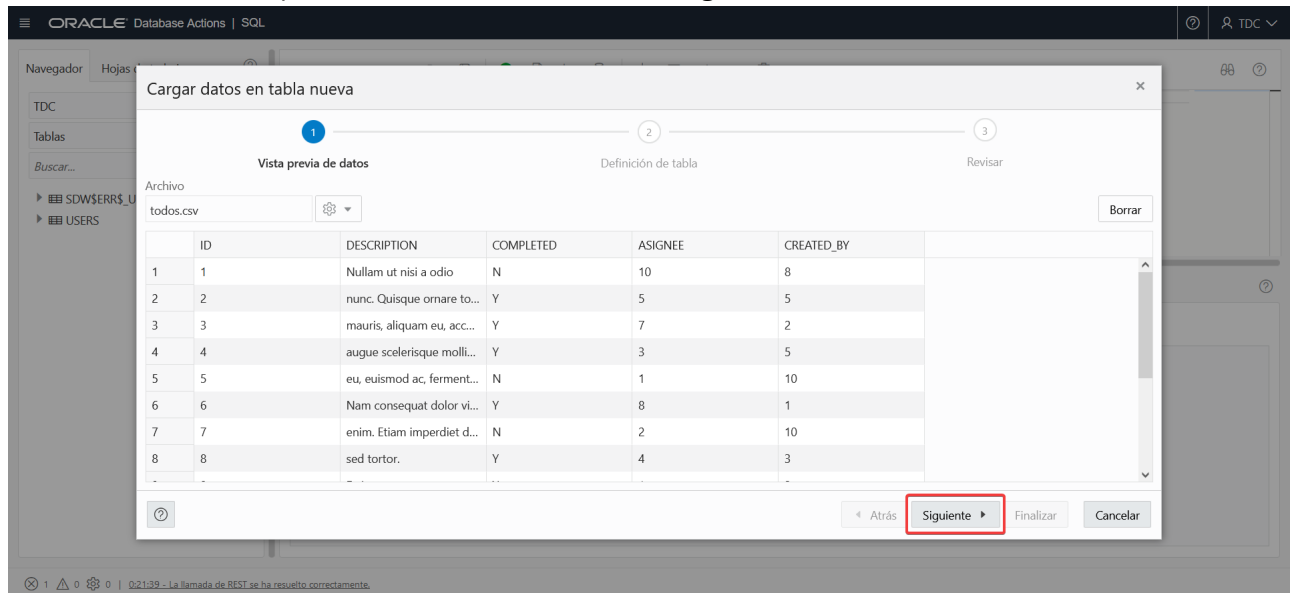
17. Haz click en **Seleccionar Archivos**



18. Selecciona el archivo **todos.csv** y haz click en **Abrir**

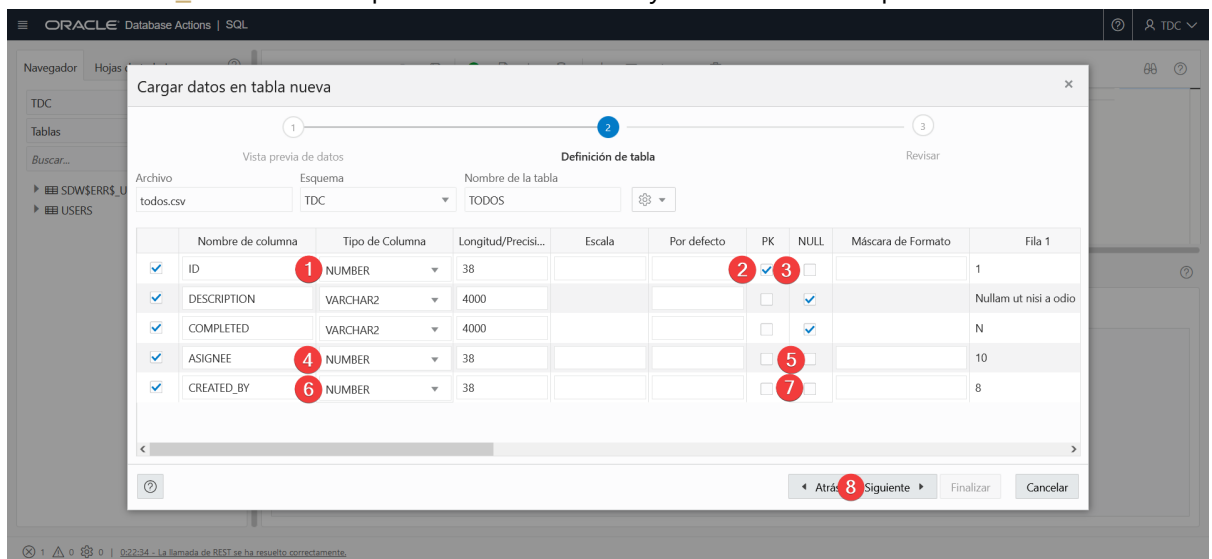


19. Se mostrará una vista previa de los datos, haz click en **Siguiente**

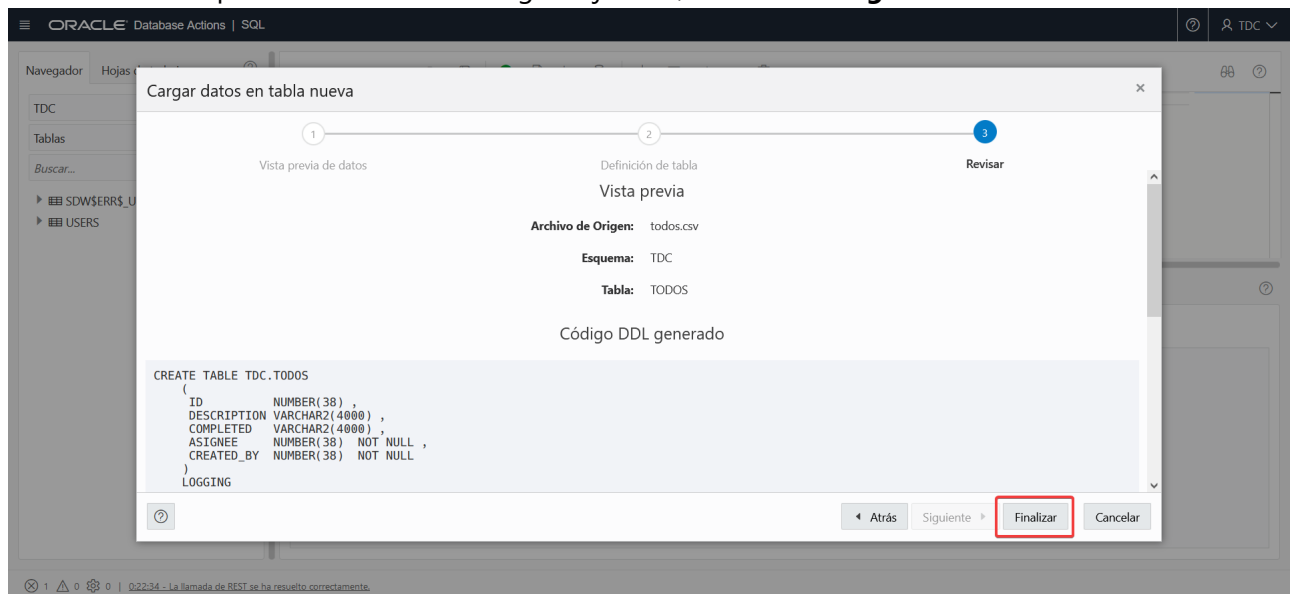


20. Cambia los setings como se indica debajo y luego haz click en **Siguiente**

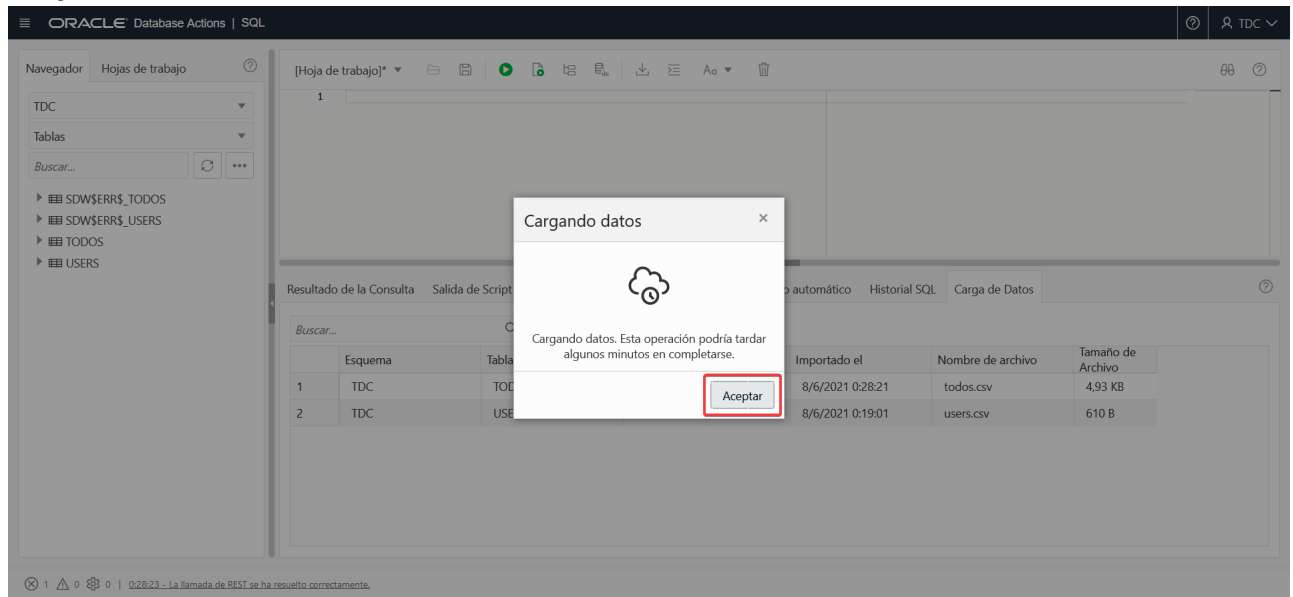
- Para **ID**, cambia el tipo de datos a **NUMBER**, selecciona la opción **PK** y deselecciona la opción **NULL**
- Para **ASIGNEE** cambia el tipo de datos a **NUMBER** y deselecciona la opción **NULL**
- Para **CREATED_BY** cambia el tipo de datos a **NUMBER** y deselecciona la opción **NULL**



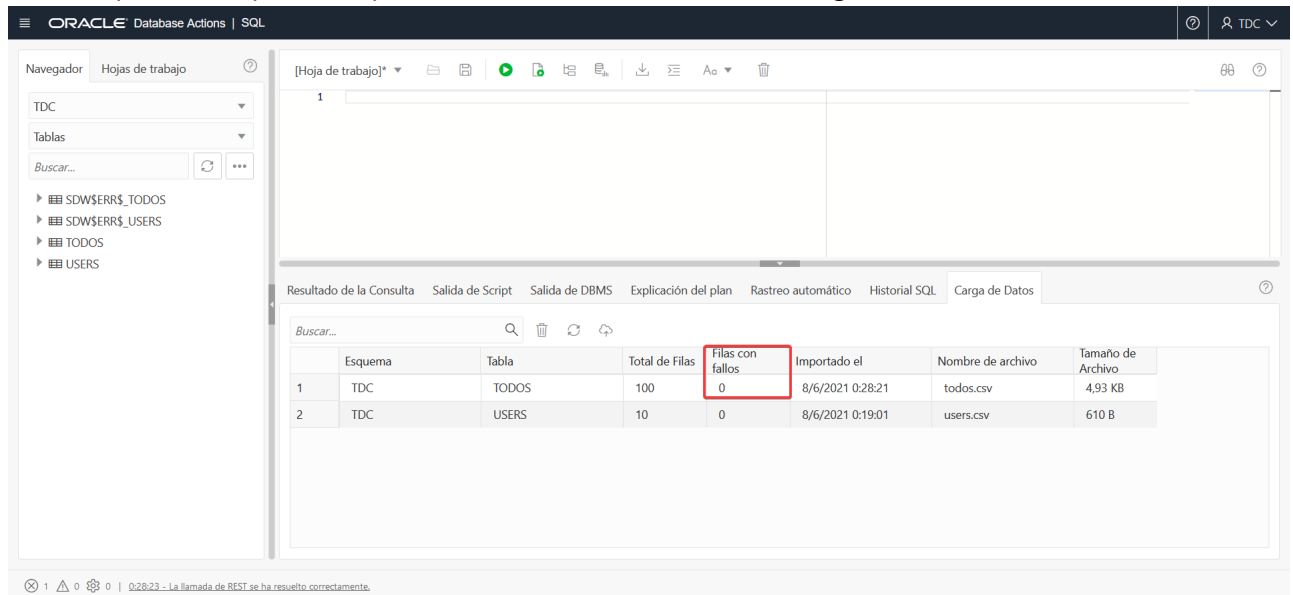
21. Se mostrará una previsualización del código a ejecutar, haz click en **Siguiente**



22. Se mostrará un diálogo indicando que la subida de datos continuará en el background. Haz click en **Aceptar**

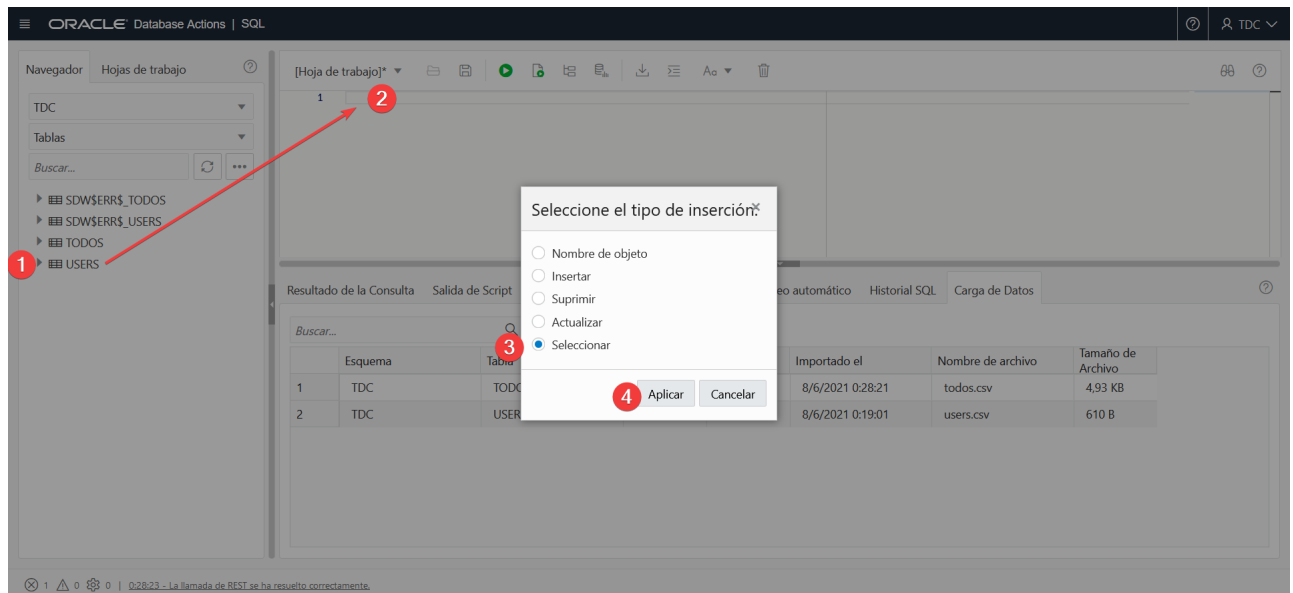


23. Verifica que en el reporte del panel inferior se muestre la nueva carga de datos con cero errores

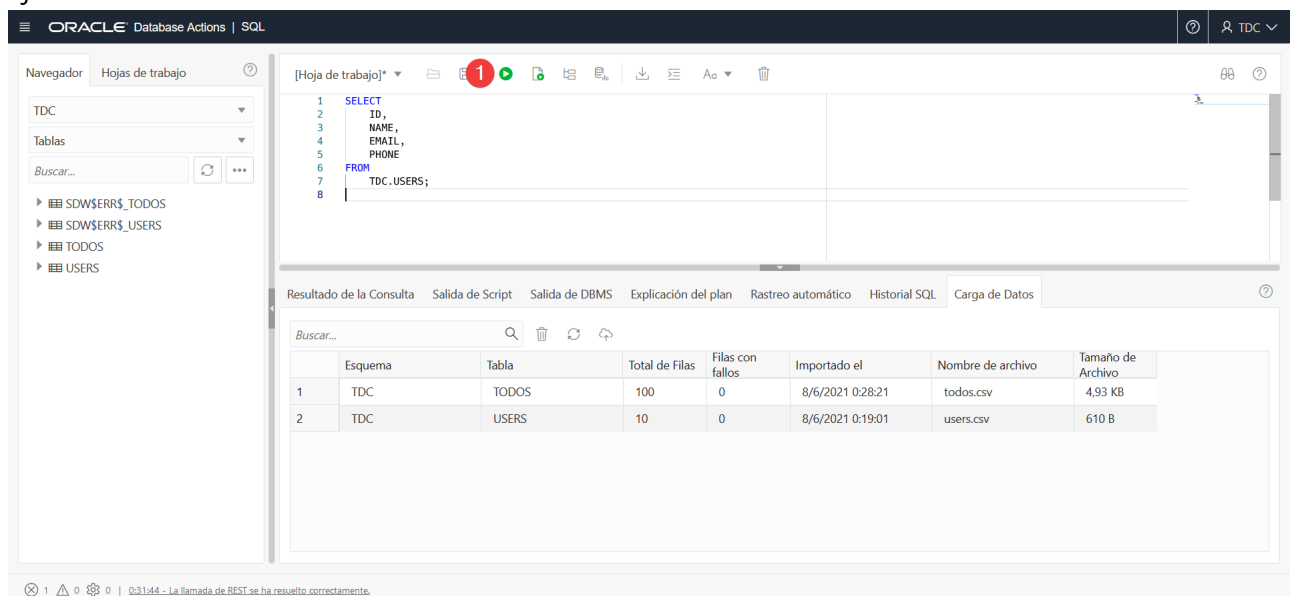


Verificación de los datos (Paso a paso)

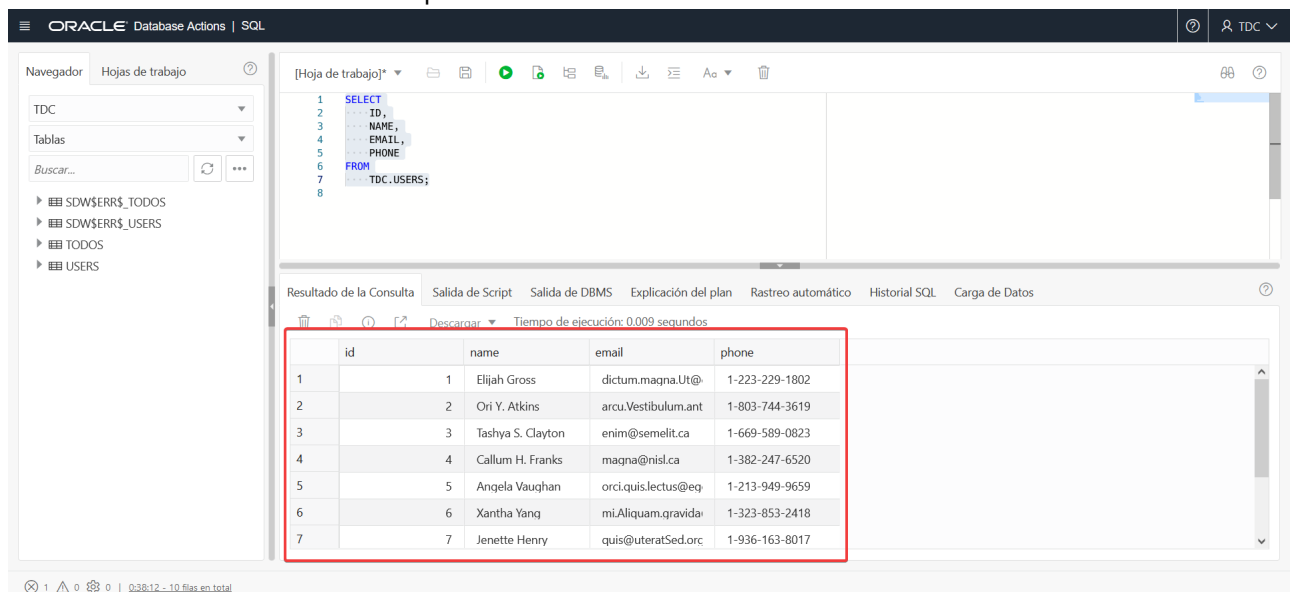
1. Arrastra la tabla **USERS** desde el panel izquierdo hacia el editor, aparecerá un diálogo. Selecciona la opción **Seleccionar** y luego haz click en **Aplicar**



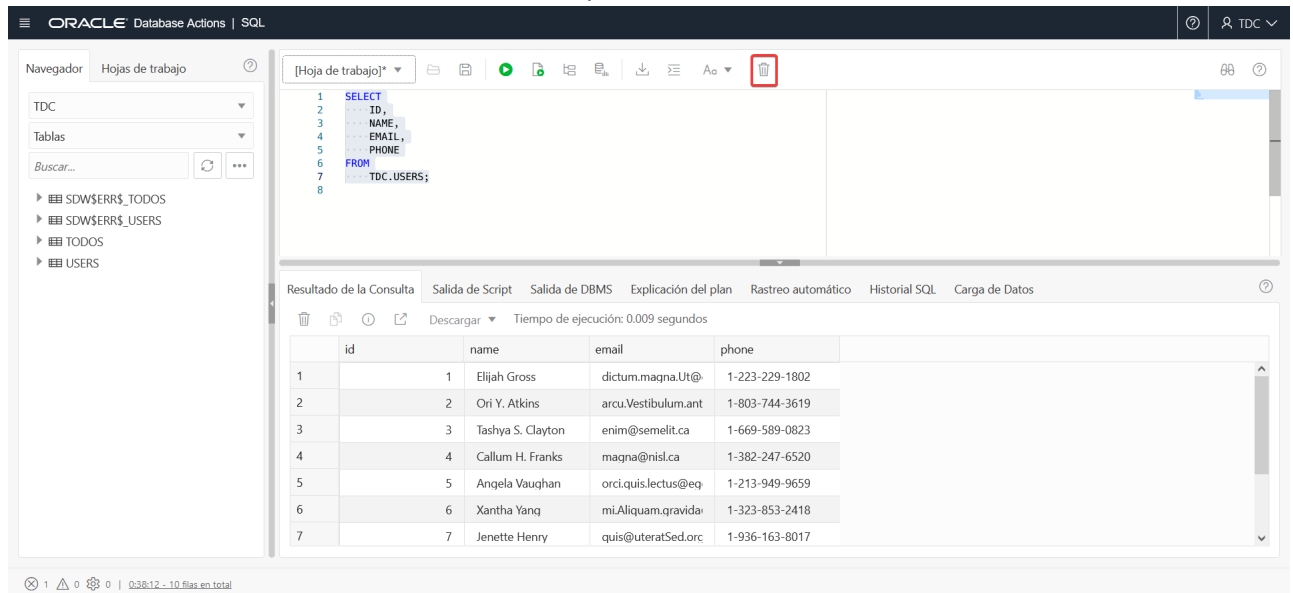
2. Se creará una consulta y se pondrá en el editor, haz click en botón con el icono de reproducción para ejecutar la consulta



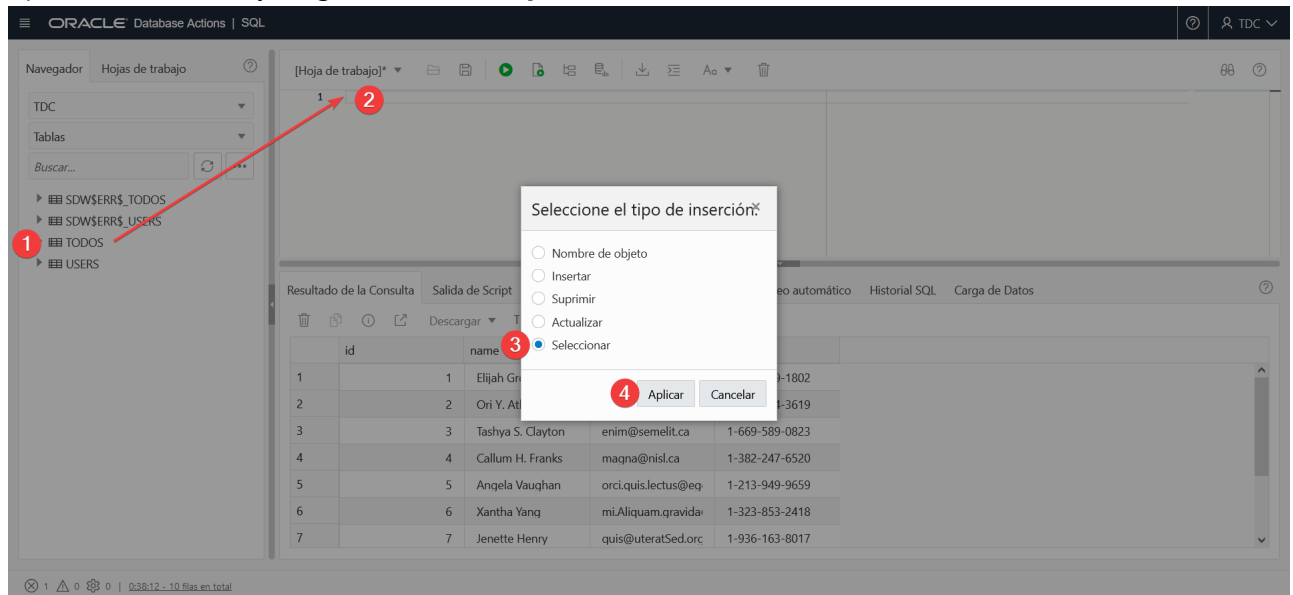
3. Se mostrarán los resultados en el panel inferior



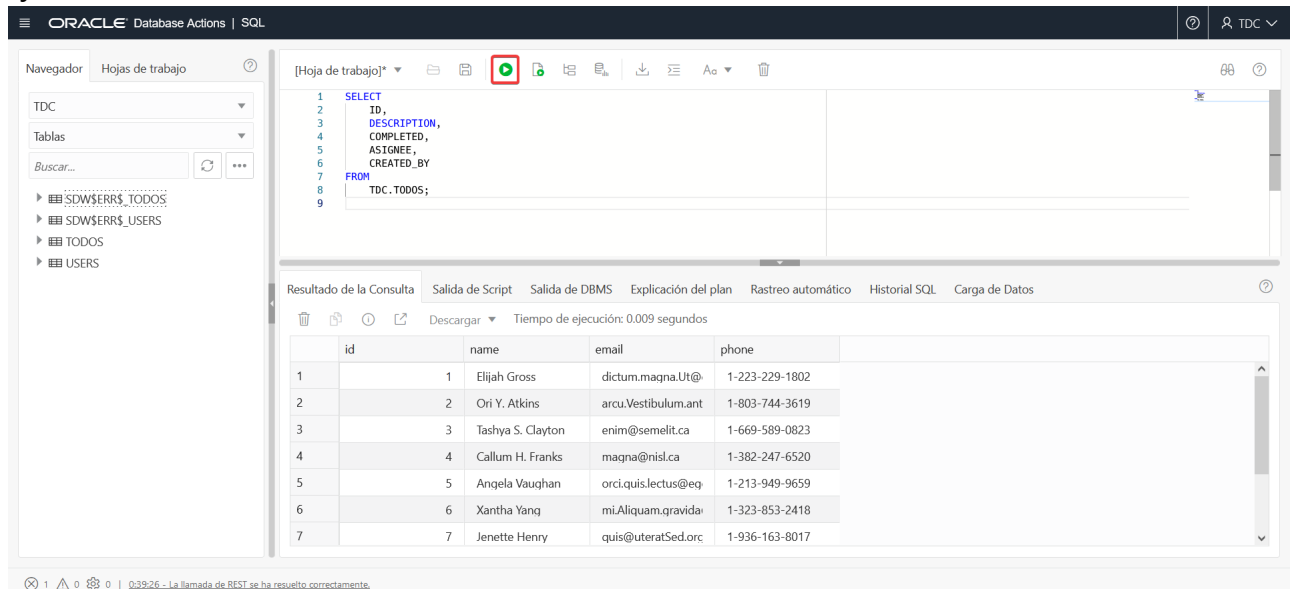
4. Haz click en el botón con el icono de basurero para borrar el contenido del editor



5. Arrastra la tabla **TODOS** desde el panel izquierdo hacia el editor, aparecerá un diálogo. Selecciona la opción **Seleccionar** y luego haz click en **Aplicar**



6. Se creará una consulta y se pondrá en el editor, haz click en botón con el icono de reproducción para ejecutar la consulta



7. Se mostrarán los resultados en el panel inferior

The screenshot shows the Oracle Database Actions SQL interface. The left sidebar contains a 'Navegador' (Navigator) panel with a tree view showing the database structure: TDC, Tablas, SDW\$ERR\$_TODOS, SDW\$ERR\$_USERS, TODOS, and USERS. The main panel displays a SQL query: `SELECT ID, DESCRIPTION, COMPLETED, ASSIGNEE, CREATED_BY FROM TDC.TODOS;`. Below the query, the 'Resultado de la Consulta' (Query Result) panel shows a table with 7 rows and 5 columns: id, description, completed, assignee, and created_by. The table is highlighted with a red border. The execution time is 0.027 segundos.

id	description	completed	assignee	created_by
1	Nullam ut nisi a odi	N	10	8
2	nunc. Quisque orna	Y	5	5
3	mauris, aliquam eu,	Y	7	2
4	auque scelerisque n	Y	3	5
5	eu, euismod ac, ferr	N	1	10
6	Nam consequat dol	Y	8	1
7	enim. Etiam imperd	N	2	10

Creación de los "endpoints" REST (Paso a paso)

1. Haz click derecho en la tabla de **USERS**, luego selecciona la opción **REST** y luego la opción **Activar**

The screenshot shows the Oracle Database Actions SQL interface. The left sidebar contains a 'Navegador' (Navigator) panel with a tree view showing the database structure: TDC, Tablas, SDW\$ERR\$_TODOS, SDW\$ERR\$_USERS, TODOS, and USERS. The main panel displays a SQL query: `SELECT ID, DESCRIPTION, COMPLETED, ASSIGNEE, CREATED_BY FROM TDC.TODOS;`. Below the query, the 'Resultado de la Consulta' (Query Result) panel shows a table with 7 rows and 5 columns: id, description, completed, assignee, and created_by. The table is highlighted with a red border. The execution time is 0.024 segundos. A context menu is open over the 'USERS' table in the sidebar, showing options: 'Abrir', 'Editar...', 'Agregar', 'Usar como Plantilla', 'REST', 'Carga de Datos', and 'DDL Rápido'. The 'REST' option is highlighted with a red circle, and the 'Activar...' option is highlighted with a red circle.

id	description	completed	assignee	created_by
1	Nullam ut nisi a odi	N	10	8
2	nunc. Quisque orna	Y	5	5
3	mauris, aliquam eu,	Y	7	2
4	auque scelerisque n	Y	3	5
5	eu, euismod ac, ferr	N	1	10
6	Nam consequat dol	Y	8	1
7	enim. Etiam imperd	N	2	10

2. Se mostrará un panel para activar la tabla como "endpoint" REST, haz click en **Activar**.

Esto creará un "endpoint" REST con soporte para los métodos **GET**, **POST**, **PUT** y **DELETE** de forma automática, los cuales probaremos más adelante.

Nota que la opción **Necesita Autenticación** está desactivada para simplificar este ejemplo, pero dependiendo de tus necesidades necesitarás activarla. Los siguientes pasos asumen que esta opción

está desactivada

Activar objeto para REST

Nombre de objeto: USERS
Tipo de objeto: TABLE

Alias de objeto: users

URL de vista previa: https://.../ords/tdc/users/

Rol de autorización: oracle.dbtools.role.autorest.TDC.USERS
Privilegio de autorización: oracle.dbtools.autorest.privilege.TDC.USERS

Necesita autenticación: ☐

Activar Cancelar

3. Una vez activada la tabla como "endpoint" REST se mostrará un icono de clavija en la tabla, indicando que la tabla esta activada.

Confirmación

Objeto REST activado
La tabla USERS se ha activado correctamente para REST

4. Haz click derecho en la tabla **TODOS** en el panel izquierdo, selecciona la opción **REST** y posteriormente selecciona la opción **Activar**

The screenshot shows the Oracle Database Actions interface. On the left, the 'Navegador' (Navigator) pane displays a tree structure with 'TDC' expanded, showing 'SDW\$ERR\$_TODOS', 'SDW\$ERR\$_USERS', 'TODOS', and 'USERS'. A context menu is open over the 'TODOS' table, with options like 'Abrir', 'Editar...', 'Agregar', 'Usar como Plantilla', 'REST', 'Carga de Datos', and 'DDL Rápido'. The 'REST' option is highlighted, and a sub-menu is visible with the 'Activar...' option. The main pane shows a SQL query: `SELECT ID, DESCRIPTION, COMPLETED, ASSIGNEE, CREATED_BY FROM TDC.TODOS;`. Below the query, the 'Resultado de la Consulta' (Query Result) table is displayed, showing 7 rows of data.

id	description	completed	assignee	created_by
1	Nullam ut nisi a odi	N	10	8
2	nunc. Quisque orna	Y	5	5
3	mauris, aliquam eu,	Y	7	2
4	auque scelerisque n	Y	3	5
5	eu, euismod ac, ferr	N	1	10
6	Nam consequat dol	Y	8	1
7	enim. Etiam imperd	N	2	10

5. Se mostrará un panel, haz click en **Activar**.

The screenshot shows the 'Activar objeto para REST' (Activate object for REST) dialog box. The dialog contains the following fields and values:

- Nombre de objeto: TODOS
- Tipo de objeto: TABLE
- Alias de objeto: todos
- URL de vista previa: https://.../ords/tdc/todos/
- Rol de autorización: oracle.dbtools.role.autorest.TDC.TODOS
- Privilegio de autorización: oracle.dbtools.autorest.privilege.TDC.TODOS
- Necesita autenticación: ☐ (unchecked)

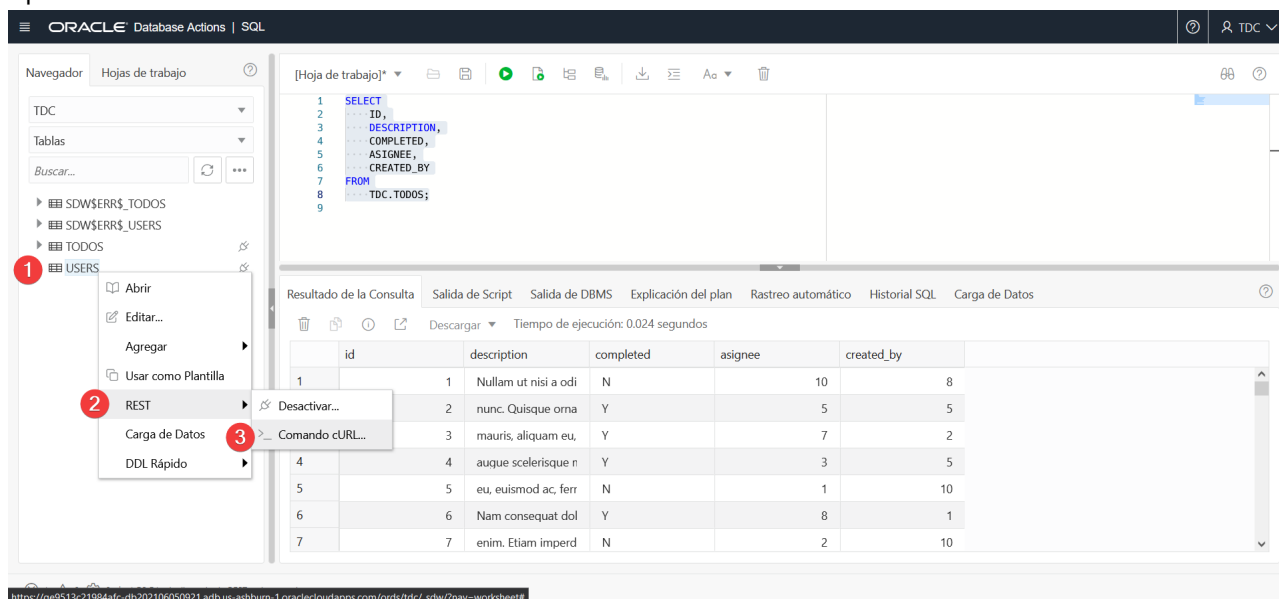
At the bottom right of the dialog, there are 'Activar' and 'Cancelar' buttons. The 'Activar' button is highlighted with a red box.

6. Una vez activada la tabla como "endpoint" REST se mostrará un icono de clavija en la tabla, indicando que la tabla esta activada.

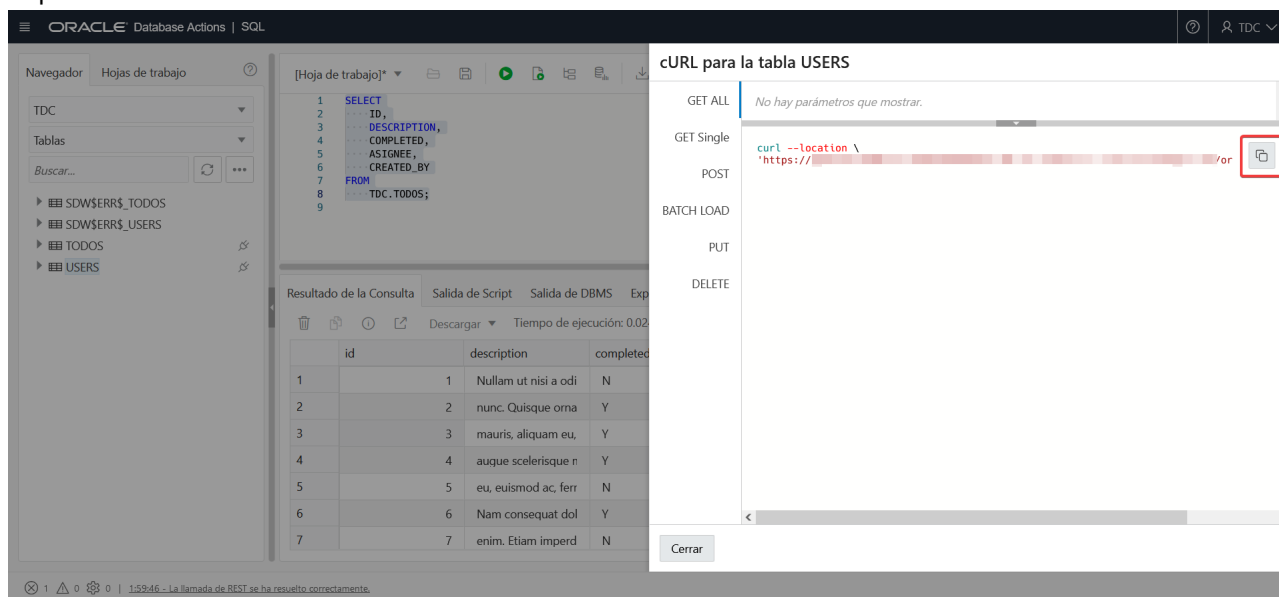
The screenshot shows the Oracle Database Actions interface after the 'TODOS' table has been activated as a REST endpoint. A confirmation dialog box is visible in the top right corner, stating: 'Confirmación: Objeto REST activado. La tabla TODOS se ha activado correctamente para REST'. In the 'Navegador' pane, the 'TODOS' table now has a small plug icon next to its name, indicating it is activated. The main pane shows the same SQL query and query result table as in the previous screenshot.

Probando los "endpoints" REST con cURL

1. Haz click derecho en la tabla **USERS** del panel izquierdo, selecciona la opción **REST** y posteriormente la opción **Comando cURL...**



2. Se abrirá un panel del lado derecho que nos mostrará distintas tabs en vertical con las acciones que podemos realizar. Por defecto, la tab **GET All** esta seleccionada. Haz click en el botón con el icono de copiar



3. Abre una terminal, en Windows puedes usar **Git Bash**, luego pega el comando que copiaste en la terminal, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el JSON contiene hasta 25 filas de la tabla y además contiene los links en la parte de abajo para paginar el recurso

```
mingw64 C:\Users\mig84
$ curl --location 'https://ords/tdc/users/' | jq
% Total % Received % Xferd Average Speed Time Time Time Current
 000 3150 0 3150 0 0 3942 0 --:--:-- --:--:-- --:--:-- 3942
{
  "items": [
    {
      "id": 11,
      "name": "Miguel",
      "email": "example@example.com",
      "phone": "1234567890",
      "links": [
        {
          "rel": "self",
          "href": "https://ords/tdc/users/11"
        }
      ]
    },
    {
      "id": 12,
      "name": null,
      "email": null,
      "phone": null,
      "links": [
        {
          "rel": "self",
          "href": "https://ords/tdc/users/12"
        }
      ]
    },
    {
      "id": 1,
      "name": "Elijah Gross",
      "email": "dictum.magna.uta@et.co.uk",
      "phone": "1-222-222-1802",
      "links": [
        {
          "rel": "self",
          "href": "https://ords/tdc/users/1"
        }
      ]
    },
    {
      "id": 2,
      "name": "Ori Y. Atkins",
      "email": "arcu.Vestibulum.ante@Morbineque.co.uk",
      "phone": "1-803-744-3619",
      "links": [

```

4. Regresa a Database Actions, selecciona la tab **GET Simple** y rellena el campo de **id** con **1**, luego haz click en el botón con el icono de copiar

Oracle Database Actions | SQL

Navegador: TDC

Tablas: SDW\$ERR\$_TODOS, SDW\$ERR\$_USERS, TODOS, USERS

[Hoja de trabajo] SELECT ID, DESCRIPTION, COMPLETED, ASSIGNED, CREATED_BY FROM TDC.TODOS;

Resultado de la Consulta

id	description	completed
1	Nullam ut nisi a odi	N
2	nunc. Quisque orna	Y
3	mauris, aliquam eu,	Y
4	augue scelerisque n	Y
5	eu, euismod ac, ferr	N
6	Nam consequat dol	Y
7	enim, Etiam imperd	N

cURL para la tabla USERS

GET ALL: Rellene las entradas para sustituir los valores del comando

GET Single: Nombre de sustitución: id, Valor: 1

POST:

BATCH LOAD:

PUT:

DELETE: curl --location \ 'https://ords/tdc/users/1' -XDELETE

Cerrar

5. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el JSON contiene una sola fila, y que el final de la URL termina con un valor de la columna **ID** de la tabla, esto debido a que en la tabla, indicamos que nuestro **PK** (Primary Key) era esa misma

columna

```
MINGW64/c/Users/mig84
$ curl --location 'https://.../ords/tdc/users/1' | jq
{"id": 1,
 "name": "Elijah Gross",
 "email": "dictum_magna.ut@et.co.uk",
 "phone": "1-223-229-1802",
 "links": [
  {
    "rel": "self",
    "href": "https://.../ords/tdc/users/1"
  },
  {
    "rel": "edit",
    "href": "https://.../ords/tdc/users/1"
  },
  {
    "rel": "describedby",
    "href": "https://.../ords/tdc/metadata-catalog/users/item"
  },
  {
    "rel": "collection",
    "href": "https://.../ords/tdc/users/"
  }
 ]
}
```

6. Regresa a Database Actions, selecciona la tab **POST**, rellena el campo de **id** con **20** (Ya que no existe en nuestro data set), después rellena los demás campos y haz click en el botón con el icono de copiar

cURL para la tabla USERS

GET ALL: Rellene las entradas para sustituir los valores del comando

GET Single

Nombre de sustitución	Valor
1 id	20
2 name	Miguel
3 email	example@example.com
4 phone	1234567890

POST

BATCH LOAD

PUT

DELETE

5 `curl --location --request POST \`
`'https://.../ords/tdc/users/1'`
`--header 'Content-Type: application/json' \`
`--data-binary '{`
 `"id": "20",`
 `"name": "Miguel",`
 `"email": "example@example.com",`
 `"phone": "1234567890"`
`}`
`'`

Cerrar

7. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método POST insertará una fila y que el JSON devuelto contiene la fila recién creada

```
MINGW64 ~/c/Users/mig84
$ curl --location --request POST \
  'https://.../ords/tdc/users/' \
  --header 'Content-type: application/json' \
  --data-binary '{
    "id": "20",
    "name": "Miguel",
    "email": "example@example.com",
    "phone": "1234567890"
  }' | jq
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100    683    0  588  100    95    386    62  0:00:01  0:00:01 --:--:--  448
{
  "id": "20",
  "name": "Miguel",
  "email": "example@example.com",
  "phone": "1234567890",
  "links": [
    {
      "rel": "self",
      "href": "https://.../ords/tdc/users/20"
    },
    {
      "rel": "edit",
      "href": "https://.../ords/tdc/users/20"
    },
    {
      "rel": "describentiv",
      "href": "https://.../ords/tdc/metadata-catalog/users/item"
    },
    {
      "rel": "collection",
      "href": "https://.../ords/tdc/users/"
    }
  ]
}
```

8. Regresa a Database Actions, selecciona la tab **PUT** y cambia el campo de **email** con un valor distinto al actual, luego haz click en el botón con el icono de copiar

Nombre de sustitución	Valor
1 id	20
2 name	Miguel
3 email	example@example.com
4 phone	1234567890

```
curl --location --request PUT \
  'https://.../ords/tdc/users/' \
  --header 'Content-Type: application/json' \
  --data-binary '{
    "name": "Miguel",
    "email": "example@example.com",
    "phone": "1234567890"
  }'
```

9. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

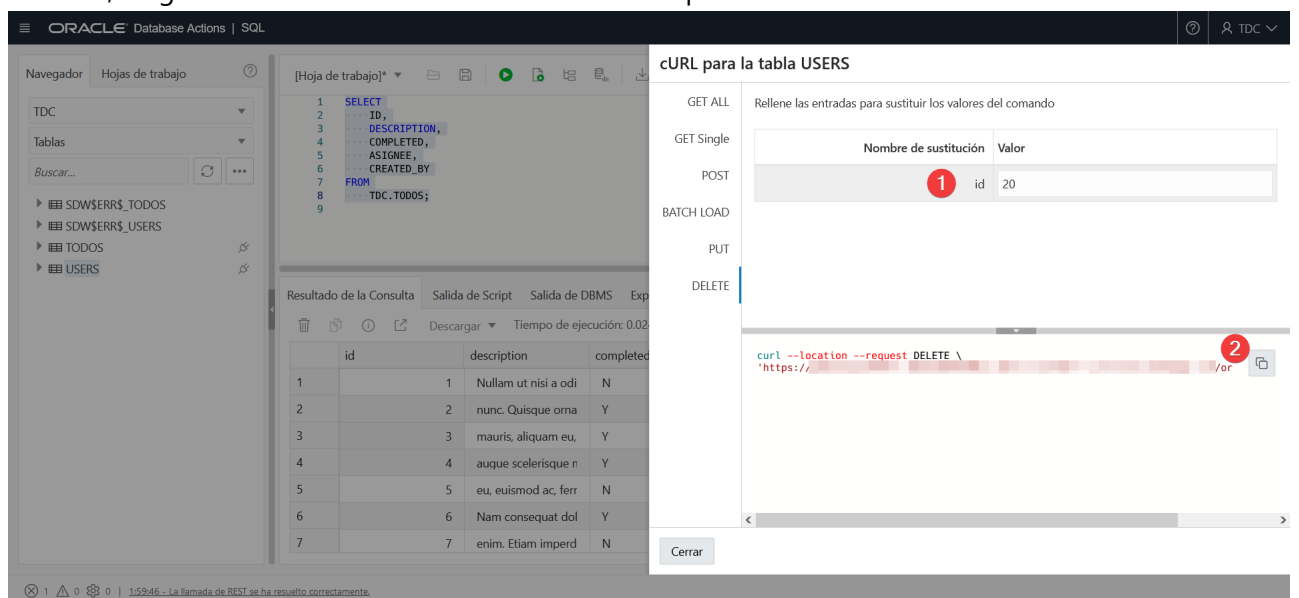
para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método PUT actualizará una fila y que el JSON devuelto contiene la fila recién actualizada

```
mingw64 C:\Users\mig84
$ curl --location --request PUT \
  https://[redacted] /ords/tdc/users/20' \
  --header 'Content-Type: application/json' \
  --data-binary '{
    "name": "Miguel",
    "email": "example2@example.com",
    "phone": "1234567890",
    "links": [
      {
        "rel": "self",
        "href": "https://[redacted] /ords/tdc/users/20"
      },
      {
        "rel": "edit",
        "href": "https://[redacted] /ords/tdc/users/20"
      },
      {
        "rel": "describedby",
        "href": "https://[redacted] /ords/tdc/metadata-catalog/users/item"
      },
      {
        "rel": "collection",
        "href": "https://[redacted] /ords/tdc/users/"
      }
    ]
  }'
% Total % Received % Xferd Average Speed Time Time Time Current
100 671 0 589 100 82 405 56 0:00:01 0:00:01 --:--:-- 461

{"id": 20,
 "name": "Miguel",
 "email": "example2@example.com",
 "phone": "1234567890",
 "links": [
  {
    "rel": "self",
    "href": "https://[redacted] /ords/tdc/users/20"
  },
  {
    "rel": "edit",
    "href": "https://[redacted] /ords/tdc/users/20"
  },
  {
    "rel": "describedby",
    "href": "https://[redacted] /ords/tdc/metadata-catalog/users/item"
  },
  {
    "rel": "collection",
    "href": "https://[redacted] /ords/tdc/users/"
  }
 ]
}
```

10. Regresa a Database Actions, selecciona la tab **DELETE** y asegurate que el campo de **id** contenga el valor **20**, luego haz click en el botón con el icono de copiar



11. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método DELETE eliminará una fila y que el JSON devuelto contiene el número de filas eliminadas

```
MINGW64 ~/c/Users/mig84
$ curl --location --request DELETE \
  https://ords/tdc/users/20' | jq
% Total    % Received % Xferd  Average Speed   Time    Time     Time    Current
           Dload  Upload   Dload  Upload   Total   Spent    Left     Speed
100    17    0    17    0    0      8      0      0      0      0      0      0      0      0      0      0      0      0      0
{"rowsDeleted": 1}
```

12. Regresa a Database Actions. Haz click en el boton **Cerrar** del panel que aparece a la izquierda y luego haz click derecho en la tabla **TODOS** del panel izquierdo, selecciona la opcion **REST** y posteriormente la opción **Comando cURL...**

The screenshot shows the Oracle Database Actions interface. On the left, the 'Navegador' (Navigator) panel displays a tree structure with 'TDC' expanded, showing 'Tablas' (Tables) and 'USERS'. The 'TODOS' table is selected. A right-click context menu is open over the 'TODOS' table. The menu options are: 'Abrir' (Open), 'Editar...' (Edit...), 'Agregar' (Add), 'Usar como Plantilla' (Use as Template), 'REST' (highlighted with a red circle 2), 'Carga de Datos' (Data Load), and 'DDL Rápido' (Quick DDL). The 'REST' option has a sub-menu open, showing 'Desactivar...' (Deactivate...), 'Comando cURL...' (highlighted with a red circle 3), and 'DDL Rápido'.

id	description	completed	assignee	created_by
1	Nullam ut nisi a odi	N	10	8
2	nunc. Quisque orna	Y	5	5
3	mauris, aliquam eu,	Y	7	2
4	auque scelerisque n	Y	3	5
5	eu, euismod ac, ferr	N	1	10
6	Nam consequat dol	Y	8	1
7	enim. Etiam imperd	N	2	10

13. El panel de cURL se abrirá en la tab **GET All**, haz click en el botón con el icono de copiar

The screenshot shows the Oracle Database Actions interface with the 'cURL para la tabla TODOS' panel open. The panel has a tab 'GET ALL' selected, showing 'No hay parámetros que mostrar.' (No parameters to show). Below this, there are radio buttons for 'GET Single', 'POST', 'BATCH LOAD', 'PUT', and 'DELETE'. The 'GET Single' radio button is selected. The 'curl --location \'https://\'/or' text is visible, and a red box highlights the copy icon (two overlapping sheets of paper) next to it. The 'Cerrar' (Close) button is at the bottom.

14. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

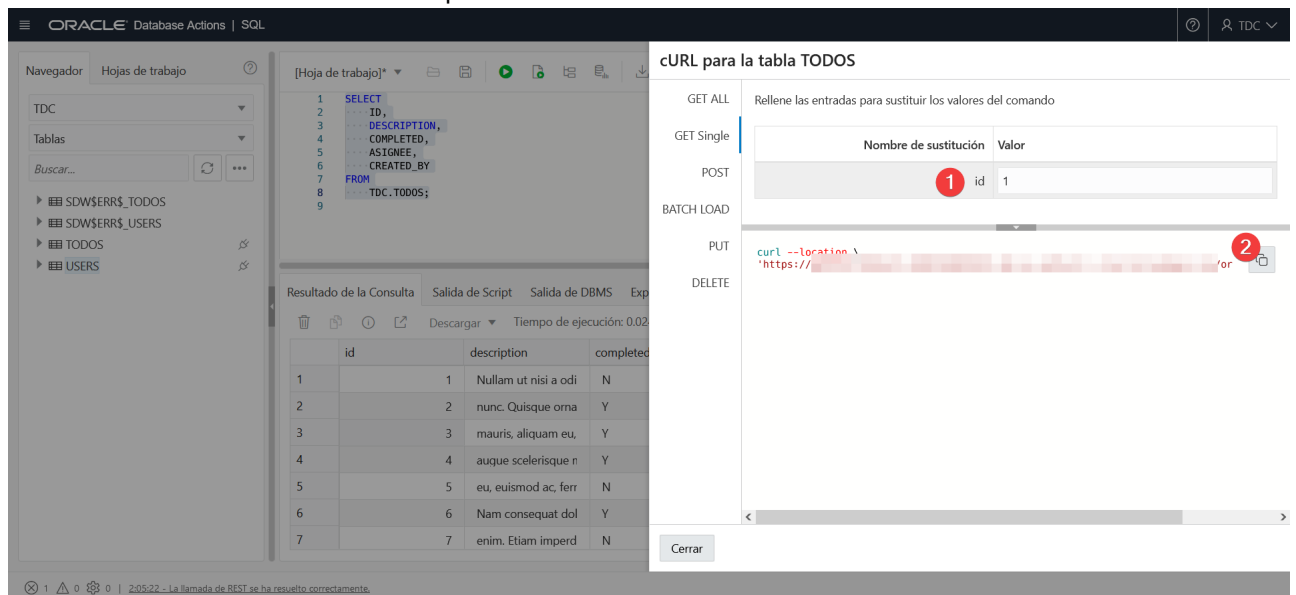
para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el JSON contiene hasta 25 filas de la tabla y además contiene los links en la parte de abajo para paginar el recurso



```
$ curl --location https://.../ords/tdc/todos/ | jq
{"items": [{"id": 1, "description": "Nullam ut nisi a odio", "completed": "N", "assignee": 10, "created_by": 8, "links": [{"rel": "self", "href": "https://.../ords/tdc/todos/1"}]}, {"id": 2, "description": "nunc. Quisque ornare tortor", "completed": "Y", "assignee": 5, "created_by": 5, "links": [{"rel": "self", "href": "https://.../ords/tdc/todos/2"}]}, {"id": 3, "description": "mauris, aliquam eu, accumsan sed,", "completed": "Y", "assignee": 7, "created_by": 2, "links": [{"rel": "self", "href": "https://.../ords/tdc/todos/3"}]}, {"id": 4,
```

15. Regresa a Database Actions, selecciona la tab **GET Simple** y rellena el campo de **id** con **1**, luego haz click en el botón con el icono de copiar



cURL para la tabla TODOS

GET ALL: Rellene las entradas para sustituir los valores del comando

GET Single: Nombre de sustitución: id, Valor: 1

POST: 1

BATCH LOAD: 2

PUT: curl --location https://.../ords/tdc/todos/1

DELETE: 2

Cerrar

Resultado de la Consulta

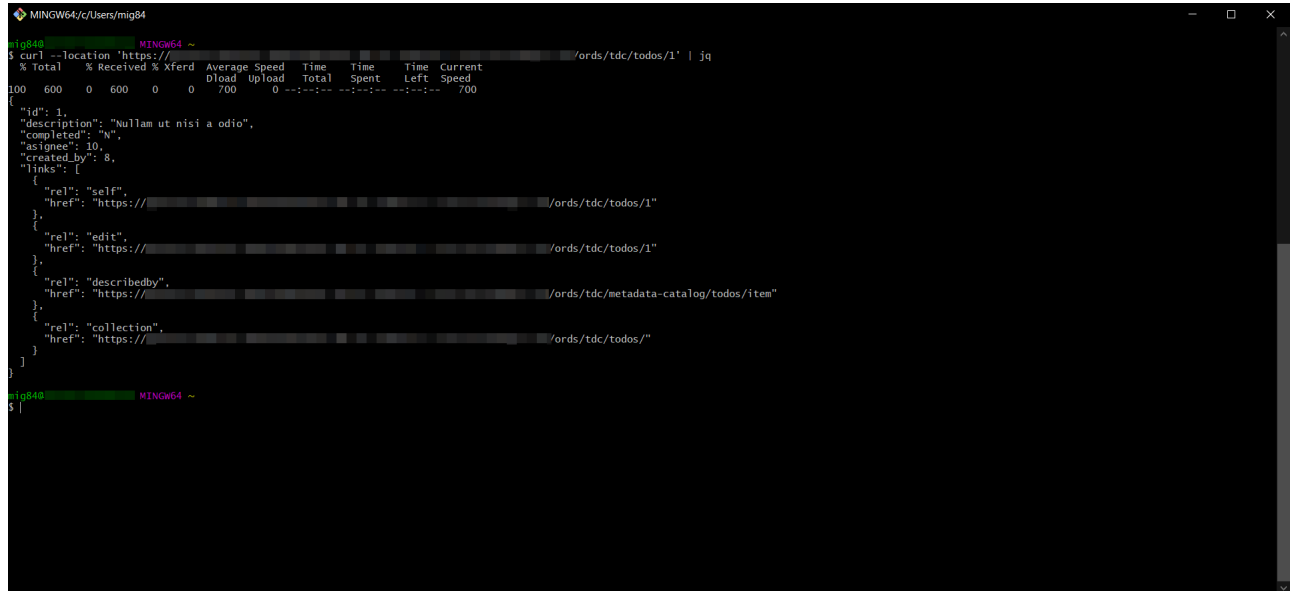
id	description	completed
1	Nullam ut nisi a odi	N
2	nunc. Quisque orna	Y
3	mauris, aliquam eu,	Y
4	auque scelerisque n	Y
5	eu, euismod ac, ferr	N
6	Nam consequat dol	Y
7	enim. Etiam imperd	N

16. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

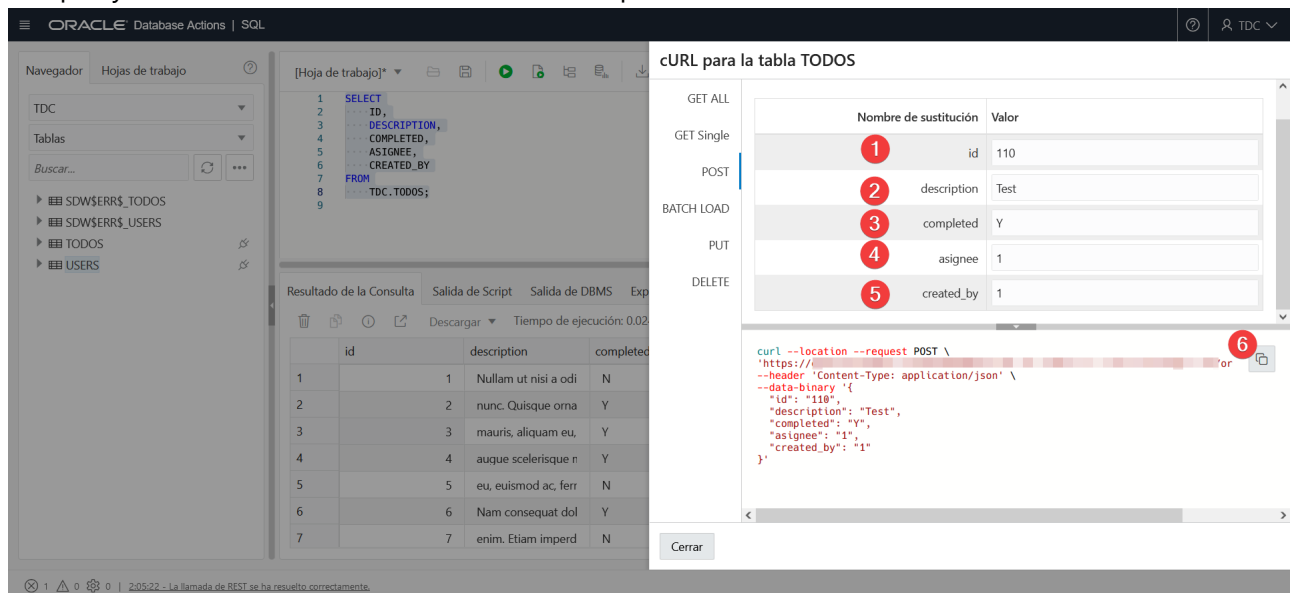
para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el JSON contiene una sola fila, y que el final de la URL termina con un valor de la columna **ID** de la tabla, esto debido a que en la tabla, indicamos que nuestro **PK** (Primary Key) era esa misma columna



```
mingw64 ~
$ curl --location 'https://ords/tdc/todos/1' | jq
{
  "id": 1,
  "description": "Nullam ut nisi a odio",
  "completed": "N",
  "assignee": 10,
  "created_by": 8,
  "links": [
    {
      "rel": "self",
      "href": "https://ords/tdc/todos/1"
    },
    {
      "rel": "edit",
      "href": "https://ords/tdc/todos/1"
    },
    {
      "rel": "describedby",
      "href": "https://ords/tdc/metadata-catalog/todos/item"
    },
    {
      "rel": "collection",
      "href": "https://ords/tdc/todos/"
    }
  ]
}
```

17. Regresa a Database Actions, selecciona la tab **POST**, rellena el campo de **id** con **110** (Ya que no existe en nuestro data set), el campo **assignee** con **1** y el campo **created_by** con **1**, después rellena los demás campos y haz click en el botón con el icono de copiar



The screenshot shows the Oracle Database Actions interface. On the left, a sidebar lists tables: TDC, SDW\$ERR\$_USERS, SDW\$ERR\$_USERS, TODOS, and USERS. The main panel displays a SQL query: `SELECT ID, DESCRIPTION, COMPLETED, ASSIGNEE, CREATED_BY FROM TDC.TODOS;`. Below the query, a table shows the results of the query. On the right, a panel titled "cURL para la tabla TODOS" shows a REST client configuration for a POST request. The configuration includes a table for field substitution and a cURL command.

Nombre de sustitución	Valor
1	id
2	description
3	completed
4	assignee
5	created_by

```
curl --location --request POST \
  'https://ords/tdc/todos/' \
  --header 'Content-Type: application/json' \
  --data-binary '{
    "id": "110",
    "description": "Test",
    "completed": "Y",
    "assignee": "1",
    "created_by": "1"
  }'
```

18. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

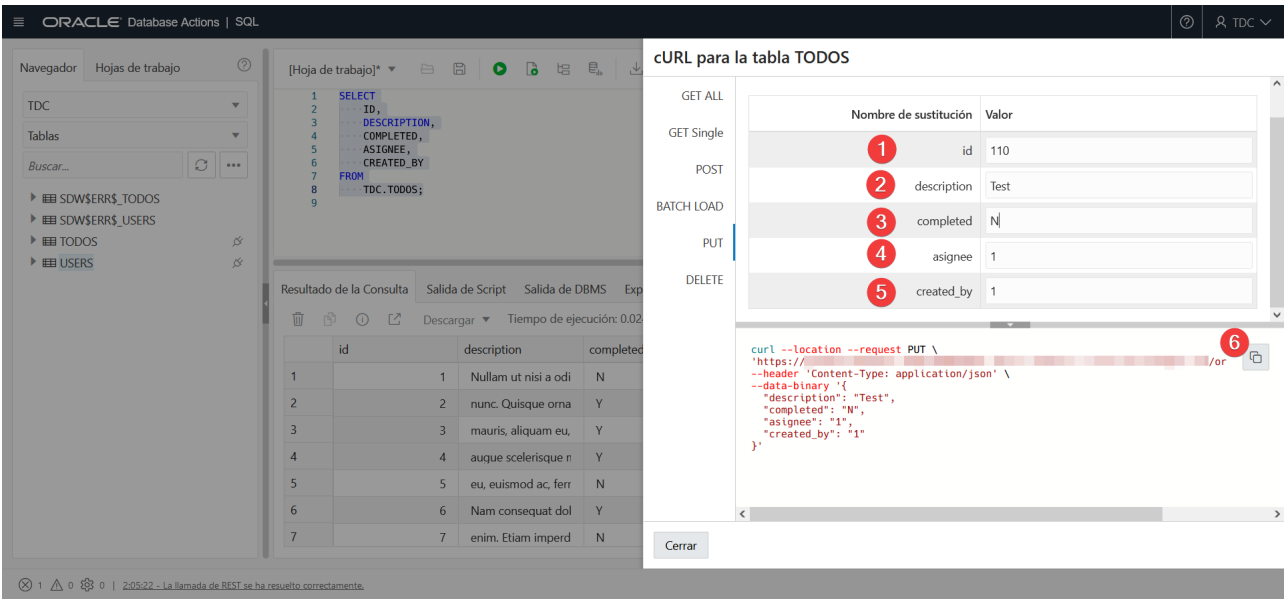
```
<COMANDO_PEGADO> | jq
```


para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método POST insertará una fila y que el JSON devuelto contiene la fila recién creada



19. Regresa a Database Actions, selecciona la tab **PUT** y cambia el campo de **completed** con un valor distinto al actual, luego haz click en el botón con el icono de copiar



20. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método PUT actualizará una fila y que el JSON devuelto contiene la fila recién actualizada

```
mingw64 MINGW64 ~
$ curl --location --request PUT \
  https://.../ords/tdc/todos/110 \
  --header 'Content-Type: application/json' \
  --data-binary '{
    "description": "Test",
    "completed": "N",
    "assignee": "1",
    "created_by": "1"
  }' | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 674 0 588 100 86 136 22 0:00:03 0:00:03 -- -- 179

{"id": 110,
 "description": "Test",
 "completed": "N",
 "assignee": "1",
 "created_by": "1",
 "links": [
   {
     "rel": "self",
     "href": "https://.../ords/tdc/todos/110"
   },
   {
     "rel": "edit",
     "href": "https://.../ords/tdc/todos/110"
   },
   {
     "rel": "describedby",
     "href": "https://.../ords/tdc/metadata-catalog/todos/item"
   },
   {
     "rel": "collection",
     "href": "https://.../ords/tdc/todos/"
   }
 ]
}
```

21. Regresa a Database Actions, selecciona la tab **DELETE** y asegurate que el campo de **id** contenga el valor **110**, luego haz click en el botón con el icono de copiar

Oracle Database Actions | SQL

Navegador: TDC, Tablas: Buscar...

[Hoja de trabajo] SELECT ID, DESCRIPTION, COMPLETED, ASSIGNEE, CREATED_BY FROM TDC.TODOS;

Resultado de la Consulta

id	description	completed
1	Nullam ut nisi a odi	N
2	nunc. Quisque orna	Y
3	mauris, aliquam eu,	Y
4	auque scelerisque n	Y
5	eu, euismod ac, ferr	N
6	Nam consequat dol	Y
7	enim. Etiam imperd	N

curl para la tabla TODOS

GET ALL: Rellene las entradas para sustituir los valores del comando

GET Single: Nombre de sustitución Valor

POST: id 110

BATCH LOAD

PUT

DELETE

curl --location --request DELETE \
 'https://.../ords/tdc/todos/110'

Cerrar

22. En la terminal, pega el comando que copiaste, si tienes **jq** instalado, puedes añadir **| jq** al comando pegado como se muestra a continuación:

```
<COMANDO_PEGADO> | jq
```

para formatear el JSON que recibamos del endpoint, de otro modo el JSON aparecerá sin espacios u otro separador.

Observa que el método DELETE eliminará una fila y que el JSON devuelto contiene el número de filas eliminadas

```
mingw64 MINGW64 ~
$ curl --location --request DELETE \
  https://api.tdcommons.org/ords/tdc/todos/110' | jq
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left  Speed
100  17    0  17    0    0    21    0  --:--:-- --:--:-- --:--:--    21
{
  "rowsDeleted": 1
}
```

Conclusión

Al terminar las secciones de paso a paso anteriores, habremos aprendido:

- Cómo cargar datos mediante Database Actions
- Cómo mostrar el reporte de datos de una tabla usando Database Actions
- Cómo habilitar tablas como "endpoints" REST para su acceso mediante cURL