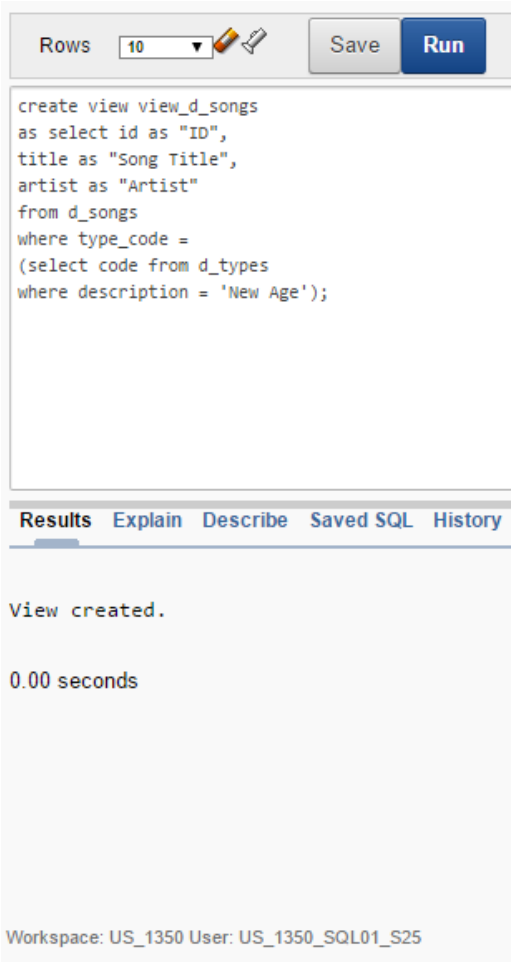
ORACLE ACADEMY

# Section 11 Lesson 1: Creating Views

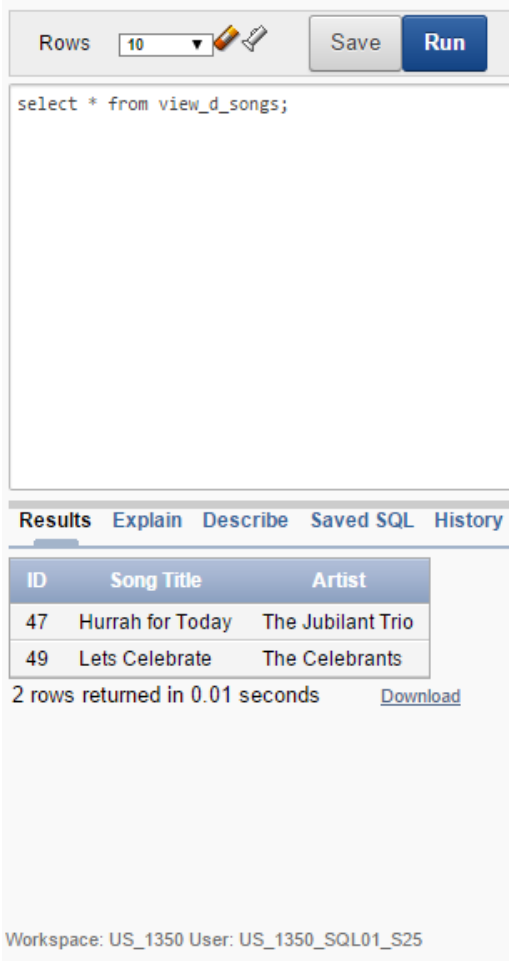
## Try It / Solve It

1. What are three uses for a view from a DBA's perspective?
   a Views can be used to retrieve data from several tables, providing data independence for users.
   b Views provide groups of users with access to data according to their particular permissions or criteria.
   c Views restrict access to base table data because the view can display selective columns from the table.

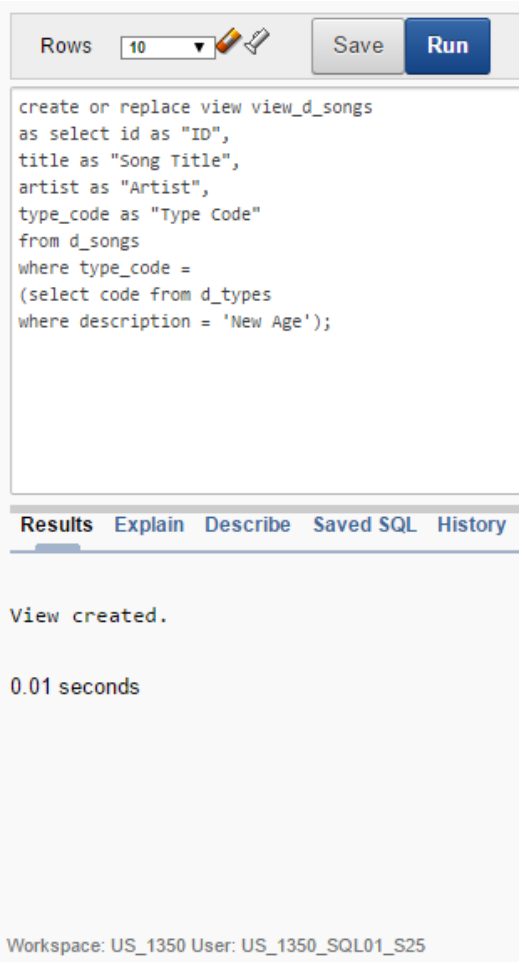
2. Create a simple view called view_d_songs that contains the ID, title, and artist from the DJs on Demand table for each "New Age" type code. In the subquery, use the alias "Song Title" for the title column.



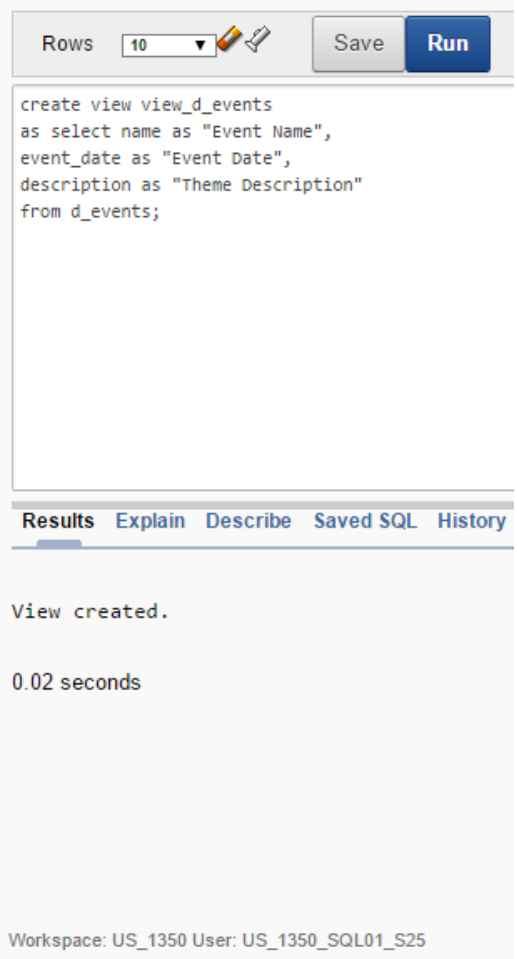
3. SELECT * FROM view_d_songs. What was returned?

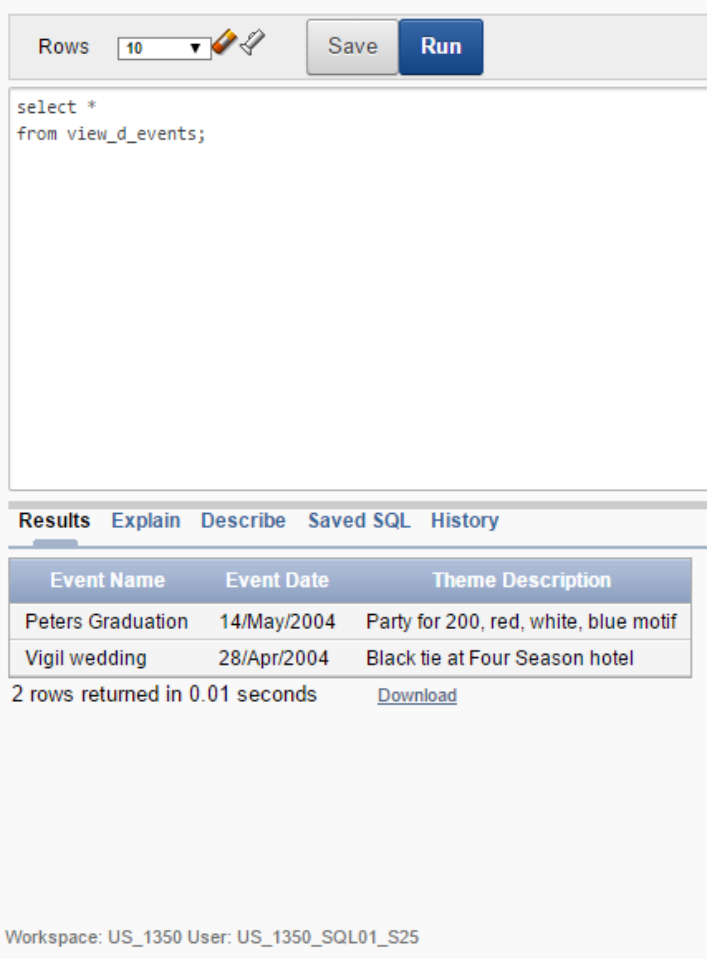


4. REPLACE view_d_songs. Add type_code to the column list. Use aliases for all columns.

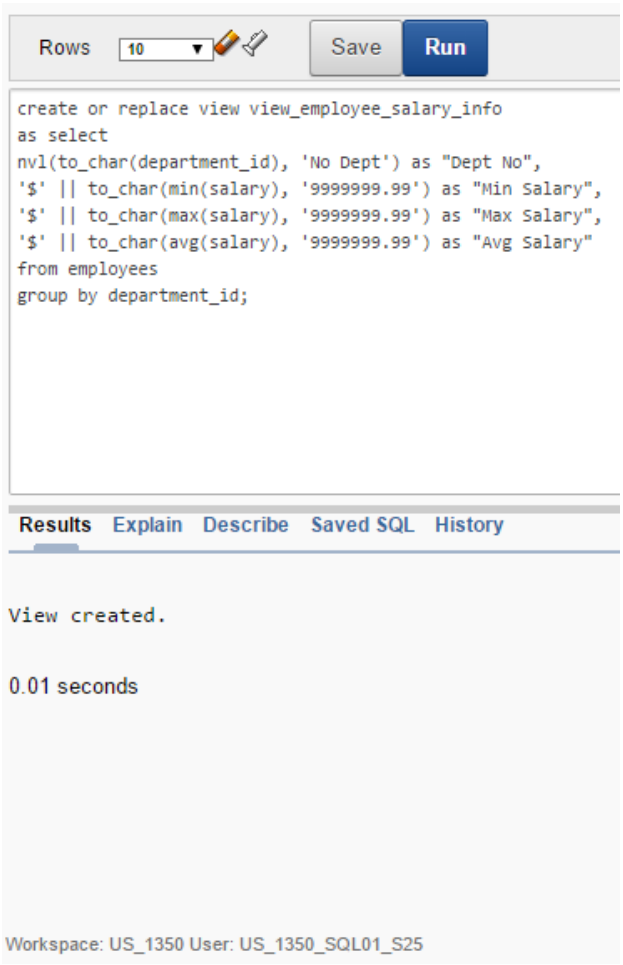


5. Jason Tsang, the disk jockey for DJs on Demand, needs a list of the past events and those planned for the coming months so he can make arrangements for each event's equipment setup. As the company manager, you do not want him to have access to the price that clients paid for their events. Create a view for Jason to use that displays the name of the event, the event date, and the theme description. Use aliases for each column name.



6. It is company policy that only upper-level management be allowed access to individual employee salaries. The department managers, however, need to know the minimum, maximum, and average salaries, grouped by department. Use the Oracle database to prepare a view that displays the needed information for department managers.



Rows 10 Save Run

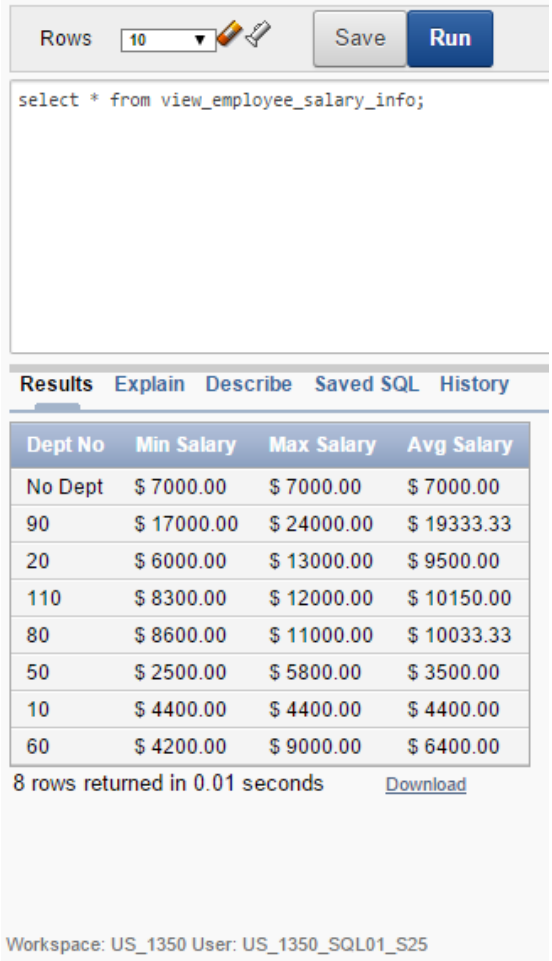
```
create or replace view view_employee_salary_info
as select
nvl(to_char(department_id), 'No Dept') as "Dept No",
'$' || to_char(min(salary), '9999999.99') as "Min Salary",
'$' || to_char(max(salary), '9999999.99') as "Max Salary",
'$' || to_char(avg(salary), '9999999.99') as "Avg Salary"
from employees
group by department_id;
```

Results Explain Describe Saved SQL History

View created.

0.01 seconds

Workspace: US_1350 User: US_1350_SQL01_S25



Rows 10 Save Run

```
select * from view_employee_salary_info;
```

Results Explain Describe Saved SQL History

| Dept No | Min Salary | Max Salary | Avg Salary |
|---|---|---|---|
| No Dept | $ 7000.00 | $ 7000.00 | $ 7000.00 |
| 90 | $ 17000.00 | $ 24000.00 | $ 19333.33 |
| 20 | $ 6000.00 | $ 13000.00 | $ 9500.00 |
| 110 | $ 8300.00 | $ 12000.00 | $ 10150.00 |
| 80 | $ 8600.00 | $ 11000.00 | $ 10033.33 |
| 50 | $ 2500.00 | $ 5800.00 | $ 3500.00 |
| 10 | $ 4400.00 | $ 4400.00 | $ 4400.00 |
| 60 | $ 4200.00 | $ 9000.00 | $ 6400.00 |

8 rows returned in 0.01 seconds Download

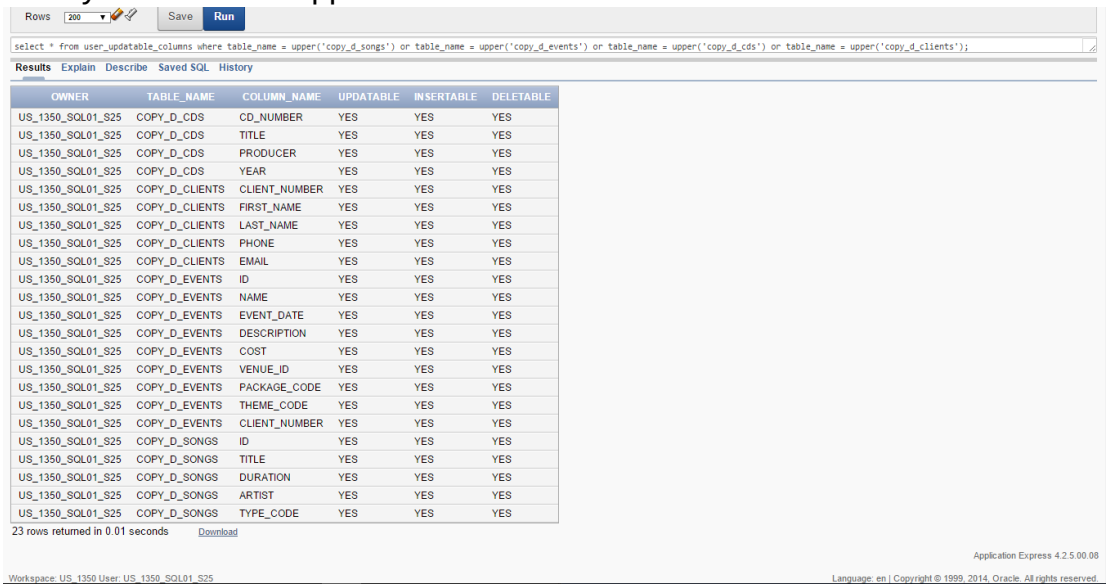
Workspace: US_1350 User: US_1350_SQL01_S25

# Section 11 Lesson 2: DML Operations and Views

## Try It / Solve It

Use the DESCRIBE statement to verify that you have tables named copy_d_songs, copy_d_events, copy_d_cds, and copy_d_clients in your schema. If you don't, write a query to create a copy of each.

1. Query the data dictionary USER_UPDATABLE_COLUMNS to make sure the columns in the base tables will allow UPDATE, INSERT, or DELETE. Use a SELECT statement or the Browse Data Dictionary feature in HTML DB. All table names in the data dictionary are stored in uppercase.


Rows 200 Save Run

```
select * from user_updatable_columns where table_name = upper('copy_d_songs') or table_name = upper('copy_d_events') or table_name = upper('copy_d_cds') or table_name = upper('copy_d_clients');
```

Results Explain Describe Saved SQL History

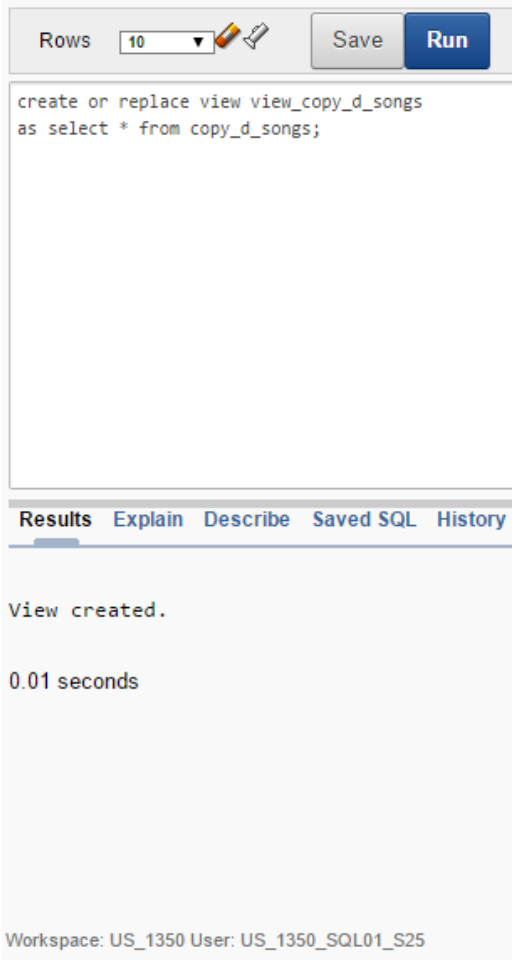
| OWNER | TABLE_NAME | COLUMN_NAME | UPDATABLE | INSERTABLE | DELETABLE |
|---|---|---|---|---|---|
| US_1350_SQL01_S25 | COPY_D_CDS | CD_NUMBER | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CDS | TITLE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CDS | PRODUCER | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CDS | YEAR | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CLIENTS | CLIENT_NUMBER | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CLIENTS | FIRST_NAME | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CLIENTS | LAST_NAME | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CLIENTS | PHONE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_CLIENTS | EMAIL | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | ID | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | NAME | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | EVENT_DATE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | DESCRIPTION | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | COST | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | VENUE_ID | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | PACKAGE_CODE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | THEME_CODE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_EVENTS | CLIENT_NUMBER | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_SONGS | ID | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_SONGS | TITLE | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_SONGS | DURATION | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_SONGS | ARTIST | YES | YES | YES |
| US_1350_SQL01_S25 | COPY_D_SONGS | TYPE_CODE | YES | YES | YES |

23 rows returned in 0.01 seconds Download

Application Express 4.2.5.00.08

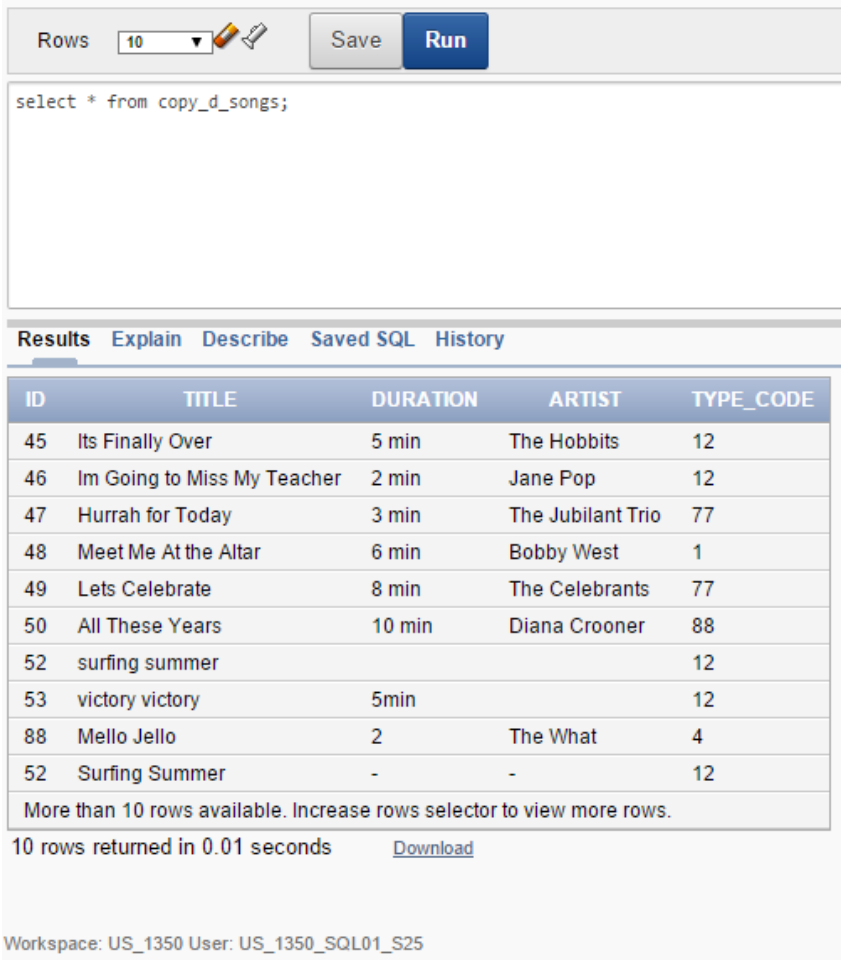
Workspace: US_1350 User: US_1350_SQL01_S25 Language: en | Copyright © 1999, 2014, Oracle. All rights reserved.

2. Use the CREATE or REPLACE option to create a view of *all* the columns in the copy_d_songs table called view_copy_d_songs.

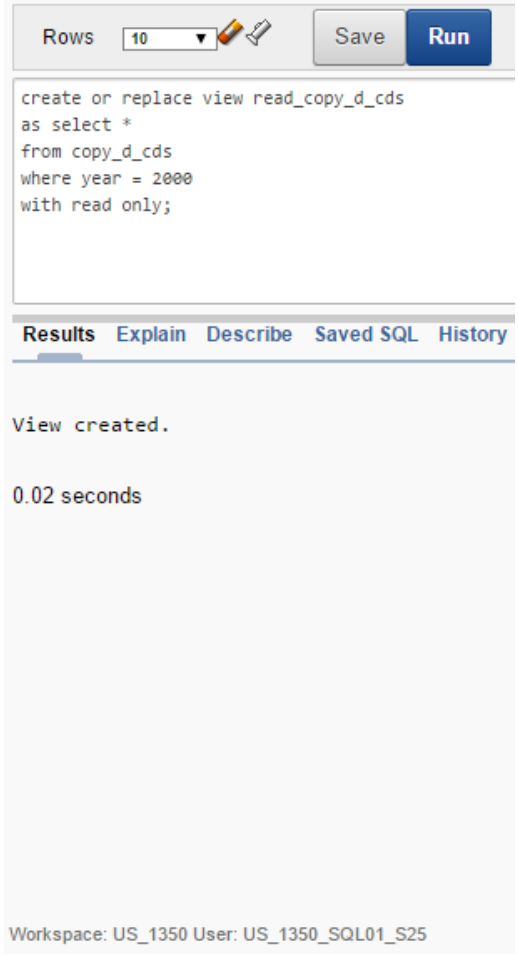


3. Use view_copy_d_songs to INSERT the following data into the underlying copy_d_songs table. Execute a SELECT * from copy_d_songs to verify your DML command. See the graphic.

| ID | TITLE | DURATION | ARTIST | TYPE_CODE |
|---|---|---|---|---|
| 88 | Mello Jello | 2 | The What | 4 |

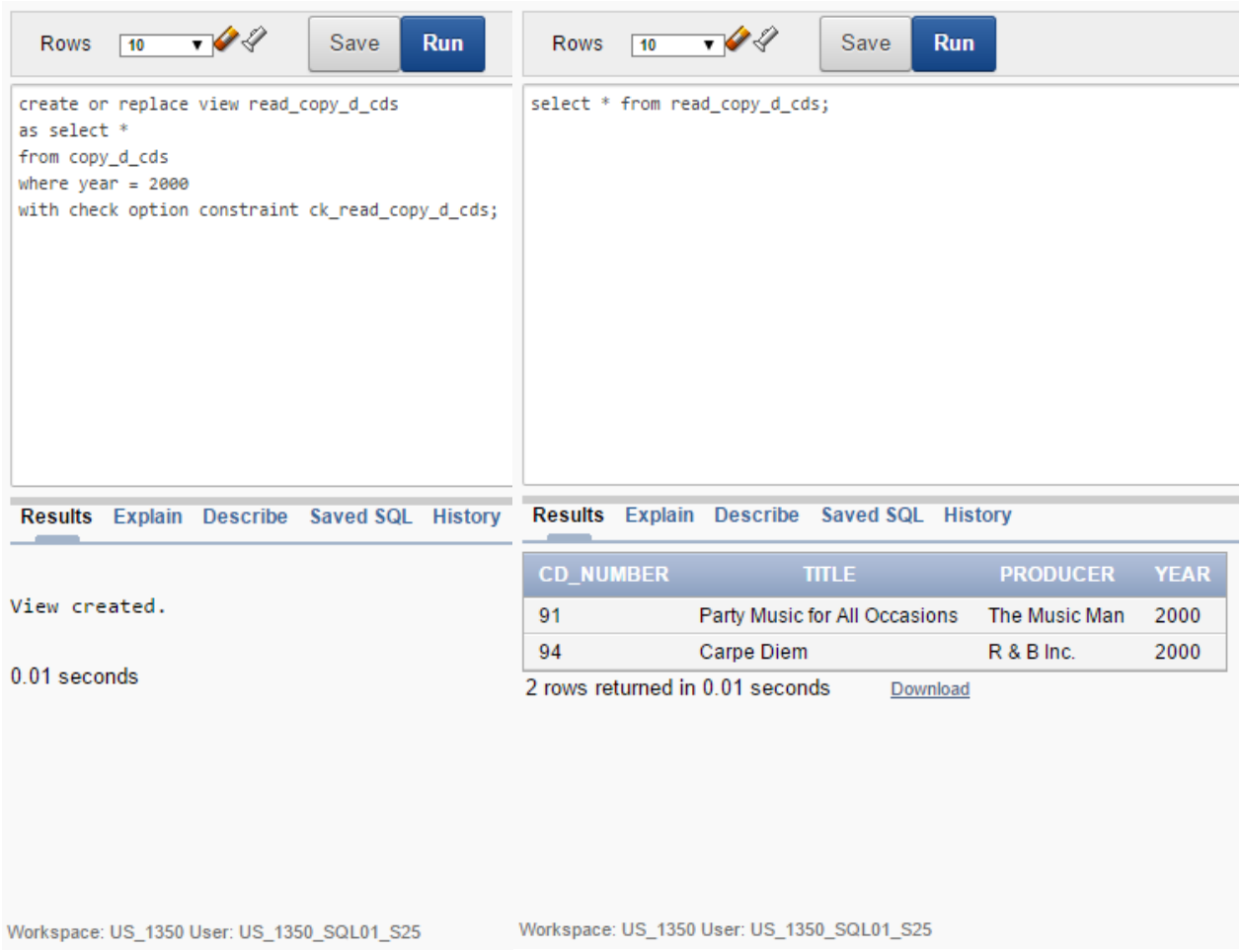


4. Create a view based on the DJs on Demand COPY_D_CDS table. Name the view read_copy_d_cds. Select all columns to be included in the view. Add a WHERE clause to restrict the year to 2000. Add the WITH READ ONLY option.

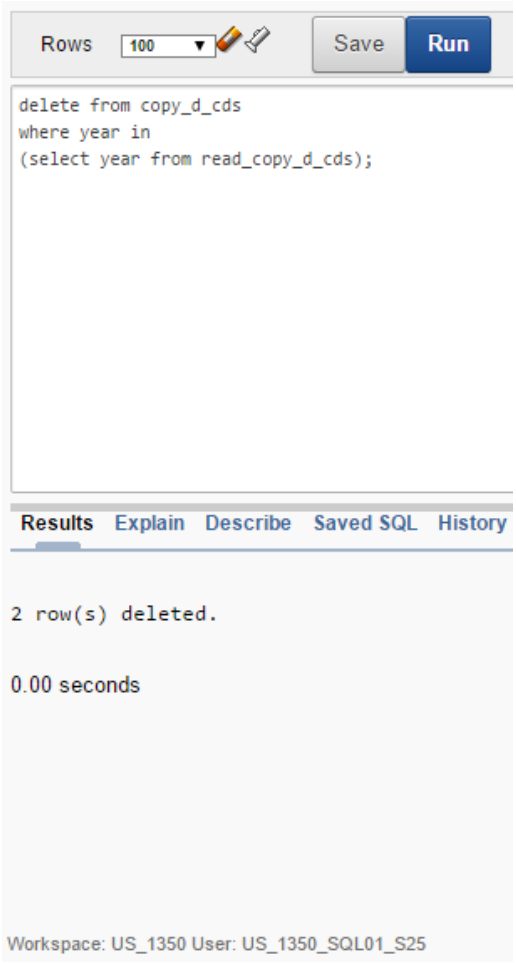


5. Using the read_copy_d_cds view, execute a DELETE FROM read_copy_d_cds WHERE cd_number = 90; Action cannot be performed.

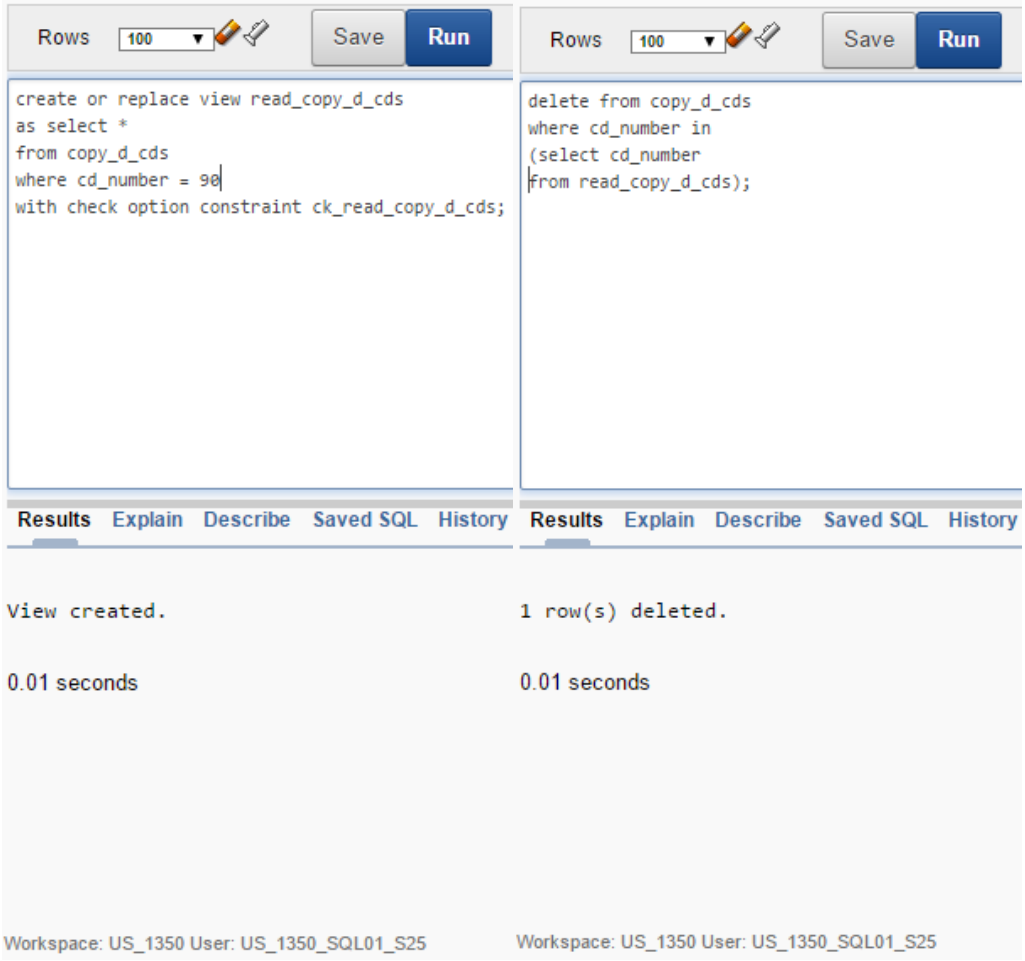
6. Use REPLACE to modify read_copy_d_cds. Replace the READ ONLY option with WITH CHECK OPTION CONSTRAINT ck_read_copy_d_cds. Execute a SELECT * statement to verify that the view exists.



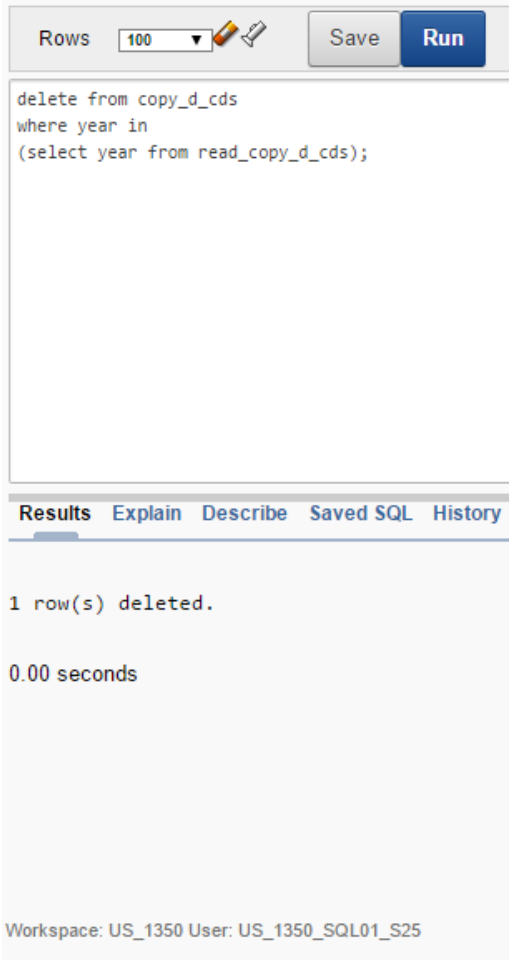
7. Use the read_copy_d_cds view to delete any CD of year 2000 from the underlying copy_d_cds.

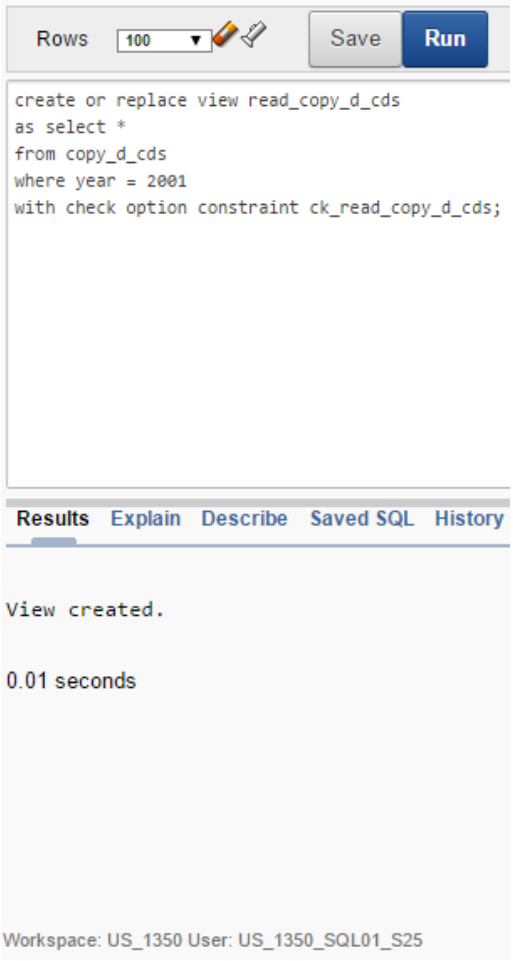


8. Use the read_copy_d_cds view to delete cd_number 90 from the underlying copy_d_cds table.

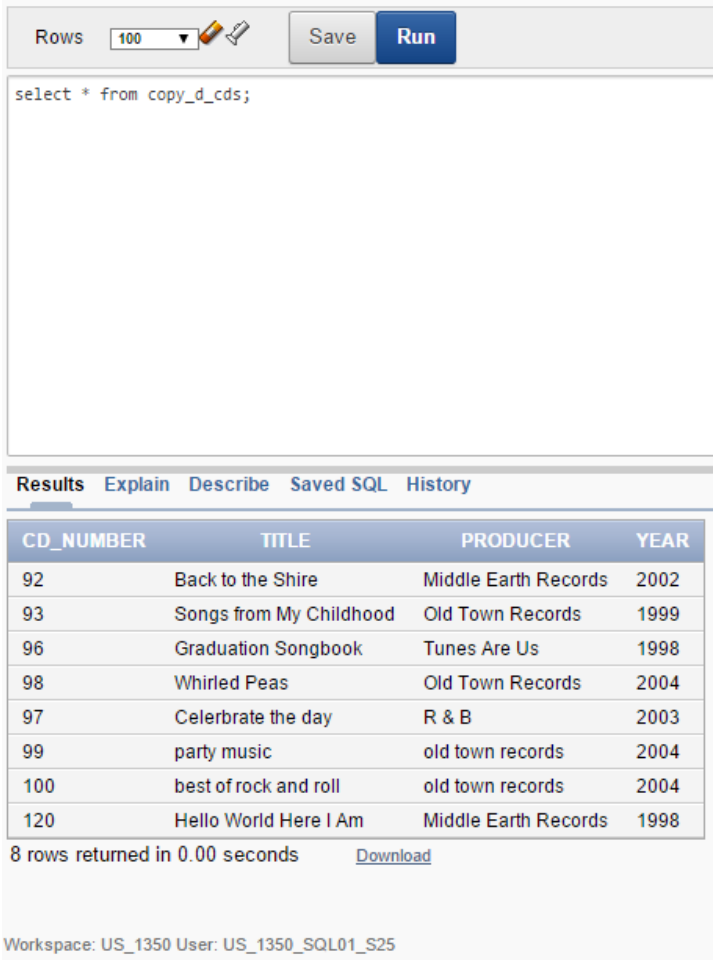


9. Use the read_copy_d_cds view to delete year 2001 records.




10. Execute a SELECT * statement for the base table copy_d_cds. What rows were deleted?



Rows 100 Save Run

```
select * from copy_d_cds;
```

Results Explain Describe Saved SQL History

| CD_NUMBER | TITLE | PRODUCER | YEAR |
|---|---|---|---|
| 92 | Back to the Shire | Middle Earth Records | 2002 |
| 93 | Songs from My Childhood | Old Town Records | 1999 |
| 96 | Graduation Songbook | Tunes Are Us | 1998 |
| 98 | Whirled Peas | Old Town Records | 2004 |
| 97 | Celerbrate the day | R & B | 2003 |
| 99 | party music | old town records | 2004 |
| 100 | best of rock and roll | old town records | 2004 |
| 120 | Hello World Here I Am | Middle Earth Records | 1998 |

8 rows returned in 0.00 seconds Download

Workspace: US_1350 User: US_1350_SQL01_S25

11. What are the restrictions on modifying data through a view?
    You cannot modify data through a view if the view contains: Group functions, a GROUP BY clause, a DISTINCT keyword, or the pseudocolumn ROWNUM Keyword, or columns defined by expressions.
12. What is Moore's Law? Do you consider that it will continue to apply indefinitely? Support your opinion with research from the internet. Moore's Law states that the number of transistors on a given chip can be doubled every two years. This has been relatively true in recent years, however given that it is not a straight line at a 45 degree angle and there appears to be an exponential curvature, this law is not an indefinite proposal.

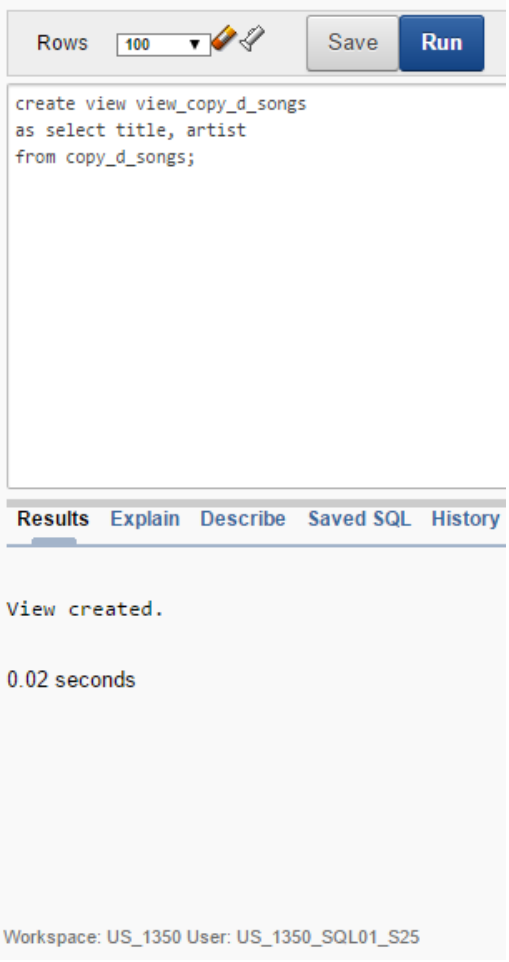
13. What is the "singularity" in terms of computing? The singularity is a hypothetical future citing the creation of highly intelligent machines which are deemed as "superintelligent"- having an artificial capacity beyond human capabilities.

ORACLE ACADEMY

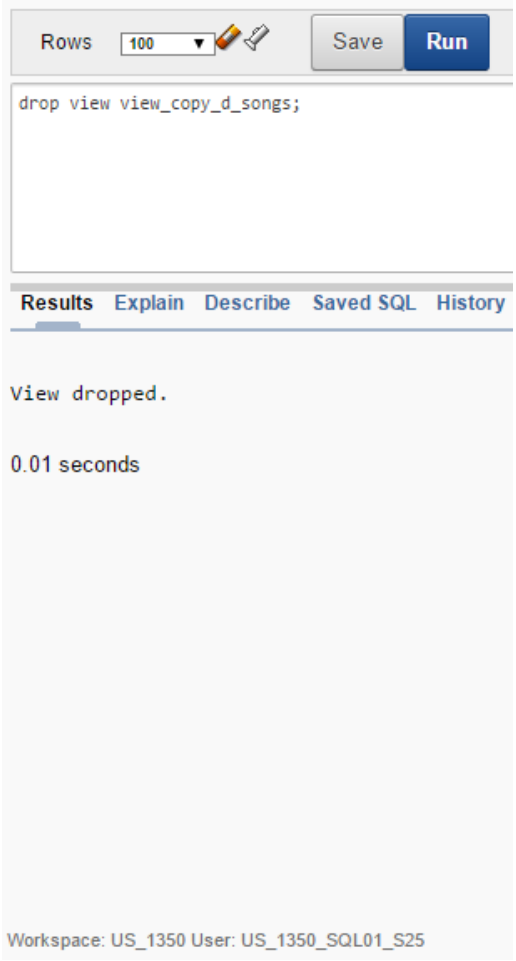
# Section 11 Lesson 3: Managing Views

## Try It / Solve It

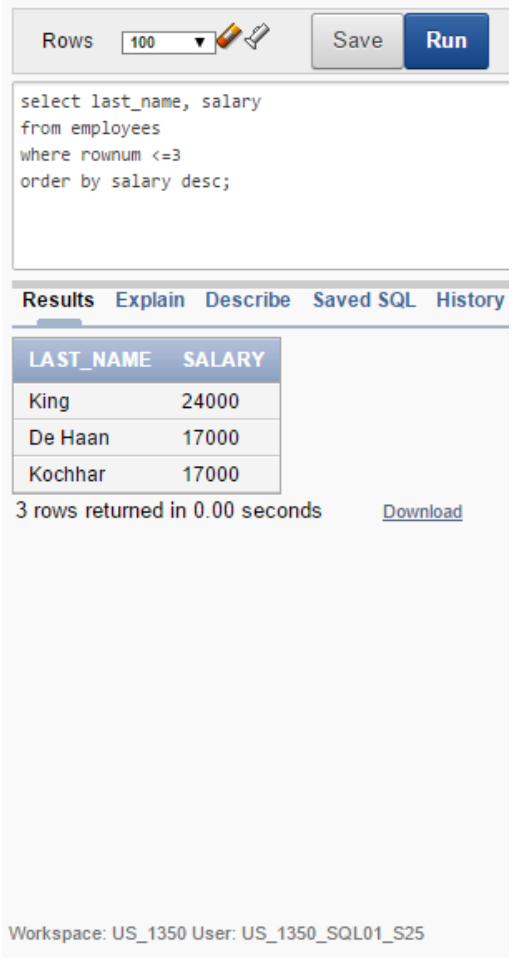
1. Create a view from the copy_d_songs table called view_copy_d_songs that includes only the title and artist. Execute a SELECT * statement to verify that the view exists.




2. Issue a DROP view_copy_d_songs. Execute a SELECT * statement to verify that the view has been deleted.



3. Create a query that selects the last name and salary from the Oracle database. Rank the salaries from highest to lowest for the top three employees.




4. Construct an inline view from the Oracle database that lists the last name, salary, department ID, and maximum salary for each department. Hint: One query will need to calculate maximum salary by department ID.

Rows 10 Save Run

```
select e.employee_id, e.last_name,
m.maxsalary, e.department_id
from employees e join
(select department_id, max(salary) maxsalary
from employees
group by department_id) m
on m.department_id = e.department_id
order by m.maxsalary desc;
```

Results Explain Describe Saved SQL History

| EMPLOYEE_ID | LAST_NAME | MAXSALARY | DEPARTMENT_ID |
|---|---|---|---|
| 102 | De Haan | 24000 | 90 |
| 101 | Kochhar | 24000 | 90 |
| 100 | King | 24000 | 90 |
| 201 | Hartstein | 13000 | 20 |
| 202 | Fay | 13000 | 20 |
| 205 | Higgins | 12000 | 110 |
| 206 | Gietz | 12000 | 110 |
| 174 | Abel | 11000 | 80 |
| 176 | Taylor | 11000 | 80 |
| 149 | Zlotkey | 11000 | 80 |

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.02 seconds Download

Workspace: US_1350 User: US_1350_SQL01_S25

5. Create a query that will return the staff members of Global Fast Foods ranked by salary from lowest to highest.