

Database Programming with SQL 4-3: Date Functions

Practice Activities

# Objectives

* Select and apply the single-row functions MONTHS\_BETWEEN, ADD\_MONTHS, NEXT\_DAY, LAST\_DAY, ROUND, and TRUNC that operate on date data
* Explain how date functions transform Oracle dates into date data or numeric values
* Demonstrate proper use of the arithmetic operators with dates
* Demonstrate the use of SYSDATE and date functions
* State the implications for world businesses to be able to easily manipulate data stored in date format

# Vocabulary

Identify the vocabulary word for each definition below.

|  |  |
| --- | --- |
| **SYSDATE** | A function that returns the current date and time of the database server. |
| **ADD\_MONTHS** | Add calendar months to date |
| **LAST\_DAY** | Last day of the month |
| **NEXT\_DAY** | Next day of the date specified |
| **MONTHS\_BETWEEN** | Number of months between due dates |

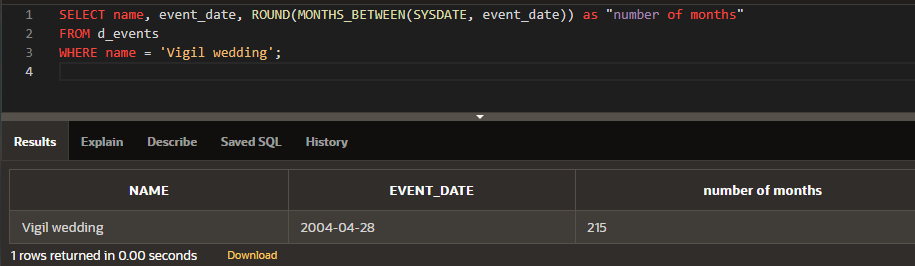
# Try It / Solve It

1. For DJs on Demand, display the number of months between the event\_date of the Vigil wedding and today’s date. Round to the nearest month.

SELECT name, event\_date, ROUND(MONTHS\_BETWEEN(SYSDATE, event\_date)) as "number of months"

FROM d\_events

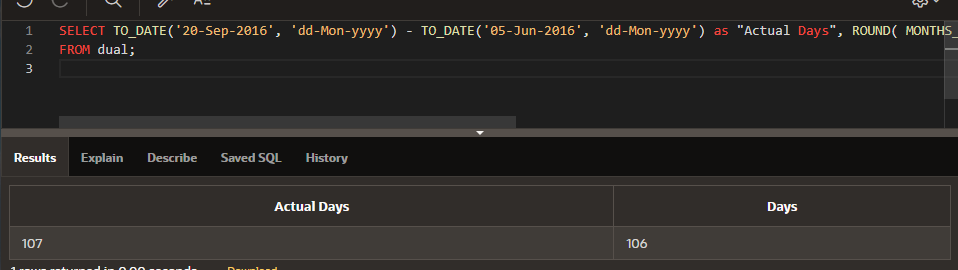
WHERE name = 'Vigil wedding';



1. Display the days between the start of last summer’s school vacation break and the day school started this year. Assume 30.5 days per month. Name the output “Days.”

SELECT TO\_DATE('20-Sep-2016', 'dd-Mon-yyyy') - TO\_DATE('05-Jun-2016', 'dd-Mon-yyyy') as "Actual Days", ROUND( MONTHS\_BETWEEN(TO\_DATE('20-Sep-2016', 'dd-Mon-yyyy'), TO\_DATE('05-Jun-2016', 'dd-Mon-yyyy'))\*30.5, 0) as "Days"

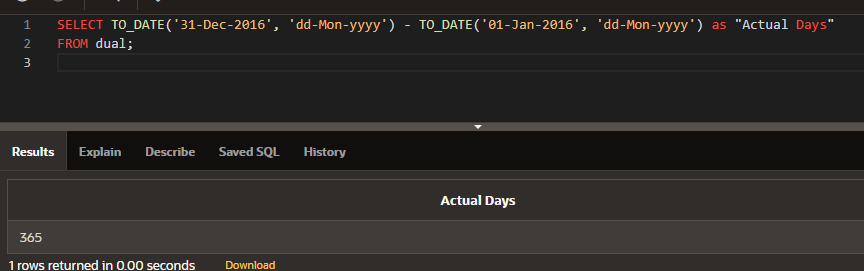
FROM dual;



1. Display the days between January 1 and December 31.

SELECT TO\_DATE('31-Dec-2016', 'dd-Mon-yyyy') - TO\_DATE('01-Jan-2016', 'dd-Mon-yyyy') as "Actual Days"

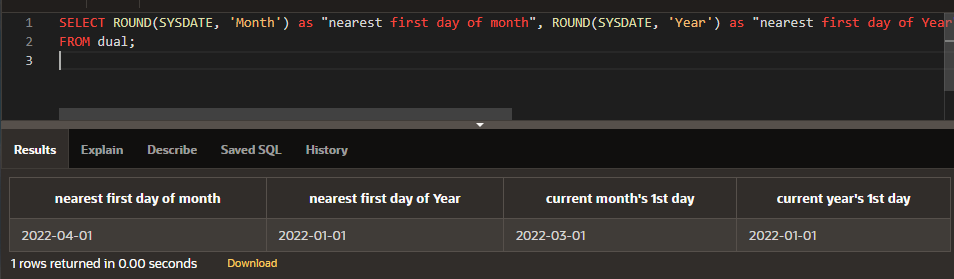
FROM dual;



1. Using one statement, round today's date to the nearest month and nearest year, and truncate it to the nearest month and nearest year. Use an alias for each column.

SELECT ROUND(SYSDATE, 'Month') as "nearest first day of month", ROUND(SYSDATE, 'Year') as "nearest first day of Year", TRUNC(SYSDATE, 'Month') as "current month's 1st day", TRUNC(SYSDATE, 'Year') as "current year's 1st day"

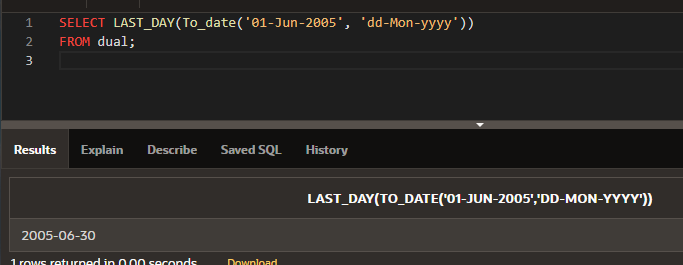
FROM dual;



1. What is the last day of the month for June 2005? Use an alias for the output.

SELECT LAST\_DAY(To\_date('01-Jun-2005', 'dd-Mon-yyyy'))

FROM dual;

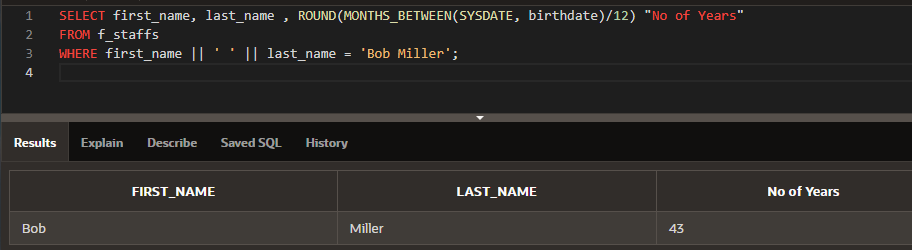


1. Display the number of years between the Global Fast Foods employee Bob Miller’s birthday and today. Round to the nearest year.

SELECT first\_name, last\_name , ROUND(MONTHS\_BETWEEN(SYSDATE, birthdate)/12) "No of Years"

FROM f\_staffs

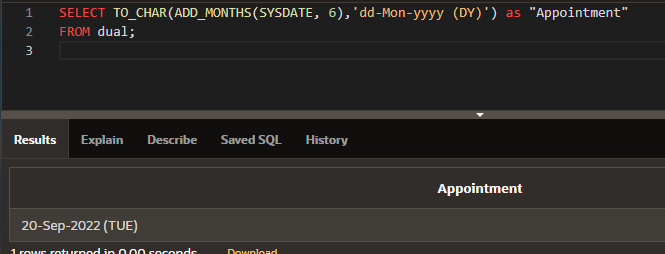
WHERE first\_name || ' ' || last\_name = 'Bob Miller';



1. Your next appointment with the dentist is six months from today. On what day will you go to the dentist? Name the output, “Appointment.”

SELECT TO\_CHAR(ADD\_MONTHS(SYSDATE, 6),'dd-Mon-yyyy (DY)') as "Appointment"

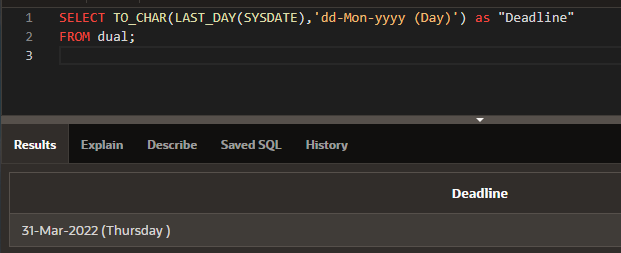
FROM dual;



1. The teacher said you have until the last day of this month to turn in your research paper. What day will this be? Name the output, “Deadline.”

SELECT TO\_CHAR(LAST\_DAY(SYSDATE),'dd-Mon-yyyy (Day)') as "Deadline"

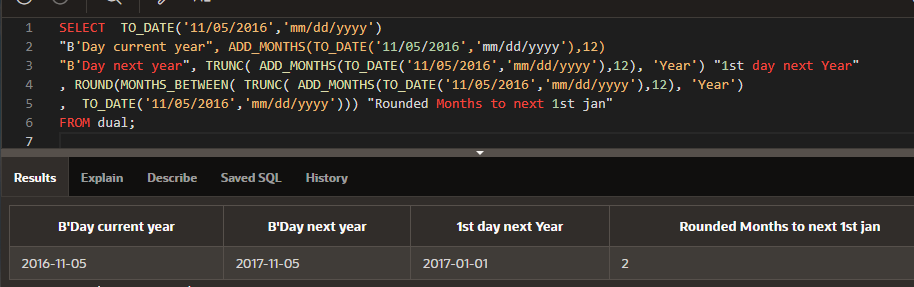
FROM dual;



1. How many months between your birthday this year and January 1 next year?

SELECT TO\_DATE('11/05/2016','mm/dd/yyyy') "B'Day current year", ADD\_MONTHS(TO\_DATE('11/05/2016','mm/dd/yyyy'),12) "B'Day next year", TRUNC( ADD\_MONTHS(TO\_DATE('11/05/2016','mm/dd/yyyy'),12), 'Year') "1st day next Year" , ROUND(MONTHS\_BETWEEN( TRUNC( ADD\_MONTHS(TO\_DATE('11/05/2016','mm/dd/yyyy'),12), 'Year') , TO\_DATE('11/05/2016','mm/dd/yyyy'))) "Rounded Months to next 1st jan"

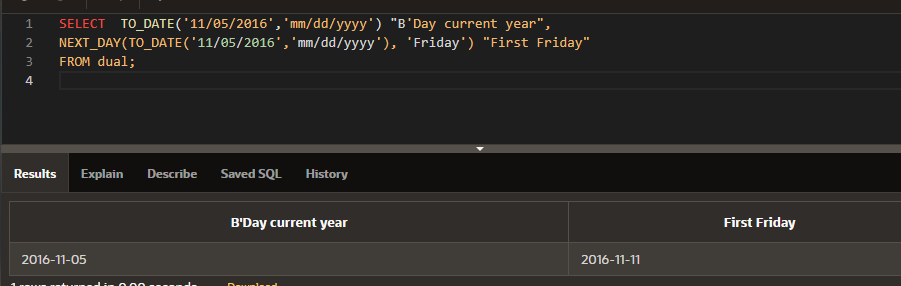
FROM dual;



1. What’s the date of the next Friday after your birthday this year? Name the output, “First Friday.”

SELECT  TO\_DATE('11/05/2016','mm/dd/yyyy') "B'Day current year", NEXT\_DAY(TO\_DATE('11/05/2016','mm/dd/yyyy'), 'Friday') "First Friday"

FROM dual;



1. Name a date function that will return a number.

MONTHS\_BETWEEN

1. Name a date function that will return a date.

ADD\_MONTHS

1. Give one example of why it is important for businesses to be able to manipulate date data?

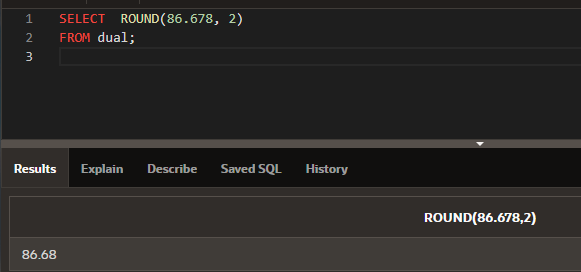
Say, my credit card payment is due on 09th every month. But if 09th is on weekend or bank holiday, this due date has to be shifted to next Monday while triggering late payment workflow. This is a very common and nearest to me example of why business needs to manipulate date data.

Extension Exercises

1. Using DUAL, write a statement that will convert 86.678 to 86.68.

SELECT  ROUND(86.678, 2)

FROM dual;

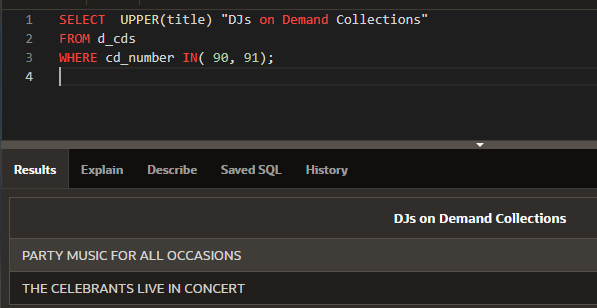


1. Write a statement that will display the DJs on Demand CD titles for cd\_numbers 90 and 91 in uppercase in a column headed “DJs on Demand Collections.”

SELECT  UPPER(title) "DJs on Demand Collections"

FROM d\_cds

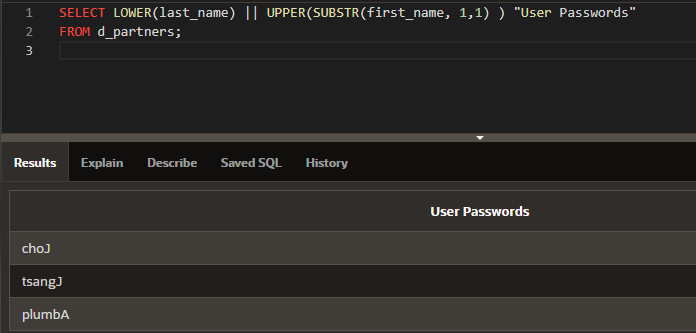
WHERE cd\_number IN( 90, 91);



1. Write a statement that will create computer usernames for the DJs on Demand partners. The usernames will be the lowercase letters of the last name + the uppercase first letter in the first name. Title the column “User Passwords.” For example, Mary Smythers would be smythersM.

SELECT LOWER(last\_name) || UPPER(SUBSTR(first\_name, 1,1) ) "User Passwords"

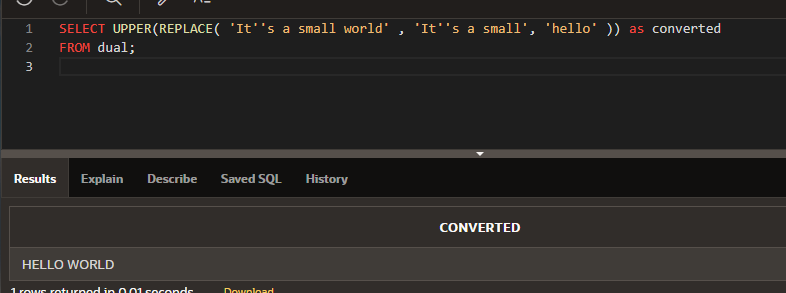
FROM d\_partners;



1. Write a statement that will convert “It’s a small world” to “HELLO WORLD.”

SELECT UPPER(REPLACE( 'It''s a small world' , 'It''s a small', 'hello' )) as converted

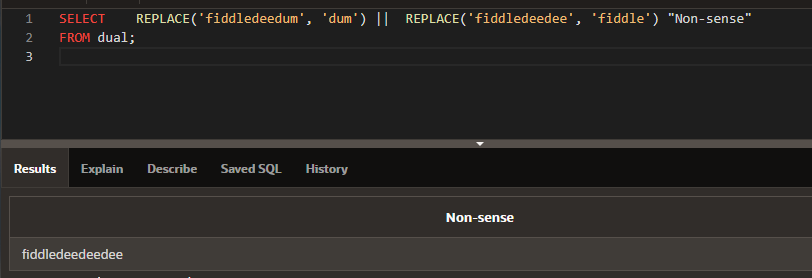
FROM dual;



1. Write a statement that will remove the “fiddle” from “fiddledeedee” and the “dum” from “fiddledeedum.” Display the result “fiddledeedeedee” in a column with the heading “Nonsense.”

SELECT    REPLACE('fiddledeedum', 'dum') ||  REPLACE('fiddledeedee', 'fiddle') "Non-sense"

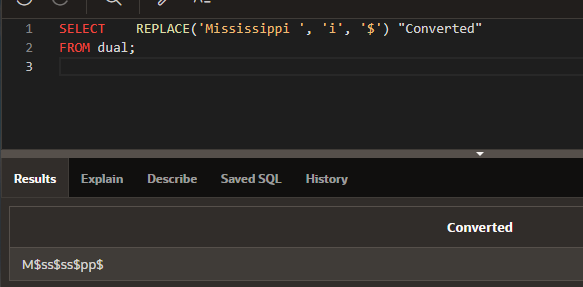
FROM dual;



1. Replace every “i” in Mississippi with “$.”

SELECT    REPLACE('Mississippi ', 'i', '$') "Converted"

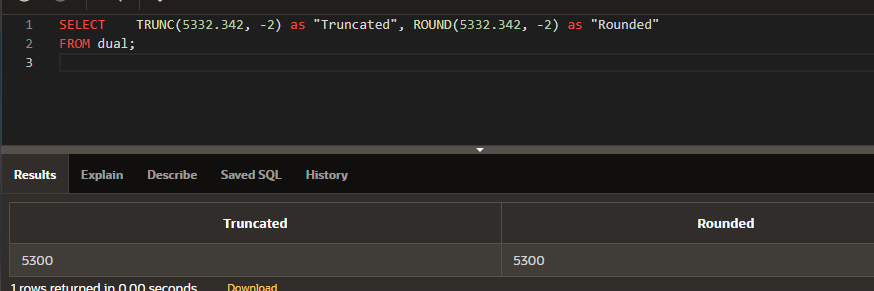
FROM dual;



1. Using DUAL, convert 5332.342 to 5300.

SELECT    TRUNC(5332.342, -2) as "Truncated", ROUND(5332.342, -2) as "Rounded"

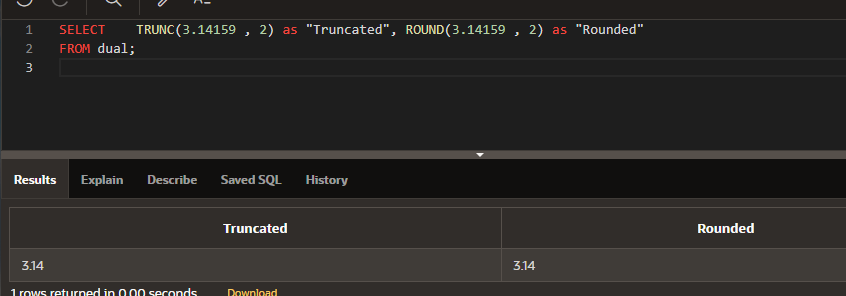
FROM dual;



1. Using DUAL, convert 3.14159 to 3.14.

SELECT    TRUNC(3.14159 , 2) as "Truncated", ROUND(3.14159 , 2) as "Rounded"

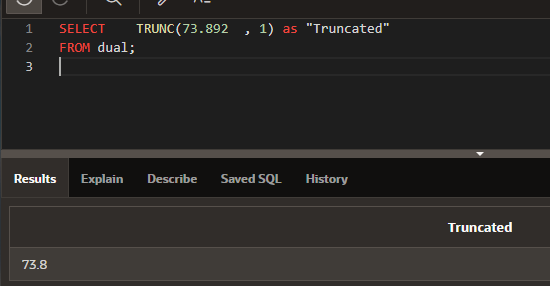
FROM dual;



1. Using DUAL, convert 73.892 to 73.8.

SELECT    TRUNC(73.892  , 1) as "Truncated"

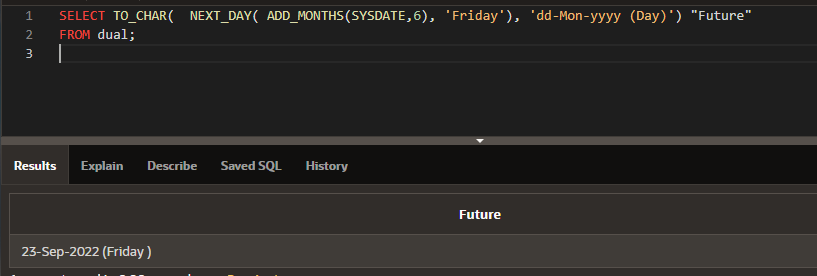
FROM dual;



1. What is the next Friday six months from now? Label the column “Future.”

SELECT TO\_CHAR(  NEXT\_DAY( ADD\_MONTHS(SYSDATE,6), 'Friday'), 'dd-Mon-yyyy (Day)') "Future"

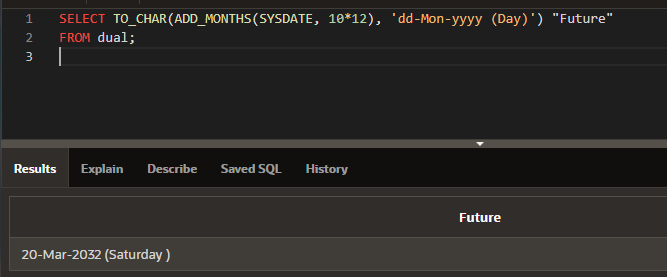
FROM dual;



1. What is the date 10 years from now? Label the column “Future.”

SELECT TO\_CHAR(ADD\_MONTHS(SYSDATE, 10\*12), 'dd-Mon-yyyy (Day)') "Future"

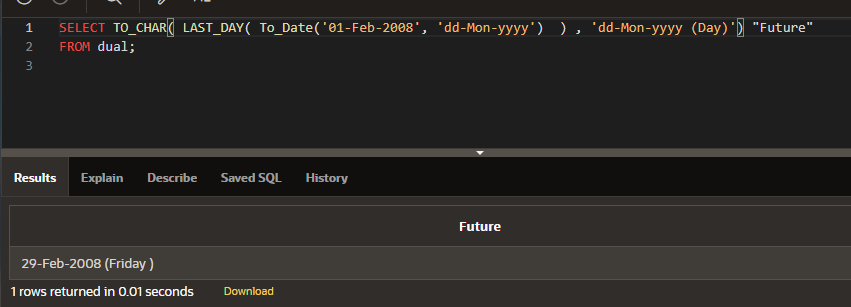
FROM dual;



1. Leap years occur every four years. Remember, 2004 was a leap year. Now create a function that will show the date of the next leap year as 29-Feb-2008. Label the column “Future.”

SELECT TO\_CHAR( LAST\_DAY( To\_Date('01-Feb-2008', 'dd-Mon-yyyy') ) , 'dd-Mon-yyyy (Day)') "Future"

FROM dual;

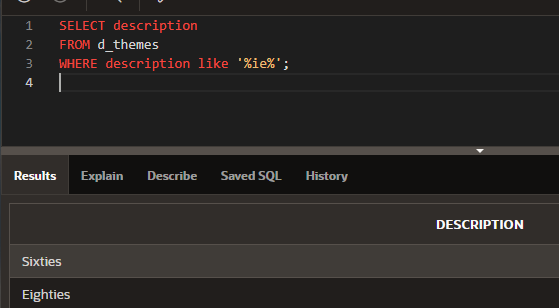


1. Write a statement that will find any of the DJs on Demand CD themes that have an “ie” in their names.

SELECT description

FROM d\_themes

WHERE description like '%ie%';

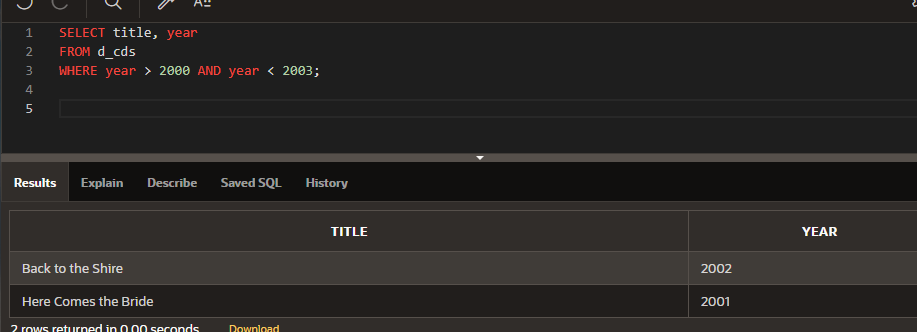


1. Write a statement that will return only the DJs on Demand CDs with years greater than 2000 but less than 2003. Display both the title and year.

SELECT title, year

FROM d\_cds

WHERE year > 2000 AND year < 2003;



1. Write a statement that will return the Oracle database employee’s employee ID and his starting hire dates between January 1, 1997 and today. Display the result ordered from most recently hired to the oldest.

SELECT employee\_id, hire\_date

FROM employees

WHERE hire\_date BETWEEN TO\_DATE( '01-Jan-1997', 'dd-Mon-yyyy') AND SYSDATE

ORDER BY hire\_date DESC;

