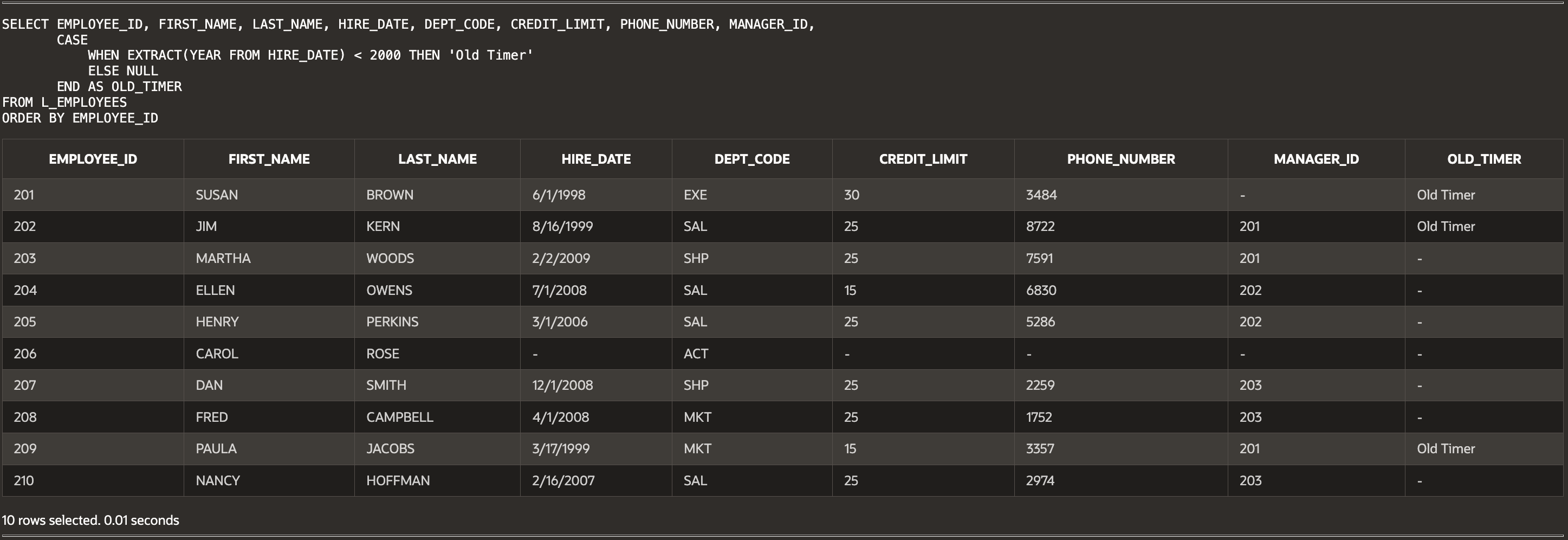
**Week 5 Performance Assessment: Union All, Cross Joins and CrossTab Queries.**

The following questions come from the “Check your understanding” examples of Chapters 15 & 16 in your textbook.

After you are finished, please submit a Microsoft Word file that contains screenshots the SQL queries, the output, and please put a comment line in each query with your name. **Note: No screen shots are required for Questions 6, 7, and 8, which should be done “by hand,” (use the Insert tab to create your own Microsoft Word table) and simply answered in the same document.** Your document should be named **W5\_PA\_Unions2\_Lastname.docx**.

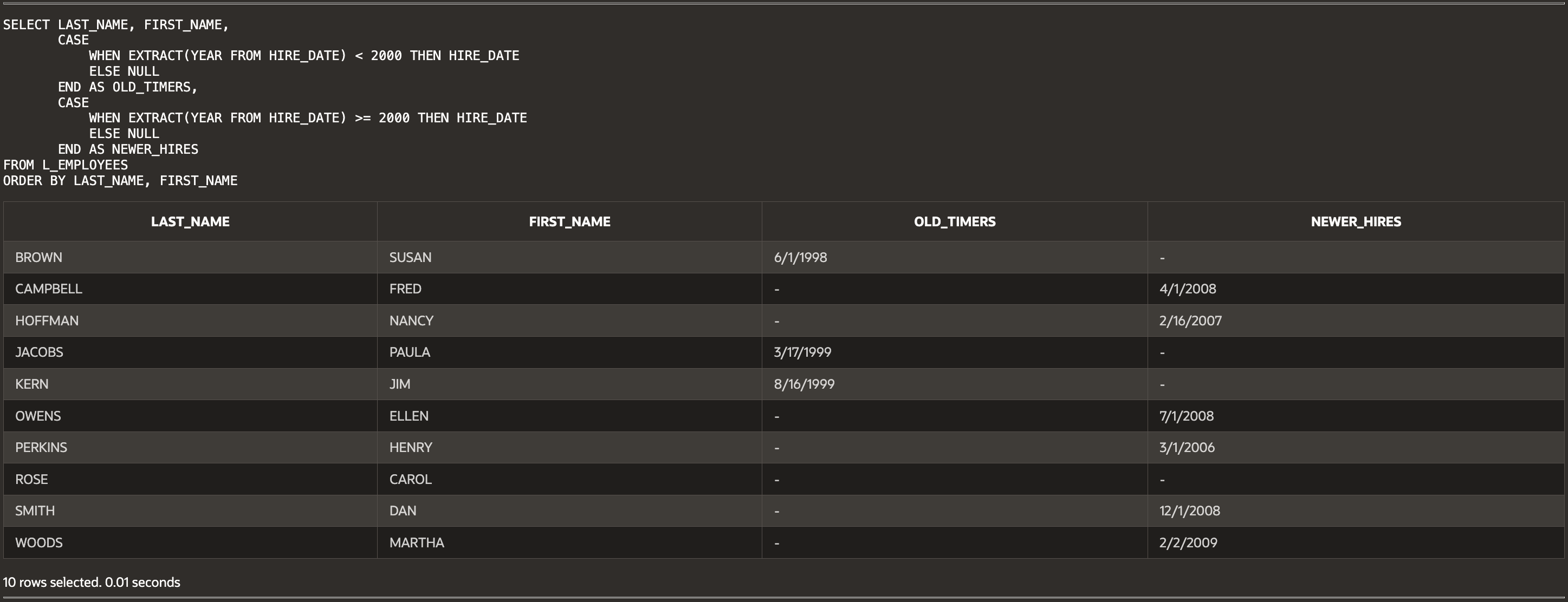
(15-11) Question 1:

List all the rows and columns of the *L\_EMPLOYEES* table. Add a new column that says “Old Timer” for any employee that was hired before the year 2000 and is blank for all other employees. Sort this by the *employee\_id* column.



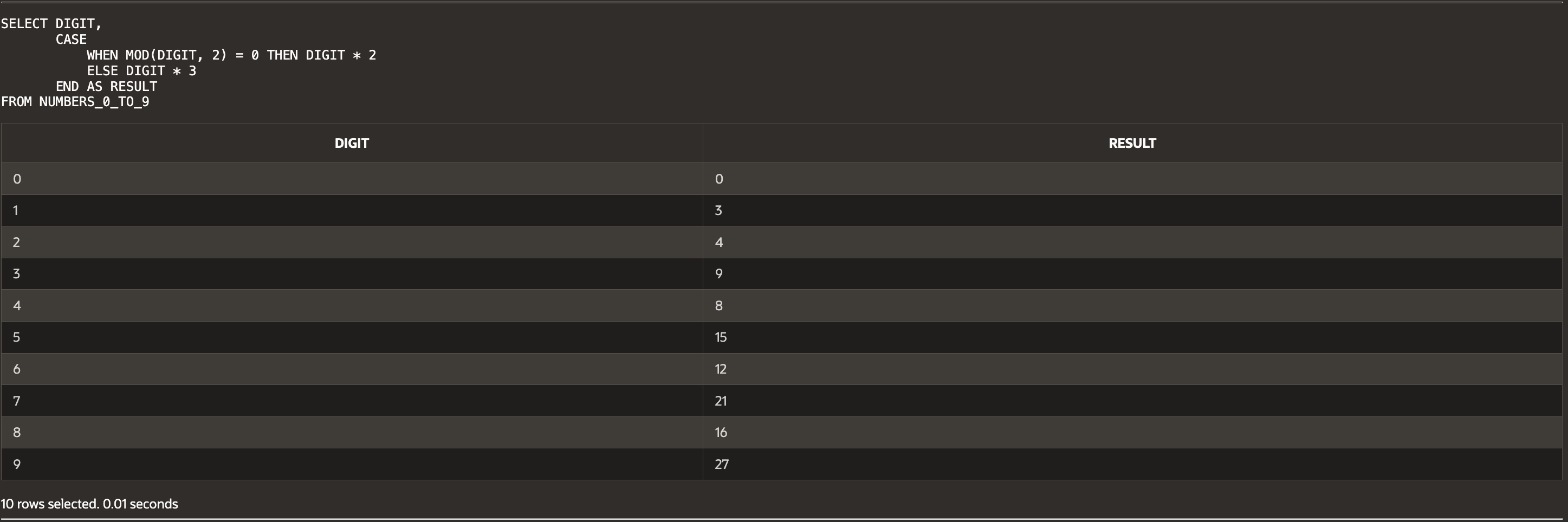
(15-12) Question 2:

List the last name, first name, and hire date of all the people in the *L\_EMPLOYEES* table. Divide the hire date column into two columns: one called “old timers” for people hired before the year 2000, and one called “newer hires” for people hired after that year. Sort the result table by last name and then by first name.



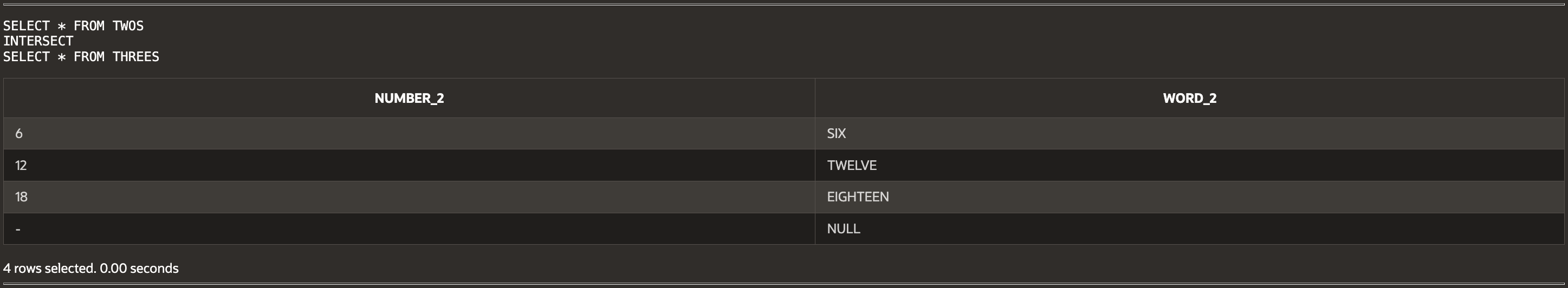
(15-13) Question 3:

The *numbers\_0\_to\_9* table contains the numbers from zero to nine. Multiply all the even numbers by two and multiply all the other numbers by three.



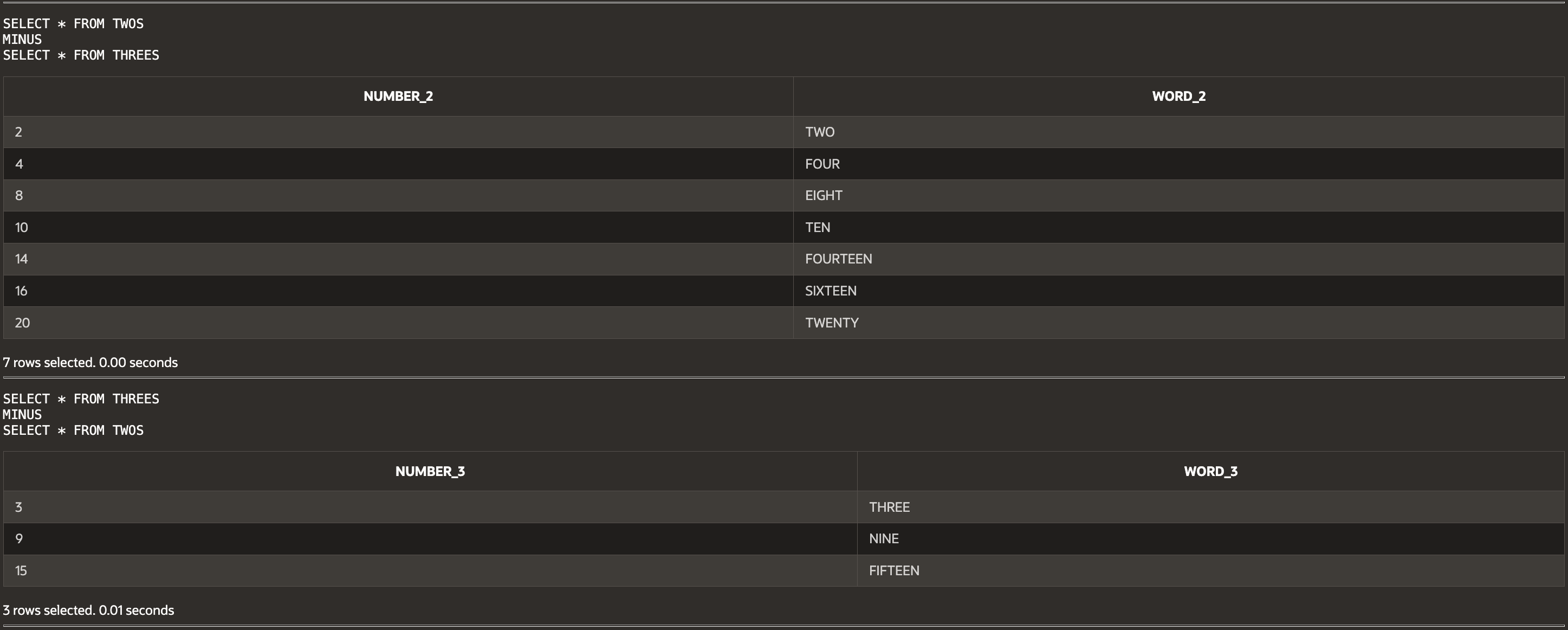
(15-15) Question 4:

Use the method of this section to find the intersection of the *twos* table and the *threes* table.



(15-16) Question 5:

Use the method of this section to find the difference of the *twos* table and the *threes* table.



(16-1) Question 6:

**Insert a table on this document showing the result set of:** the cross join of the following two tables:

|  |  |
| --- | --- |
| **Table 1** | **Table 2** |

|  |  |
| --- | --- |
| U | 10 |

|  |  |
| --- | --- |
| U | 20 |

|  |  |
| --- | --- |
| U | 30 |

|  |  |
| --- | --- |
| U | 40 |

|  |  |
| --- | --- |
| U | 50 |

|  |  |
| --- | --- |
| V | 10 |

|  |  |
| --- | --- |
| V | 20 |

|  |  |
| --- | --- |
| V | 30 |

|  |  |
| --- | --- |
| V | 40 |

|  |  |
| --- | --- |
| V | 50 |

|  |  |
| --- | --- |
| W | 10 |

|  |  |
| --- | --- |
| W | 20 |

|  |  |
| --- | --- |
| W | 30 |

|  |  |
| --- | --- |
| W | 40 |

|  |  |
| --- | --- |
| W | 50 |

|  |  |
| --- | --- |
| X | 10 |

|  |  |
| --- | --- |
| X | 20 |

|  |  |
| --- | --- |
| X | 30 |

|  |  |
| --- | --- |
| X | 40 |

|  |  |
| --- | --- |
| X | 50 |

|  |  |
| --- | --- |
| Y | 10 |

|  |  |
| --- | --- |
| Y | 20 |

|  |  |
| --- | --- |
| Y | 30 |

|  |  |
| --- | --- |
| Y | 40 |

|  |  |
| --- | --- |
| Y | 50 |

|  |  |
| --- | --- |
| Z | 10 |

|  |  |
| --- | --- |
| Z | 20 |

|  |  |
| --- | --- |
| Z | 30 |

|  |  |
| --- | --- |
| Z | 40 |

|  |  |
| --- | --- |
| Z | 50 |

(16-3) Question 7:

**Insert a table on this document showing the result set of:** the cross join of the following two tables:

|  |  |
| --- | --- |
| **Table 1** | **Table 2** |

|  |  |
| --- | --- |
| 2 | NULL |

|  |  |
| --- | --- |
| 2 | 3 |

|  |  |
| --- | --- |
| 2 | 6 |

|  |  |
| --- | --- |
| 2 | 9 |

|  |  |
| --- | --- |
| 2 | 12 |

|  |  |
| --- | --- |
| 4 | NULL |

|  |  |
| --- | --- |
| 4 | 3 |

|  |  |
| --- | --- |
| 4 | 6 |

|  |  |
| --- | --- |
| 4 | 9 |

|  |  |
| --- | --- |
| 4 | 12 |

|  |  |
| --- | --- |
| 6 | NULL |

|  |  |
| --- | --- |
| 6 | 3 |

|  |  |
| --- | --- |
| 6 | 6 |

|  |  |
| --- | --- |
| 6 | 9 |

|  |  |
| --- | --- |
| 6 | 12 |

|  |  |
| --- | --- |
| 8 | NULL |

|  |  |
| --- | --- |
| 8 | 3 |

|  |  |
| --- | --- |
| 8 | 6 |

|  |  |
| --- | --- |
| 8 | 9 |

|  |  |
| --- | --- |
| 8 | 12 |

|  |  |
| --- | --- |
| 10 | NULL |

|  |  |
| --- | --- |
| 10 | 3 |

|  |  |
| --- | --- |
| 10 | 6 |

|  |  |
| --- | --- |
| 10 | 9 |

|  |  |
| --- | --- |
| 10 | 12 |

|  |  |
| --- | --- |
| 12 | NULL |

|  |  |
| --- | --- |
| 12 | 3 |

|  |  |
| --- | --- |
| 12 | 6 |

|  |  |
| --- | --- |
| 12 | 9 |

|  |  |
| --- | --- |
| 12 | 12 |

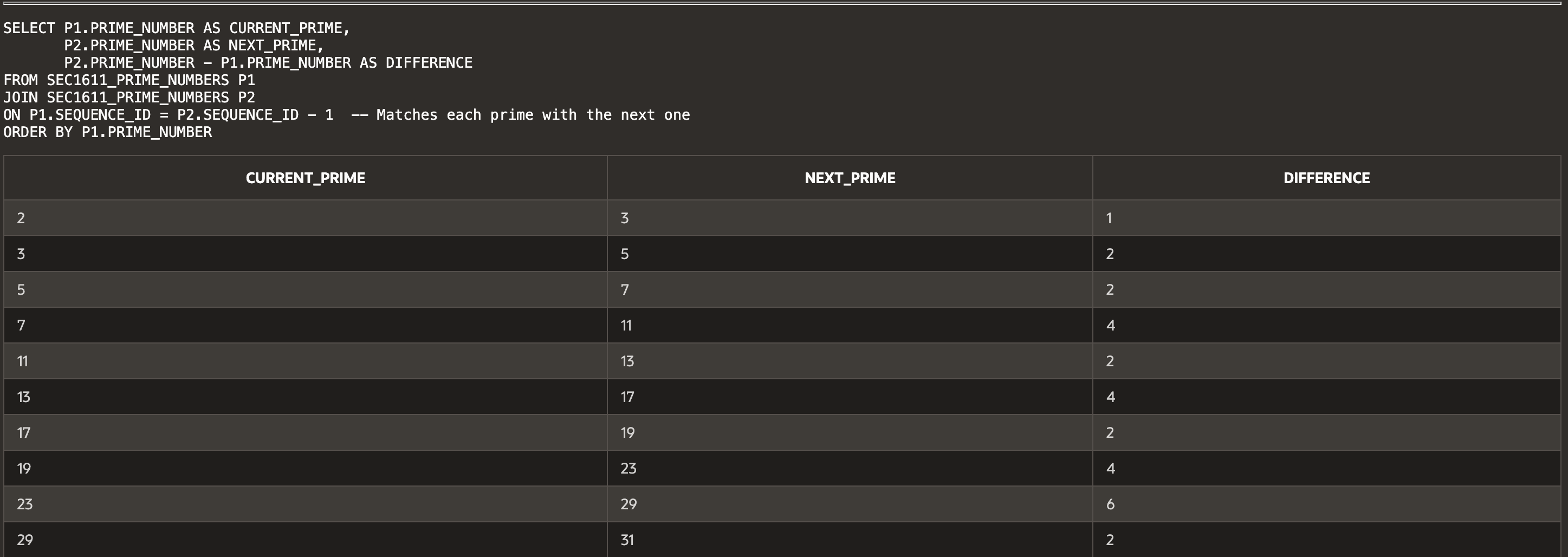
(16-5) Question 8:

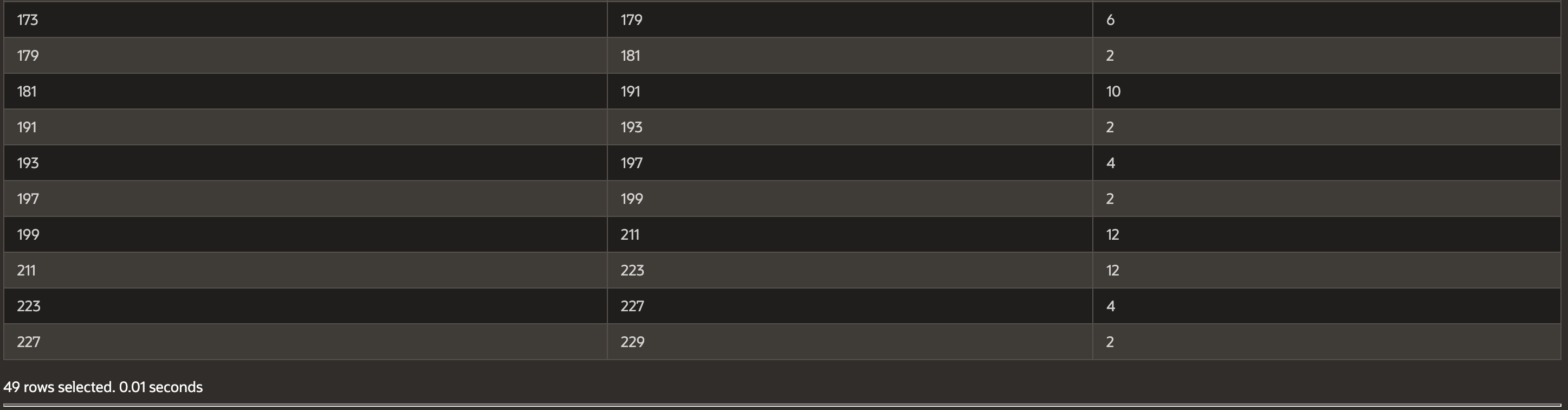
Suppose you are developing a new *select* statement. It is fairly complex and you are using several tables. You are expecting a result with about 400 rows, but the result you get is about 2,000 rows. What part of your *select* statement would you examine first?

If a query returns far more rows than expected, the **first thing to check is the JOIN conditions**. A missing or incorrect join condition can create a **Cartesian join**, multiplying rows instead of matching them correctly. Also, verify that filters (WHERE clauses) are applied correctly and that there are no unintended duplicates in the source tables.

(16-11) Question 9:

Table *sec1611\_prime\_numbers* contains the first 50 prime numbers. Use a self join to determine the difference between each prime and the next one.

...



**Script:**

-- Haley Archer

-- (15-11) Question 1:

-- List all rows and columns of L\_EMPLOYEES.

-- Add a column "Old Timer" for employees hired before 2000.

-- Sort by employee\_id.

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, HIRE\_DATE, DEPT\_CODE, CREDIT\_LIMIT, PHONE\_NUMBER, MANAGER\_ID,

CASE

WHEN EXTRACT(YEAR FROM HIRE\_DATE) < 2000 THEN 'Old Timer'

ELSE NULL

END AS OLD\_TIMER

FROM L\_EMPLOYEES

ORDER BY EMPLOYEE\_ID;

-- (15-12) Question 2:

-- Divide hire date into "Old Timers" (before 2000) and "Newer Hires" (2000+).

-- Sort by last name, then first name.

SELECT LAST\_NAME, FIRST\_NAME,

CASE

WHEN EXTRACT(YEAR FROM HIRE\_DATE) < 2000 THEN HIRE\_DATE

ELSE NULL

END AS OLD\_TIMERS,

CASE

WHEN EXTRACT(YEAR FROM HIRE\_DATE) >= 2000 THEN HIRE\_DATE

ELSE NULL

END AS NEWER\_HIRES

FROM L\_EMPLOYEES

ORDER BY LAST\_NAME, FIRST\_NAME;

-- (15-13) Question 3:

-- Multiply even numbers by 2 and odd numbers by 3 from numbers\_0\_to\_9.

SELECT DIGIT,

CASE

WHEN MOD(DIGIT, 2) = 0 THEN DIGIT \* 2

ELSE DIGIT \* 3

END AS RESULT

FROM NUMBERS\_0\_TO\_9;

-- (15-15) Question 4:

-- Find the intersection of the twos table and the threes table.

SELECT \* FROM TWOS

INTERSECT

SELECT \* FROM THREES;

-- (15-16) Question 5:

-- Find the difference between the twos table and the threes table (both ways).

SELECT \* FROM TWOS

MINUS

SELECT \* FROM THREES;

SELECT \* FROM THREES

MINUS

SELECT \* FROM TWOS;

-- (16-11) Question 9:

-- Find the difference between each prime number and the next in sec1611\_prime\_numbers.

SELECT P1.PRIME\_NUMBER AS CURRENT\_PRIME,

P2.PRIME\_NUMBER AS NEXT\_PRIME,

P2.PRIME\_NUMBER - P1.PRIME\_NUMBER AS DIFFERENCE

FROM SEC1611\_PRIME\_NUMBERS P1

JOIN SEC1611\_PRIME\_NUMBERS P2

ON P1.SEQUENCE\_ID = P2.SEQUENCE\_ID - 1

ORDER BY P1.PRIME\_NUMBER;