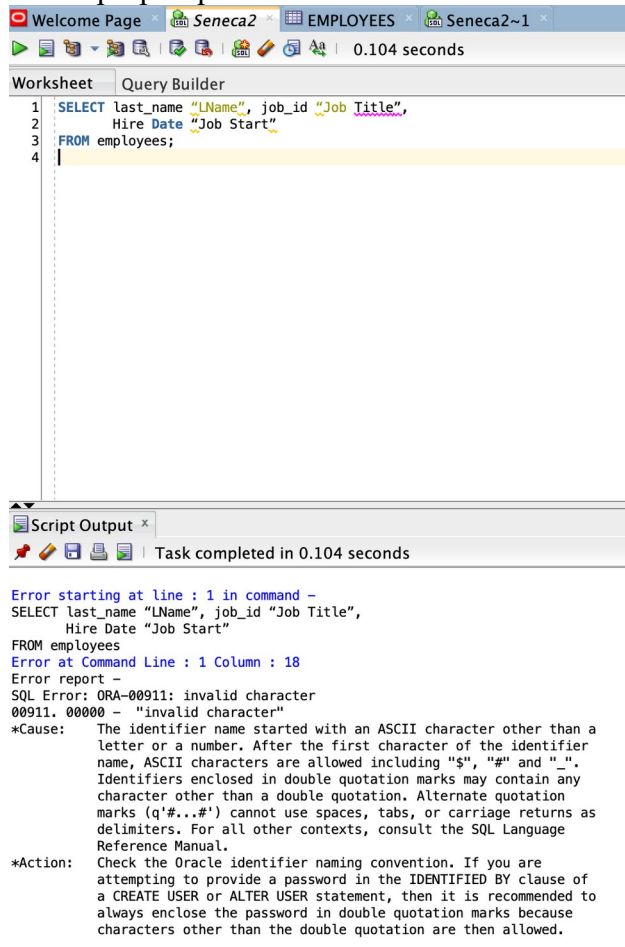


1. False/ The error occurred because there is an invalid character in the SQL statement. Specifically, the column alias "Hire Date" contains a space, which is not allowed without proper quotation.



The screenshot shows the SQL Developer interface. The top pane, titled 'Query Builder', contains the following SQL query:

```
1 SELECT last_name "LName", job_id "Job Title",  
2       Hire Date "Job Start",  
3 FROM employees;  
4
```

The bottom pane, titled 'Script Output', displays the error message:

```
Error starting at line : 1 in command -  
SELECT last_name "LName", job_id "Job Title",  
       Hire Date "Job Start",  
FROM employees  
Error at Command Line : 1 Column : 18  
Error report -  
SQL Error: ORA-00911: invalid character  
00911. 00000 - "invalid character"  
*Cause:   The identifier name started with an ASCII character other than a  
          letter or a number. After the first character of the identifier  
          name, ASCII characters are allowed including "$", "#", and ".".  
          Identifiers enclosed in double quotation marks may contain any  
          character other than a double quotation. Alternate quotation  
          marks (q'...'') cannot use spaces, tabs, or carriage returns as  
          delimiters. For all other contexts, consult the SQL Language  
          Reference Manual.  
*Action:  Check the Oracle identifier naming convention. If you are  
          attempting to provide a password in the IDENTIFIED BY clause of  
          a CREATE USER or ALTER USER statement, then it is recommended to  
          always enclose the password in double quotation marks because  
          characters other than the double quotation are then allowed.
```

2.

The errors include missing underscores in column aliases, and there's a comma at the end of the SELECT list without a subsequent column.

3.

SELECT location_id AS "City#", city, state_province || ' ' || country_id || ' ' || 'IN THE' AS
 "Province with Country Code"
 FROM locations;

Worksheet Query Builder

```
1 SELECT location_id AS "City#", city, state_province || ' ' || country_id || ' ' || 'IN THE' AS "Province with Country Code"
2 FROM locations;
3
```

Script Output

Task completed in 0.06 seconds

| City# | CITY | Province with Country Code |
|-------|-------------|-----------------------------|
| 1900 | Whitehorse | Yukon CA IN THE |
| 2000 | Beijing | CN IN THE |
| 2100 | Bombay | Maharashtra IN IN THE |
| 2200 | Sydney | New South Wales AU IN THE |
| 2300 | Singapore | SG IN THE |
| 2400 | London | UK IN THE |
| 2500 | Oxford | Oxford UK IN THE |
| 2600 | Stretford | Manchester UK IN THE |
| 2700 | Munich | Bavaria DE IN THE |
| 2800 | Sao Paulo | Sao Paulo BR IN THE |
| 2900 | Geneva | Geneve CH IN THE |
| 3000 | Bern | BE CH IN THE |
| 3100 | Utrecht | Utrecht NL IN THE |
| 3200 | Mexico City | Distrito Federal, MX IN THE |

23 rows selected.

4. Select distinct department id from employees;

Worksheet Query Builder

```
1 Select distinct department_id from employees;
```

Script Output

Query Result

SQL | All Rows Fetched: 8 in 0.051 seconds

| | DEPARTMENT_ID |
|---|---------------|
| 1 | (null) |
| 2 | 90 |
| 3 | 20 |
| 4 | 110 |
| 5 | 50 |
| 6 | 80 |
| 7 | 60 |
| 8 | 10 |

5. Select employee_id, last_name, salary from employees where salary between 8000 and 15000 order by salary desc, last_name;

The screenshot shows the Oracle SQL Developer interface. The left pane displays the 'Oracle Connections' tree with 'Seneca2' selected. The 'Tables (Filtered)' list includes COUNTRIES, DEPARTMENTS, EMPLOYEES, EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, PHONE_NUMBER, HIRE_DATE, JOB_ID, SALARY, COMMISSION_PCT, MANAGER_ID, DEPARTMENT_ID, JOB_GRADES, JOB_HISTORY, JOBS, LOCATIONS, PRODUCT_CATEGORIES, PRODUCTS, and WAREHOUSES. The main pane shows a query in the 'Query Builder' tab:

```
1 Select employee_id, last_name, salary from employees where salary between 8000 and 15000 order by salary desc, last_name;
```

The 'Script Output' pane shows the query result:

| EMPLOYEE_ID | LAST_NAME | SALARY |
|-------------|---------------|--------|
| 1 | 201 Hartstein | 13000 |
| 2 | 205 Higgins | 12000 |
| 3 | 174 Abel | 11000 |
| 4 | 149 Zlotkey | 10500 |
| 5 | 103 Hunold | 9000 |
| 6 | 176 Taylor | 8600 |
| 7 | 206 Gietz | 8300 |

6. Select employee_id, last_name, salary, job_id from employee where (salary between 8000 and 15000) and (job_id = 'SA_REP' or job_id = 'IT_PROG') order by salary desc, last_name;

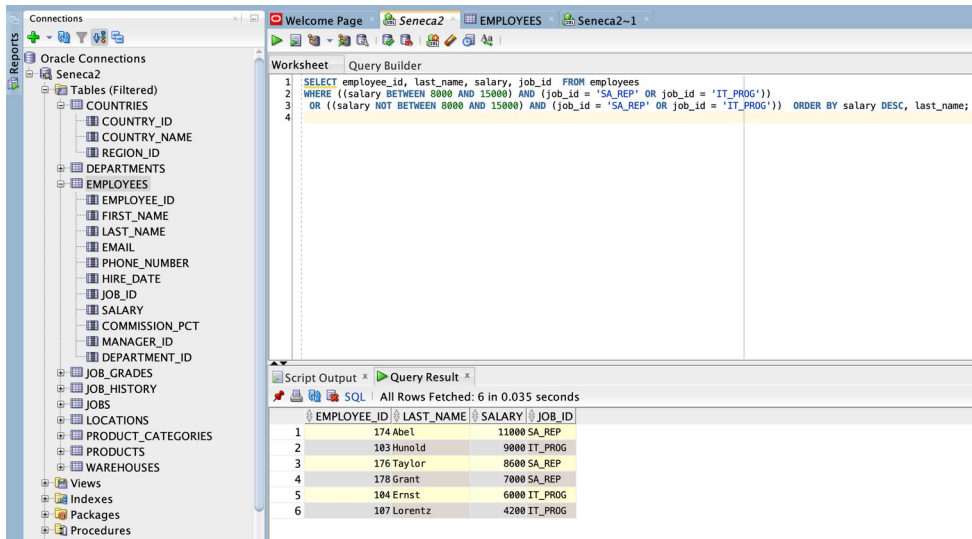
The screenshot shows the Oracle SQL Developer interface. The left pane displays the 'Oracle Connections' tree with 'Seneca2' selected. The 'Tables (Filtered)' list includes COUNTRIES, DEPARTMENTS, EMPLOYEES, EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, PHONE_NUMBER, HIRE_DATE, JOB_ID, SALARY, COMMISSION_PCT, MANAGER_ID, DEPARTMENT_ID, JOB_GRADES, JOB_HISTORY, JOBS, LOCATIONS, PRODUCT_CATEGORIES, PRODUCTS, and WAREHOUSES. The main pane shows a query in the 'Query Builder' tab:

```
1 SELECT employee_id, last_name, salary, job_id
2 FROM employees
3 WHERE (salary BETWEEN 8000 AND 15000)
4 AND (job_id = 'SA_REP' OR job_id = 'IT_PROG')
5 ORDER BY salary DESC, last_name;
6
```

The 'Script Output' pane shows the query result:

| EMPLOYEE_ID | LAST_NAME | SALARY | JOB_ID |
|-------------|------------|--------|---------|
| 1 | 174 Abel | 11000 | SA_REP |
| 2 | 103 Hunold | 9000 | IT_PROG |
| 3 | 176 Taylor | 8600 | SA_REP |

7. SELECT employee_id, last_name, salary, job_id FROM employees
 WHERE ((salary BETWEEN 8000 AND 15000) AND (job_id = 'SA_REP' OR job_id = 'IT_PROG'))
 OR ((salary NOT BETWEEN 8000 AND 15000) AND (job_id = 'SA_REP' OR job_id = 'IT_PROG')) ORDER BY salary DESC, last_name;



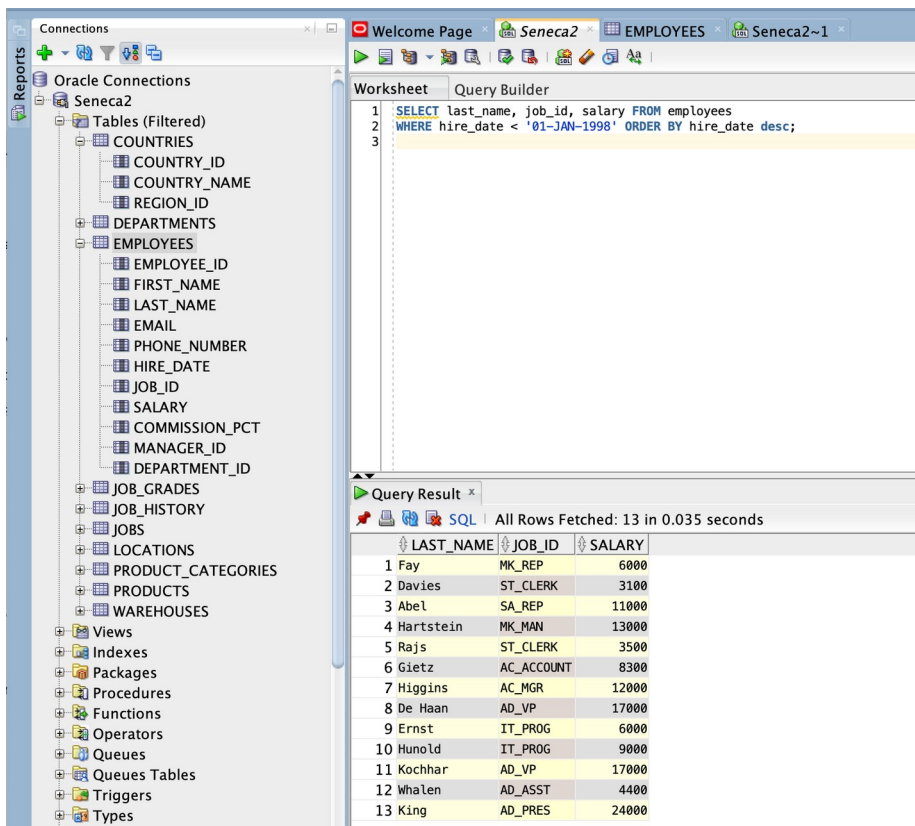
The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane displays 'Seneca2' as the selected connection. The 'Reports' pane shows a tree view of the database schema, including tables like COUNTRIES, DEPARTMENTS, EMPLOYEES, and JOBS. The 'Query Builder' pane shows the following SQL query:

```
1 SELECT employee_id, last_name, salary, job_id FROM employees
2 WHERE ((salary BETWEEN 8000 AND 15000) AND (job_id = 'SA_REP' OR job_id = 'IT_PROG'))
3 OR ((salary NOT BETWEEN 8000 AND 15000) AND (job_id = 'SA_REP' OR job_id = 'IT_PROG')) ORDER BY salary DESC, last_name;
```

The 'Query Result' pane shows the results of the query, with 6 rows fetched in 0.035 seconds. The results are as follows:

| EMPLOYEE_ID | LAST_NAME | SALARY | JOB_ID |
|-------------|-----------|--------|---------|
| 174 | Abel | 11000 | SA_REP |
| 183 | Hunold | 9000 | IT_PROG |
| 176 | Taylor | 8600 | SA_REP |
| 178 | Grant | 7000 | SA_REP |
| 184 | Ernst | 6000 | IT_PROG |
| 187 | Lorentz | 4200 | IT_PROG |

8. SELECT last_name, job_id, salary FROM employees
 WHERE hire_date < '01-JAN-1998' ORDER BY hire_date desc;



The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane displays 'Seneca2' as the selected connection. The 'Reports' pane shows a tree view of the database schema, including tables like COUNTRIES, DEPARTMENTS, EMPLOYEES, and JOBS. The 'Query Builder' pane shows the following SQL query:

```
1 SELECT last_name, job_id, salary FROM employees
2 WHERE hire_date < '01-JAN-1998' ORDER BY hire_date desc;
```

The 'Query Result' pane shows the results of the query, with 13 rows fetched in 0.035 seconds. The results are as follows:

| LAST_NAME | JOB_ID | SALARY |
|-----------|------------|--------|
| Fay | MK_REP | 6000 |
| Davies | ST_CLERK | 3100 |
| Abel | SA_REP | 11000 |
| Hartstein | MK_MAN | 13000 |
| Rajs | ST_CLERK | 3500 |
| Gietz | AC_ACCOUNT | 8300 |
| Higgins | AC_MGR | 12000 |
| De Haan | AD_VP | 17000 |
| Ernst | IT_PROG | 6000 |
| Hunold | IT_PROG | 9000 |
| Kochhar | AD_VP | 17000 |
| Whalen | AD_ASST | 4400 |
| King | AD PRES | 24000 |

9. SELECT last_name, job_id, salary FROM employees
 WHERE salary > 10000
 ORDER BY job_id, salary desc;

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane displays the 'Seneca2' database structure, including tables like COUNTRIES, DEPARTMENTS, EMPLOYEES, and JOBS. The 'EMPLOYEES' table is selected. The main window shows the 'Query Builder' tab with the following SQL query:

```
1 SELECT last_name, job_id, salary FROM employees
2 WHERE salary > 10000
3 ORDER BY job_id, salary DESC;
```

The 'Query Result' pane shows the results of the query, with all rows fetched in 0.036 seconds. The results are displayed in a table with columns LAST_NAME, JOB_ID, and SALARY.

| | LAST_NAME | JOB_ID | SALARY |
|---|-----------|---------|--------|
| 1 | Higgins | AC_MGR | 12000 |
| 2 | King | AD_PRES | 24000 |
| 3 | Kochhar | AD_VP | 17000 |
| 4 | De Haan | AD_VP | 17000 |
| 5 | Hartstein | MK_MAN | 13000 |
| 6 | Zlotkey | SA_MAN | 10500 |
| 7 | Abel | SA_REP | 11000 |

10.
 SELECT job_id, last_name, first_name FROM employees
 WHERE first_name LIKE '%c%' OR first_name LIKE '%E%';

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane displays the 'Seneca2' database structure, including tables like COUNTRIES, DEPARTMENTS, EMPLOYEES, and JOBS. The 'EMPLOYEES' table is selected. The main window shows the 'Query Builder' tab with the following SQL query:

```
1 SELECT job_id, last_name, first_name FROM employees
2 WHERE first_name LIKE 'c%' OR first_name LIKE 'E%';
3
4
```

The 'Query Result' pane shows the results of the query, with all rows fetched in 0.033 seconds. The results are displayed in a table with columns JOB_ID, LAST_NAME, and FIRST_NAME.

| | JOB_ID | LAST_NAME | FIRST_NAME |
|---|---------|-----------|------------|
| 1 | IT_PROG | Ernst | Bruce |
| 2 | MK_MAN | Hartstein | Michael |
| 3 | SA_MAN | Zlotkey | Eleni |
| 4 | SA_REP | Abel | Ellen |