**1.List all rows from L\_FOODS where price is less than $1.00 or greater than $5.00**

SELECT \*   
FROM L\_FOODS   
WHERE price < 1.00 OR price > 5.00

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SUPPLIER\_ID** | **PRODUCT\_CODE** | **MENU\_ITEM** | **DESCRIPTION** | **PRICE** | **PRICE\_INCREASE** |
| CBC | GS | 4 | GRILLED STEAK | 6 | .7 |
| JBR | VR | 9 | COFFEE | .85 | .15 |

**2. List all foods that do NOT have a null in price\_increase column**  
SELECT \*   
FROM L\_FOODS   
WHERE price\_increase IS NOT NULL

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SUPPLIER\_ID** | **PRODUCT\_CODE** | **MENU\_ITEM** | **DESCRIPTION** | **PRICE** | **PRICE\_INCREASE** |
| ASP | FS | 1 | FRESH SALAD | 2 | .25 |
| ASP | SW | 3 | SANDWICH | 3.5 | .4 |
| CBC | GS | 4 | GRILLED STEAK | 6 | .7 |
| CBC | SW | 5 | HAMBURGER | 2.5 | .3 |
| FRV | BR | 6 | BROCCOLI | 1 | .05 |
| JBR | AS | 8 | SODA | 1.25 | .25 |
| JBR | VR | 9 | COFFEE | .85 | .15 |
| VSB | AS | 10 | DESSERT | 3 | .5 |

**3. Put the NOT clause into standard form**  
SELECT \*  
FROM L\_EMPLOYEES  
WHERE NOT (first\_name = 'JIM' OR first\_name = 'DAN')  
 OR NOT (last\_name = 'BROWN' OR last\_name = 'SMITH');

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **FIRST\_NAME** | **LAST\_NAME** | **DEPT\_CODE** | **HIRE\_DATE** | **CREDIT\_LIMIT** | **PHONE\_NUMBER** | **MANAGER\_ID** |
| 201 | SUSAN | BROWN | EXE | 6/1/1998 | 30 | 3484 | - |
| 202 | JIM | KERN | SAL | 8/16/1999 | 25 | 8722 | 201 |
| 203 | MARTHA | WOODS | SHP | 2/2/2009 | 25 | 7591 | 201 |
| 204 | ELLEN | OWENS | SAL | 7/1/2008 | 15 | 6830 | 202 |
| 205 | HENRY | PERKINS | SAL | 3/1/2006 | 25 | 5286 | 202 |
| 206 | CAROL | ROSE | ACT | - | - | - | - |
| 208 | FRED | CAMPBELL | SHP | 4/1/2008 | 25 | 1752 | 203 |
| 209 | PAULA | JACOBS | MKT | 3/17/1999 | 15 | 3357 | 201 |
| 210 | NANCY | HOFFMAN | SAL | 2/16/2007 | 25 | 2974 | 203 |

**4. Add parentheses to prevent logical errors in WHERE clause**  
SELECT \*   
FROM L\_FOODS   
WHERE (description = 'FRESH SALAD'   
 OR description = 'SANDWICH'   
 OR description = 'DESSERT')   
 AND price <= 2.50   
 AND price\_increase <= 0.25

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SUPPLIER\_ID** | **PRODUCT\_CODE** | **MENU\_ITEM** | **DESCRIPTION** | **PRICE** | **PRICE\_INCREASE** |
| ASP | FS | 1 | FRESH SALAD | 2 | .25 |

**5. Using a table of constants instead of hard-coded values**  
SELECT description, min\_price, max\_price   
FROM L\_FOODS, sec0306\_price\_constants   
WHERE price BETWEEN min\_price AND max\_price   
ORDER BY description

|  |  |  |
| --- | --- | --- |
| **DESCRIPTION** | **MIN\_PRICE** | **MAX\_PRICE** |
| BROCCOLI | 1 | 2 |
| FRENCH FRIES | 1 | 2 |
| FRESH SALAD | 1 | 2 |
| SODA | 1 | 2 |
| SOUP OF THE DAY | 1 | 2 |

5 rows selected. 0.00 seconds

**6. Correcting syntax error caused by single quote in string**

SELECT \*   
FROM L\_SUPPLIERS   
WHERE supplier\_name = UPPER('frank reed''s vegetables')

|  |  |
| --- | --- |
| **SUPPLIER\_ID** | **SUPPLIER\_NAME** |
| FRV | FRANK REED'S VEGETABLES |

1 row(s) selected. 0.01 seconds

**7. Making search case-insensitive using UPPER()**  
SELECT \*   
FROM L\_EMPLOYEES   
WHERE UPPER(last\_name) = UPPER('de haan')

no data found

8. **Introducing an intentional error and checking error message**

SELECT last\_name, first\_name, phone\_number   
FROM L\_EMPLOYEES   
WHERE job\_code == 'MGR'; -- Intentional syntax error: '==' should be '='

Error at line 3/17: ORA-00936: missing expression