|  |  |  |
| --- | --- | --- |
| SELECT DML | | |
| SELECT Clause | | |
| **Basic SELECT Query** | SELECT columnlist  FROM tablelist; | SELECT P\_CODE, P\_DESCRIPT, P\_PRICE, P\_QOH  FROM PRODUCT; |
| **Using Column Aliases** |  | SELECT P\_CODE,  DESCRIPT AS DESCRIPTION,  P\_PRICE AS “Unit Price”,  P\_QOH QTY  FROM PRODUCT; |
| **Using Computed Columns** |  | SELECT P\_DESCRIPT, P\_QOH, P\_PRICE, P\_QOH \* P\_PRICE AS TOTVALUE  FROM PRODUCT; |
| **Date Arithmetic** |  | SELECT P\_CODE, P\_INDATE, P\_INDATE + 90 AS EXPDATE  FROM PRODUCT; |
| **Listing Unique Values** |  | SELECT DISTINCT V\_CODE  FROM PRODUCT; |
| FROM Clause | | |
| **Natural Join** | SELECT column-list  FROM table1 NATURAL JOIN table2 | SELECT CUS\_CODE, CUS\_LNAME, INV\_NUMBER, INV\_DATE  FROM CUSTOMER NATURAL JOIN INVOICE; |
| **JOIN USING** | SELECT column-list  FROM table1 JOIN table2  USING (common column) | SELECT P\_CODE, P\_DESCRIPT, V\_CODE, V\_NAME, V\_AREACODE, V\_PHONE  FROM PRODUCT JOIN VENDOR USING (V\_CODE); |
| **JOIN ON** | SELECT *column-list*  FROM *table1* JOIN *table2*  ON *join-condition* | SELECT INVOICE.INV\_NUMBER, PRODUCT.P\_CODE,P\_DESCRIPT, LINE\_UNITS,  LINE\_PRICE  FROM INVOICE JOIN LINE ON INVOICE.INV\_NUMBER = LINE.INV\_NUMBER  JOIN PRODUCT ON LINE.P\_CODE = PRODUCT.P\_CODE; |
| **Left Outer Joins** | SELECT column-list  FROM *table1* LEFT [OUTER] JOIN *table2*  ON *join-condition* | SELECT P\_CODE, VENDOR.V\_CODE, V\_NAME  FROM VENDOR LEFT JOIN PRODUCT  ON VENDOR. V\_CODE = PRODUCT.V\_CODE; |
| **Right Outer Joins** | SELECT column-list  FROM *table1* RIGHT [OUTER] JOIN *table2*  ON *join-condition* | SELECT P\_CODE, VENDOR.V\_CODE, V\_NAME  FROM VENDOR RIGHT JOIN PRODUCT  ON VENDOR. V\_CODE = PRODUCT.V\_CODE; |
| **Full Outer Joins** | SELECT column-list  FROM *table1* FULL [OUTER] JOIN *table2*  ON *join-condition* | SELECT P\_CODE, VENDOR.V\_CODE, V\_NAME  FROM VENDOR FULL JOIN PRODUCT  ON VENDOR. V\_CODE = PRODUCT.V\_CODE; |
| **Cross Join** | SELECT *column-list*  FROM *table1* CROSS JOIN *table2* | SELECT INVOICE.INV\_NUMBER, CUS\_CODE, INV\_DATE, P\_CODE  FROM INVOICE CROSS JOIN LINE; |
| **Joining Tables with an Alias** |  | SELECT P\_DESCRIPT, P\_PRICE, V\_NAME, P.V\_CODE, V\_CONTACT  FROM PRODUCT P JOIN VENDOR V ON P.V\_CODE = V.V\_CODE; |
| **Recursive Joins** |  | SELECT E.EMP\_NUM, E.EMP\_LNAME, E.EMP\_MGR, M.EMP\_LNAME  FROM EMP E JOIN EMP M ON E.EMP\_MGR = M.EMP\_NUM; |
| WHERE Clause | | |
| **Selecting Rows with Conditional Restrictions** | SELECT *columnlist*  FROM *tablelist*  [WHERE *conditionlist* ]  [ORDER BY *columnlist* [ASC | DESC] ]; | SELECT P\_DESCRIPT, P\_QOH, P\_PRICE, V\_CODE  FROM PRODUCT  WHERE V\_CODE = 21344; |
| **Old-Style Joins** |  | SELECT P\_CODE, P\_DESCRIPT, P\_PRICE, V\_NAME  FROM PRODUCT, VENDOR  WHERE PRODUCT.V\_CODE = VENDOR.V\_CODE; |
| GROUP BY Clause | | |
| **Grouping Data** | SELECT *columnlist*  FROM *tablelist*  [WHERE *conditionlist* ]  [GROUP BY *columnlist* ]  [ORDER BY *columnlist* [ASC | DESC] ]; | SELECT V\_CODE, V\_NAME, COUNT(P\_CODE) AS NUMPROD  FROM PRODUCT JOIN VENDOR  ON PRODUCT.V\_CODE = VENDOR.V\_CODE  GROUP BY V\_CODE, V\_NAME  ORDER BY V\_NAME; |
| HAVING Clause | | |
|  | SELECT *columnlist*  FROM *tablelist*  [WHERE *conditionlist* ]  [GROUP BY *columnlist* ]  [HAVING *conditionlist* ]  [ORDER BY *columnlist* [ASC | DESC] ]; | SELECT V\_CODE, V\_NAME, SUM(P\_QOH \* P\_PRICE) AS TOTCOST  FROM PRODUCT JOIN VENDOR  ON PRODUCT.V\_CODE = VENDOR.V\_CODE  WHERE P\_DISCOUNT > 0  GROUP BY V\_CODE, V\_NAME  HAVING (SUM(P\_QOH \* P\_PRICE) > 500)  ORDER BY SUM(P\_QOH \* P\_PRICE) DESC; |
| ORDER BY Clause | | |
| **Ascending** | SELECT *columnlist*  FROM *tablelist*  [ORDER BY *columnlist* [ASC | DESC] ]; | SELECT P\_CODE, P\_DESCRIPT, P\_QOH, P\_PRICE  FROM PRODUCT  ORDER BY P\_PRICE; |
| **Descending** |  | SELECT P\_CODE, P\_DESCRIPT, P\_QOH, P\_PRICE  FROM PRODUCT  ORDER BY P\_PRICE DESC; |
| **Both with Alias** |  | SELECT P\_CODE, P\_DESCRIPT, V\_CODE, P\_PRICE \* P\_QOH AS TOTAL  FROM PRODUCT  ORDER BY V\_CODE, TOTAL DESC; |
| Subqueries | | |
| **Select Subquery** |  | SELECT P\_CODE,  SUM(LINE\_UNITS \* LINE\_PRICE) AS SALES,  (SELECT COUNT(\*) FROM EMPLOYEE) AS ECOUNT,  SUM(LINE\_UNITS \* LINE\_PRICE)/(SELECT COUNT(\*)  FROM EMPLOYEE) AS CONTRIB  FROM LINE  GROUP BY P\_CODE; |
| **From Subqueries** |  | SELECT DISTINCT CUSTOMER.CUS\_CODE, CUSTOMER.CUS\_LNAME  FROM CUSTOMER JOIN (SELECT INVOICE.CUS\_CODE  FROM INVOICE JOIN LINE  ON INVOICE.INV\_NUMBER = LINE.INV\_NUMBER  WHERE P\_CODE = '13-Q2/P2') CP1  ON CUSTOMER.CUST\_CODE = CP1.CUS\_CODE |
| **Where Comparison Subqueries** |  | SELECT P\_CODE, P\_PRICE  FROM PRODUCT  WHERE P\_PRICE >= (SELECT AVG(P\_PRICE) FROM PRODUCT); |
| **Where IN Subqueries** |  | SELECT P\_CODE, P\_PRICE  FROM PRODUCT  WHERE P\_CODE IN (SELECT DISTINCT P\_CODE FROM LINE); |
| **Having Subqueries** |  | SELECT P\_CODE, SUM(LINE\_UNITS) AS TOTALUNITS  FROM LINE  GROUP BY P\_CODE  HAVING SUM(LINE\_UNITS) > (SELECT AVG(LINE\_UNITS) FROM LINE); |
| **Other DML** | | |
| INSERT | | |
| **Adding Table Rows** | INSERT INTO *tablename*  VALUES (*value1*, *value2*, …, *valuen*) | INSERT INTO VENDOR  VALUES (21225',Bryson, Inc.','Smithson','615','223-3234','TN','Y'); |
| UPDATE | | |
| **Updating Table Rows** | UPDATE *tablename*  SET *columnname* = *expression*  [, *columnname* = *expression*]  [WHERE *conditionlist* ]; | UPDATE PRODUCT  SET P\_INDATE = '18-JAN-2018', P\_PRICE = 17.99, P\_MIN = 10  WHERE P\_CODE = '13-Q2/P2'; |
| DELETE | | |
| **Deleting T able Rows** | DELETE FROM *tablename*  [WHERE *conditionlist* ]; | DELETE FROM PRODUCT  WHERE P\_CODE = 'BRT-345'; |
| **DDL** | | |
| CREATE | | |
| **Creating Table** | CREATE TABLE *tablename* (  *column1 data type* [*constraint*] [,  *column2 data type* [*constraint*] ] [,  PRIMARY KEY (*column1* [, *column2*]) ] [,  FOREIGN KEY (*column1* [, *column2*]) REFERENCES *tablename*] [,  CONSTRAINT *constraint* ] ); | CREATE TABLE PRODUCT (  P\_CODE VARCHAR(10) NOT NULL UNIQUE,  P\_DESCRIPT VARCHAR(35) NOT NULL,  P\_INDATE DATE NOT NULL,  P\_QOH SMALLINT NOT NULL,  P\_MIN SMALLINT NOT NULL,  P\_PRICE NUMBER(8,2) NOT NULL,  P\_DISCOUNT NUMBER(5,2) NOT NULL,  V\_CODE INTEGER,  PRIMARY KEY (P\_CODE),  FOREIGN KEY (V\_CODE) REFERENCES VENDOR (V\_CODE)); |
| ALTER | | |
| **Adding or Modifing column** | ALTER TABLE *tablename*  {ADD | MODIFY} ( *columnname datatype* [ {ADD | MODIFY} *columnname datatype*] ); | ALTER TABLE PRODUCT  MODIFY (V\_CODE CHAR(5));  ALTER TABLE PRODUCT  ADD (P\_SALECODE CHAR(1)); |
| **Adding Constraint** | ALTER TABLE *tablename*  ADD *constraint* [ ADD *constraint* ]; | ALTER TABLE PART  ADD PRIMARY KEY (PART\_CODE); |
| **Deleting a column** | ALTER TABLE *tablename*  DROP {PRIMARY KEY | COLUMN *columnname* |  CONSTRAINT *constraintname* }; | ALTER TABLE VENDOR  DROP COLUMN V\_ORDER; |
| DELETE | | |
| **Deleting Table** | DROP TABLE *tablename*; | DROP TABLE PART; |
| **Transaction DDL** | | |
| **Saving Table Changes** | COMMIT [WORK] | COMMIT; |
| **Restoring Table Contents** |  | ROLLBACK; |



