



# **COSC 3380**

## **Design of Database Systems**

Basic Structured Query Language (SQL)

February 28, 2024

# Basic SQL Queries:

## SELECT-FROM-WHERE Structure

<b>SELECT</b>	<attribute list>
<b>FROM</b>	<table list>
<b>WHERE</b>	<condition>;

where

- <attribute list> is a list of attribute names whose values are to be retrieved by the query.
- <table list> is a list of the relation names required to process the query.
- <condition> is a conditional (Boolean) expression that identifies the tuples to be retrieved by the query.

# Basic SQL Queries:

## SELECT-FROM-WHERE Structure

- Logical comparison operators  
=, <, <=, >, >=, and <>
- **Projection attributes**
  - Attributes whose values are to be retrieved
- **Selection condition**
  - Boolean condition that must be true for any retrieved tuple
  - Selection conditions include **join** conditions when multiple relations are involved



# Product Supplier Database

**SUPPLIER**(SUPNR, SUPNAME, SUPADDRESS, SUPCITY, SUPSTATUS)

**PRODUCT**(PRODNR, PRODNAME, PRODTYPE, AVAILABLE\_QUANTITY)

**SUPPLIES**(SUPNR, PRODNR, PURCHASE\_PRICE, DELIV\_PERIOD)

**PURCHASE\_ORDER**(PONR, PODATE, *SUPNR*)

**PO\_LINE**(PONR, PRODNR, QUANTITY)

# Product Supplier Database

```
CREATE TABLE SUPPLIER
(SUPNR CHAR(4) NOT NULL PRIMARY KEY,
SUPNAME VARCHAR(40) NOT NULL,
SUPADDRESS VARCHAR(50),
SUPCITY VARCHAR(20),
SUPSTATUS SMALLINT);

CREATE TABLE PRODUCT
(PRODNR CHAR(6) NOT NULL PRIMARY KEY,
PRODNAME VARCHAR(60) NOT NULL,
CONSTRAINT UC1 UNIQUE(PRODNAME),
PRODTYPE VARCHAR(10),
CONSTRAINT CC1 CHECK(PRODTYPE IN ('white', 'red',
'rose', 'sparkling')),
AVAILABLE_QUANTITY INTEGER);
```

# Product Supplier Database

```
CREATE TABLE SUPPLIES  
(SUPNR CHAR(4) NOT NULL,  
PRODNR CHAR(6) NOT NULL,  
PURCHASE_PRICE DOUBLE(8,2)  
COMMENT 'PURCHASE_PRICE IN EUR',  
DELIV_PERIOD TIME  
COMMENT 'DELIV_PERIOD IN DAYS',  
PRIMARY KEY (SUPNR, PRODNR),  
FOREIGN KEY (SUPNR) REFERENCES SUPPLIER (SUPNR)  
ON DELETE CASCADE ON UPDATE CASCADE,  
FOREIGN KEY (PRODNR) REFERENCES PRODUCT (PRODNR)  
ON DELETE CASCADE ON UPDATE CASCADE);
```



# Product Supplier Database

```
CREATE TABLE PURCHASE_ORDER  
(PONR CHAR(7) NOT NULL PRIMARY KEY,  
PODATE DATE,  
SUPNR CHAR(4) NOT NULL,  
FOREIGN KEY (SUPNR) REFERENCES SUPPLIER (SUPNR)  
ON DELETE CASCADE ON UPDATE CASCADE);
```

```
CREATE TABLE PO_LINE  
(PONR CHAR(7) NOT NULL,  
PRODNR CHAR(6) NOT NULL,  
QUANTITY INTEGER,  
PRIMARY KEY (PONR, PRODNR),  
FOREIGN KEY (PONR) REFERENCES PURCHASE_ORDER (PONR)  
ON DELETE CASCADE ON UPDATE CASCADE,  
FOREIGN KEY (PRODNR) REFERENCES PRODUCT (PRODNR)  
ON DELETE CASCADE ON UPDATE CASCADE);
```

# Simple Queries

- SQL statements that retrieve data from only one table

**Q1: SELECT SUPNR, SUPNAME, SUPADDRESS,  
SUPCITY, SUPSTATUS FROM SUPPLIER;**

**Q1: SELECT \* FROM SUPPLIER;**



# Simple Queries

**Q1: SELECT SUPNR, SUPNAME, SUPADDRESS, SUPCITY, SUPSTATUS  
FROM SUPPLIER;**

SUPNR	SUPNAME	SUPADDRESS	SUPCITY	SUPSTATUS
21	Deliwines	240, Avenue of the Americas	New York	20
32	Best Wines	660, Market Street	San Francisco	90
37	Ad Fundum	82, Wacker Drive	Chicago	95
52	Spirits & co.	928, Strip	Las Vegas	NULL
68	The Wine Depot	132, Montgomery Street	San Francisco	10
69	Vinos del Mundo	4, Collins Avenue	Miami	92

# Simple Queries

**Q2: SELECT SUPNR, SUPNAME FROM SUPPLIER;**

SUPNR	SUPNAME	SUPADDRESS	SUPCITY	SUPSTATUS
21	Deliwines	240, Avenue of the Americas	New York	20
32	Best Wines	660, Market Street	San Francisco	90
37	Ad Fundum	82, Wacker Drive	Chicago	95
52	Spirits & co.	928, Strip	Las Vegas	NULL
68	The Wine Depot	132, Montgomery Street	San Francisco	10
69	Vinos del Mundo	4, Collins Avenue	Miami	92

SUPNR	SUPNAME
21	Deliwines
32	Best Wines
37	Ad Fundum
52	Spirits & co.
68	The Wine Depot
69	Vinos del Mundo

# Simple Queries

**Q3: SELECT SUPNR  
FROM PURCHASE\_ORDER;**

SUPNR
32
32
37
37
37
37
37
68
69
94



# Simple Queries – using DISTINCT keyword

**Q4: SELECT DISTINCT SUPNR  
FROM PURCHASE\_ORDER;**

SUPNR
32
37
68
69
94

SUPNR
32
32
37
37
37
37
37
68
69
94

# Simple Queries – create a new data item

**Q5: SELECT SUPNR, PRODNR, DELIV\_PERIOD/30 AS MONTH\_DELIV\_PERIOD FROM SUPPLIES;**

SUPNR	PRODNR	MONTH_DELIV_PERIOD
21	0119	0.0333
21	0178	NULL
21	0289	0.0333
21	0327	0.2000
21	0347	0.0667
21	0384	0.0667
...	...	...

# Simple Queries – WHERE clause

**Q6: SELECT SUPNR, SUPNAME, SUPSTATUS FROM SUPPLIER  
WHERE SUPCITY = 'San Francisco';**

SUPNR	SUPNAME	SUPSTATUS
32	Best Wines	90
68	The Wine Depot	10



# Simple Queries - WHERE clause

**Q7: SELECT SUPNR, SUPNAME, SUPSTATUS  
FROM SUPPLIER  
WHERE SUPCITY = 'San Francisco' AND  
SUPSTATUS > 80;**

SUPNR	SUPNAME	SUPSTATUS
32	Best Wines	90

# Simple Queries – BETWEEN keyword

**Q8: SELECT SUPNR, SUPNAME, SUPSTATUS  
FROM SUPPLIER  
WHERE SUPSTATUS BETWEEN 70 AND 80;**

SUPNR	SUPNAME	SUPSTATUS
94	The Wine Crate	75

# Simple Queries – IN keyword

```
Q9: SELECT PRODNR, PRODNAME  
      FROM PRODUCT  
      WHERE PRODTYPE IN ('WHITE', 'SPARKLING');
```

PRODNR	PRODNAME
0178	Meerdael, Methode Traditionnelle Chardonnay, 2014
0199	Jacques Selsosse, Brut Initial, 2012
0212	Billecart-Salmon, Brut Réserve, 2014
0300	Chateau des Rontets, Chardonnay, Birbettes
0494	Veuve-Cliquot, Brut, 2012
0632	Meneghetti, Chardonnay, 2010
...	....



# Simple Queries – LIKE keyword

**Q10: SELECT PRODNR, PRODNAME  
FROM PRODUCT  
WHERE PRODNAME LIKE '%CHARD%';**

PRODNR	PRODNAME
0300	Chateau des Rontets, Chardonnay, Birbettes
0783	Clos D'Opleeuw, Chardonnay, 2012
0178	Meerdael, Methode Traditionnelle Chardonnay, 2014
0632	Meneghetti, Chardonnay, 2010

# Simple Queries – IS NULL check

**Q11: SELECT SUPNR, SUPNAME, SUPSTATUS  
FROM SUPPLIER  
WHERE SUPSTATUS IS NULL;**

SUPNR	SUPNAME	SUPADDRESS	SUPCITY	SUPSTATUS
21	Deliwines	240, Avenue of the Americas	New York	20
32	Best Wines	660, Market Street	San Francisco	90
37	Ad Fundum	82, Wacker Drive	Chicago	95
52	Spirits & co.	928, Strip	Las Vegas	NULL
68	The Wine Depot	132, Montgomery Street	San Francisco	10
69	Vinos del Mundo	4, Collins Avenue	Miami	92

SUPNR	SUPNAME	SUPSTATUS
52	Spirits & Co.	NULL

## EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

## DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

## DEPT\_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston



# Simple Queries

**Query 0.** Retrieve the birth date and address of the employee(s) whose name is 'John B. Smith'.

**Q0:**      **SELECT**      Bdate, Address  
             **FROM**        EMPLOYEE  
             **WHERE**      Fname='John' AND Minit='B' AND Lname='Smith';

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

<u>Bdate</u>	<u>Address</u>
1965-01-09	731Fondren, Houston, TX

# Simple Queries

**Query 1.** Retrieve the name and address of all employees who work for the 'Research' department.

**Q1:**     **SELECT**     Fname, Lname, Address  
             **FROM**       EMPLOYEE, DEPARTMENT  
             **WHERE**     Dname='Research' **AND** Dnumber=Dno;

<u>Fname</u>	<u>Lname</u>	<u>Address</u>
John	Smith	731 Fondren, Houston, TX
Franklin	Wong	638 Voss, Houston, TX
Ramesh	Narayan	975 Fire Oak, Humble, TX
Joyce	English	5631 Rice, Houston, TX

**WORKS\_ON**

<u>Essn</u>	<u>Pno</u>	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

**PROJECT**

Pname	<u>Pnumber</u>	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

**DEPENDENT**

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse



# Basic Retrieval Queries

**Query 2.** For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.

**Q2:**      **SELECT**      Pnumber, Dnum, Lname, Address, Bdate  
             **FROM**        PROJECT, DEPARTMENT, EMPLOYEE  
             **WHERE**      Dnum=Dnumber **AND** Mgr\_ssn=Ssn **AND**  
                         Plocation='Stafford';

<u>Pnumber</u>	<u>Dnum</u>	<u>Lname</u>	<u>Address</u>	<u>Bdate</u>
10	4	Wallace	291 Berry, Bellaire, TX	1941-06-20
30	4	Wallace	291 Berry, Bellaire, TX	1941-06-20