ISE 211 - Industrial Information Systems

Databases and Analysis

Lecture 5 - Chapter 2

SQL

Relational Database Model

İzmir University of Economics

Halil POSACI

2011, İzmir

Agenda

- SQL
  - Creating DB & Tables
  - Managing The Data In The Database Table
    - INSERT
    - SELECT
    - UPDATE
    - DELETE
  - Converting DATA into INFORMATION
    - Aggregate Functions
    - Grouping
    - Sub queries
    - Appending Tables using Joins

- HOMEWORK
- Lab – 2.4 ACME

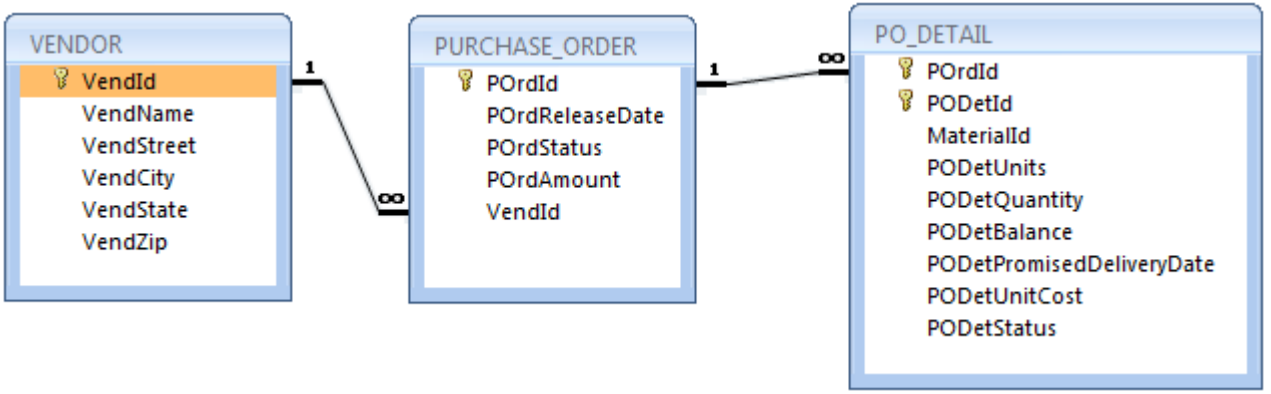
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

INSERT INTO <table name>
([<attribute1 name>], <[attribute2 name]>,...)
VALUES (<value1>, <value2>,...);

Probably will not work. Why?

```
INSERT INTO PURCHASE_ORDER
VALUES (2591, "02/10/06", "CLOSED", 4300.00, "V110");
```



Managing The Data In database Table

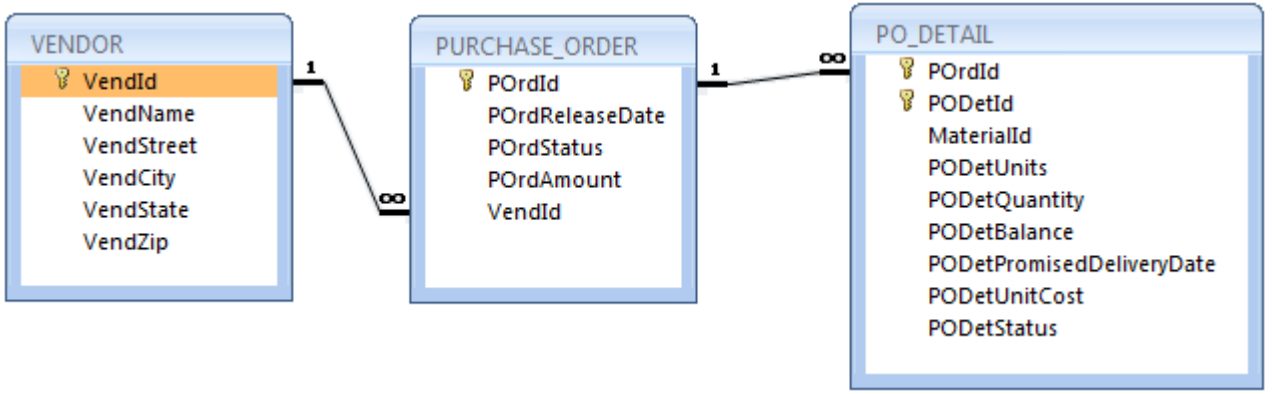
INSERT – SELECT – UPDATE – DELETE

SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;

```
SELECT *
FROM PURCHASE_ORDER;
```

POrdId	POrdStatus	POrdAmour
2591	CLOSED	$4,300.00
2592	OPEN	$505.50
2593	OPEN	$4,000.00
2594	OPEN	$3,280.00
2595	OPEN	$500.00
2596	HOLD	$1,000.00

```
SELECT POrdId, POrdStatus, POrdAmount
FROM PURCHASE_ORDER;
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

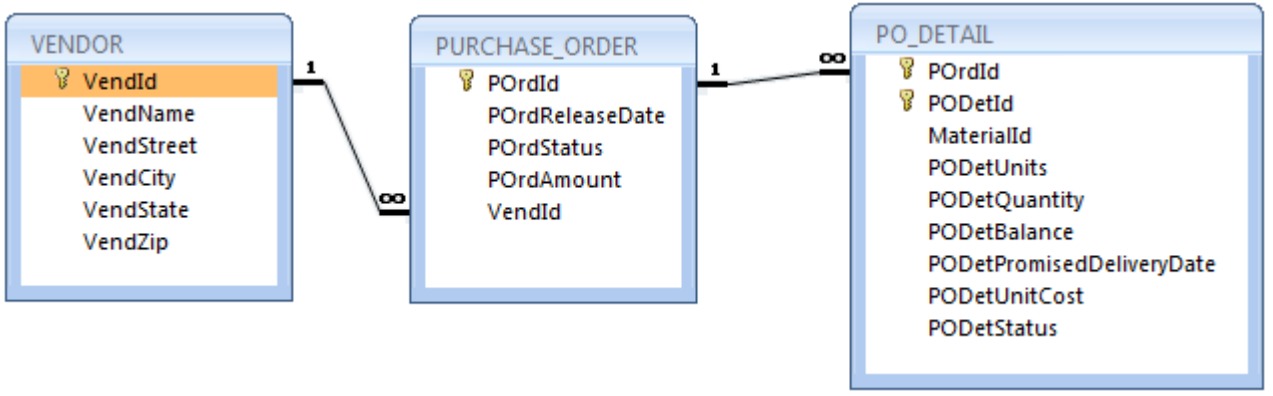
```
SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;
```

Default

POrdId	POrdStatus	POrdAmour
2595	OPEN	$500.00
2592	OPEN	$505.50
2596	HOLD	$1,000.00
2594	OPEN	$3,280.00
2593	OPEN	$4,000.00
2591	CLOSED	$4,300.00

Query2

```
SELECT POrdId, POrdStatus, POrdAmount
FROM PURCHASE_ORDER
ORDER BY POrdAmount;
```



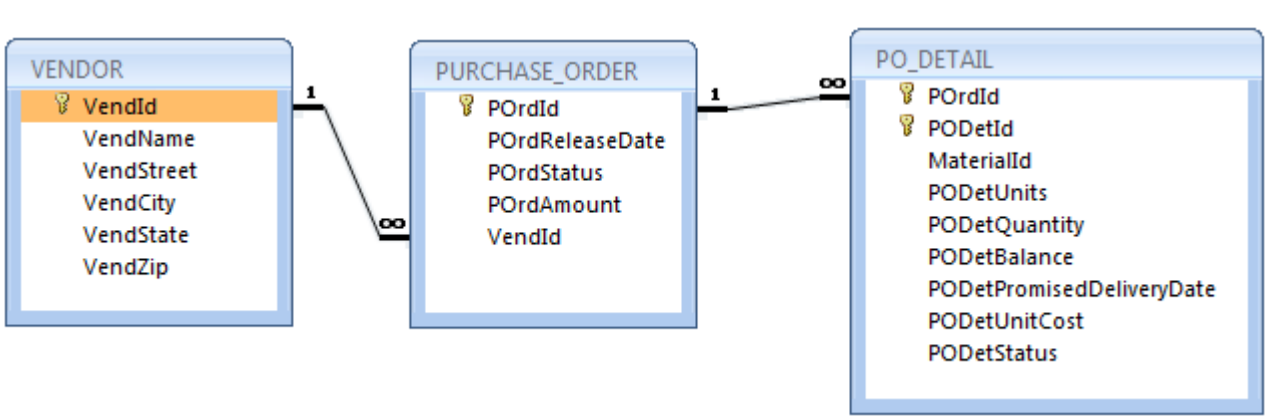
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;

VendId	POrdAmour	POrdId	POrdStatus
V110	$4,000.00	2593	OPEN
V110	$4,300.00	2591	CLOSED
V25	$505.50	2592	OPEN
V250	$500.00	2595	OPEN
V250	$3,280.00	2594	OPEN
V75	$1,000.00	2596	HOLD

```
SELECT PURCHASE_ORDER.VendId, PURCHASE_ORDER.POrdAmount,
PURCHASE_ORDER.POrdId, PURCHASE_ORDER.POrdStatus
FROM PURCHASE_ORDER
ORDER BY PURCHASE_ORDER.VendId, PURCHASE_ORDER.POrdAmount;
```



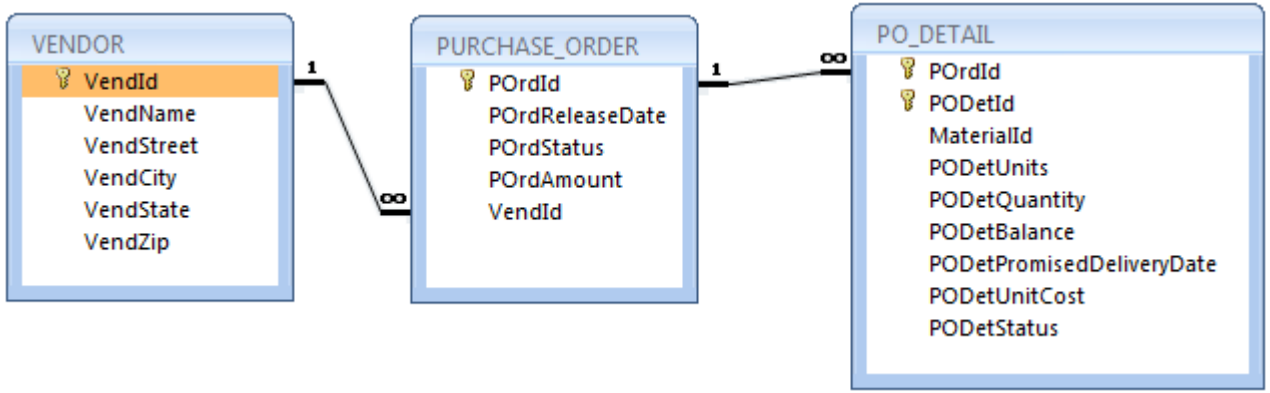
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;

VendId
V110
V25
V250
V75

```
SELECT DISTINCT VendId
FROM PURCHASE_ORDER
```



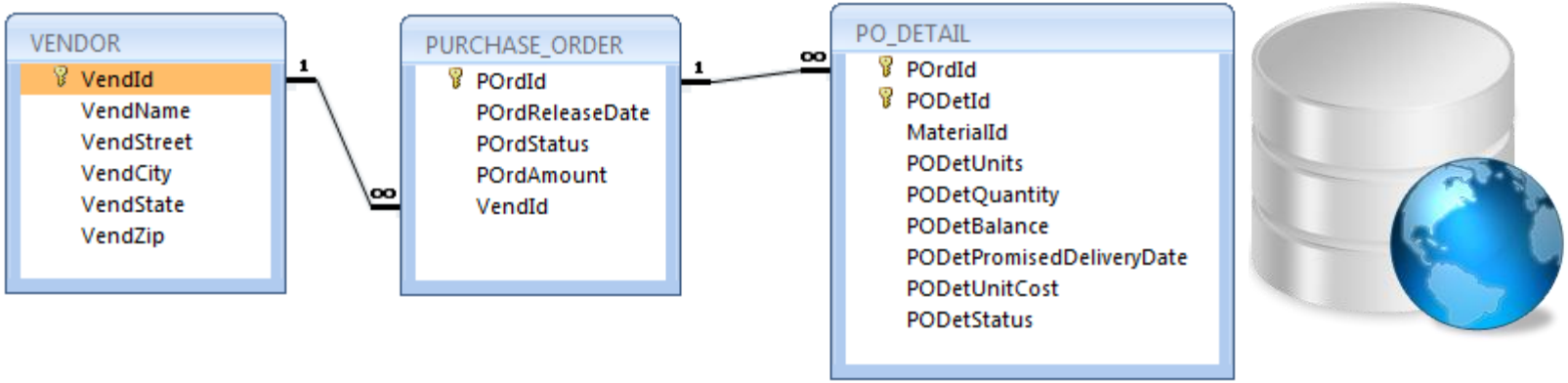
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2591	2/10/2001	CLOSED	$4,300.00	V110
2593	2/11/2001	OPEN	$4,000.00	V110

```
SELECT *
FROM PURCHASE_ORDER
WHERE VendId="V110";
```



Managing The Data In database Table

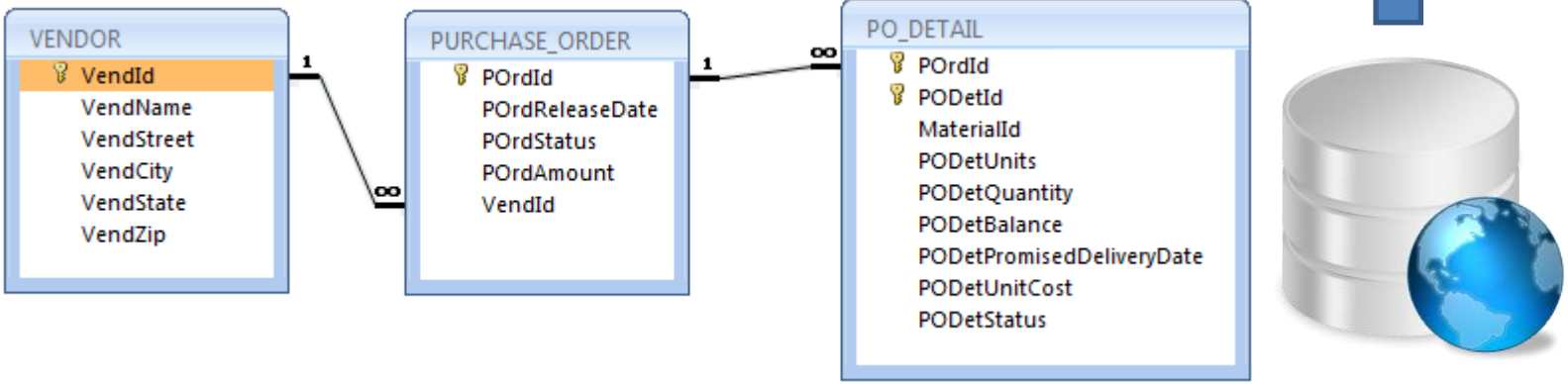
INSERT – SELECT – UPDATE – DELETE

```
SELECT [DISTINCT] <attributes/*>
FROM <table name>
WHERE <condition>
ORDER BY <attribute name> ASC/DESC;
```

VendId	POrdId	POrdAmour
V250	2594	$3,280.00
V110	2593	$4,000.00
V110	2591	$4,300.00

```
 = equal to
!= not equal to
<  less than
<= less than or equal to
>  greater than
>= greater than or equal to
```

```
SELECT VendId,POrdId, POrdAmount
FROM PURCHASE_ORDER
WHERE POrdAmount> 1000
ORDER BY POrdAmount;
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

LOGICAL OPERATORS

AND

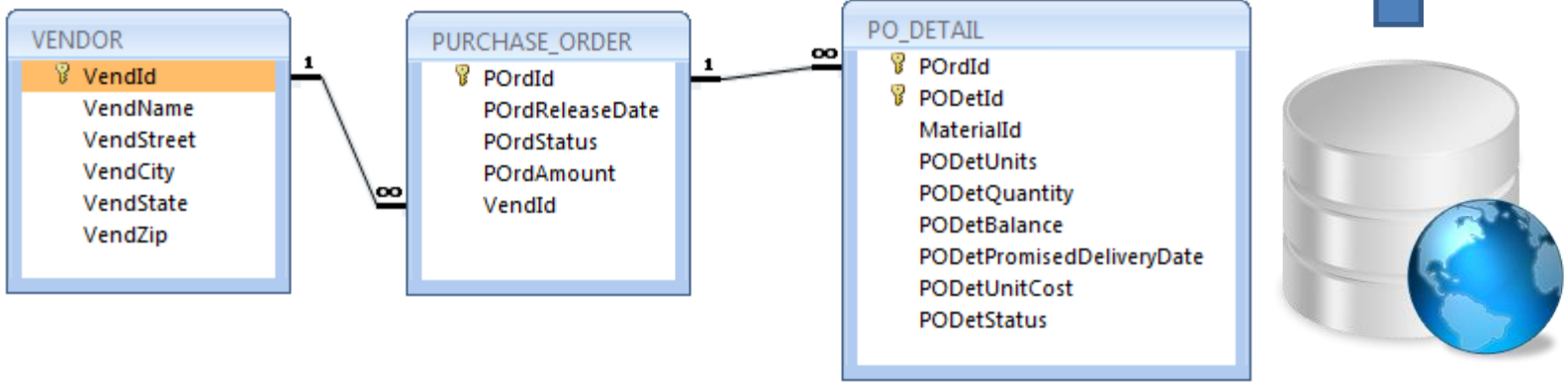
OR

NOT

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2592	2/10/2001	OPEN	$505.50	V25
2593	2/11/2001	OPEN	$4,000.00	V110
2594	2/12/2001	OPEN	$3,280.00	V250

= equal to
!= not equal to
< less than
<= less than or equal to
> greater than
>= greater than or equal to

```
SELECT *
FROM PURCHASE_ORDER
WHERE POrdStatus ="OPEN"
AND POrdAmount> 500
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

LOGICAL OPERATORS

AND

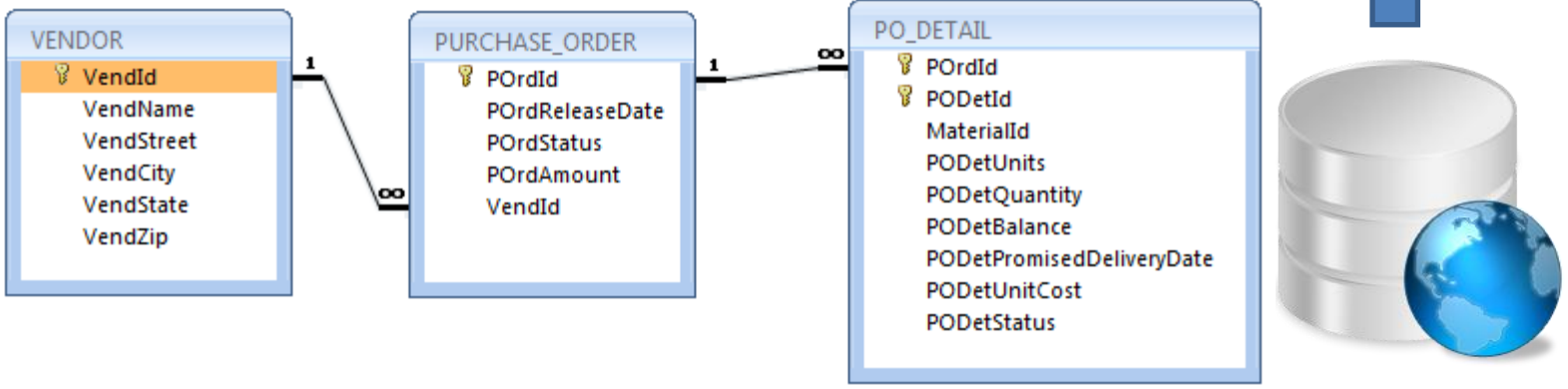
OR

NOT

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2591	2/10/2001	CLOSED	$4,300.00	V110
2596		HOLD	$1,000.00	V75

= equal to
!= not equal to
< less than
<= less than or equal to
> greater than
>= greater than or equal to

```
SELECT *
FROM PURCHASE_ORDER
WHERE NOT POrdStatus="OPEN";
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

LOGICAL OPERATORS

AND

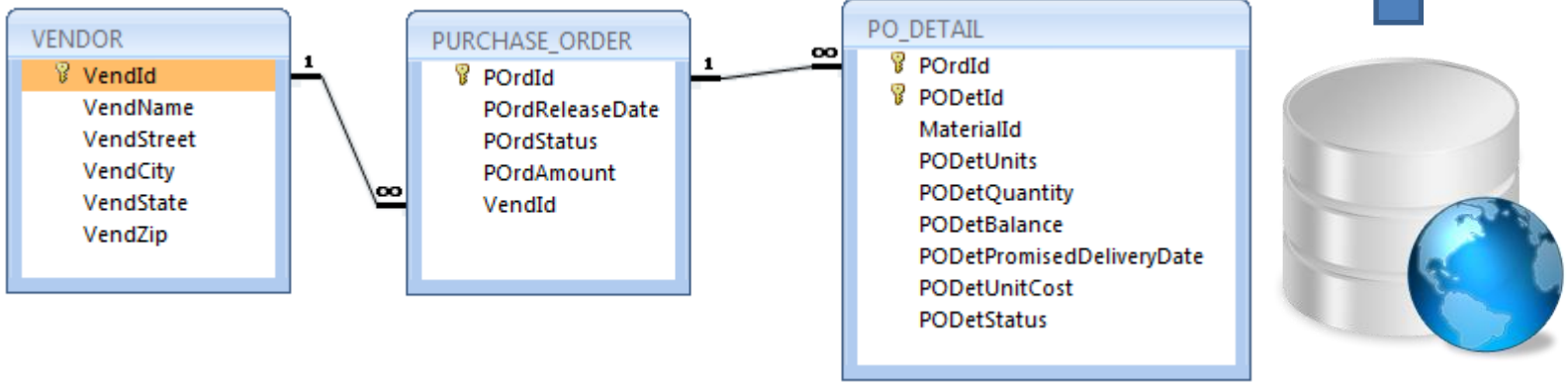
OR

NOT

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2592	2/10/2001	OPEN	$505.50	V25
2594	2/12/2001	OPEN	$3,280.00	V250

= equal to
!= not equal to
< less than
<= less than or equal to
> greater than
>= greater than or equal to

```
SELECT *
FROM PURCHASE_ORDER
WHERE POrdStatus="Open"
AND (VendId>"V150" AND POrdAmount>500)
ORDER BY POrdAmount;
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

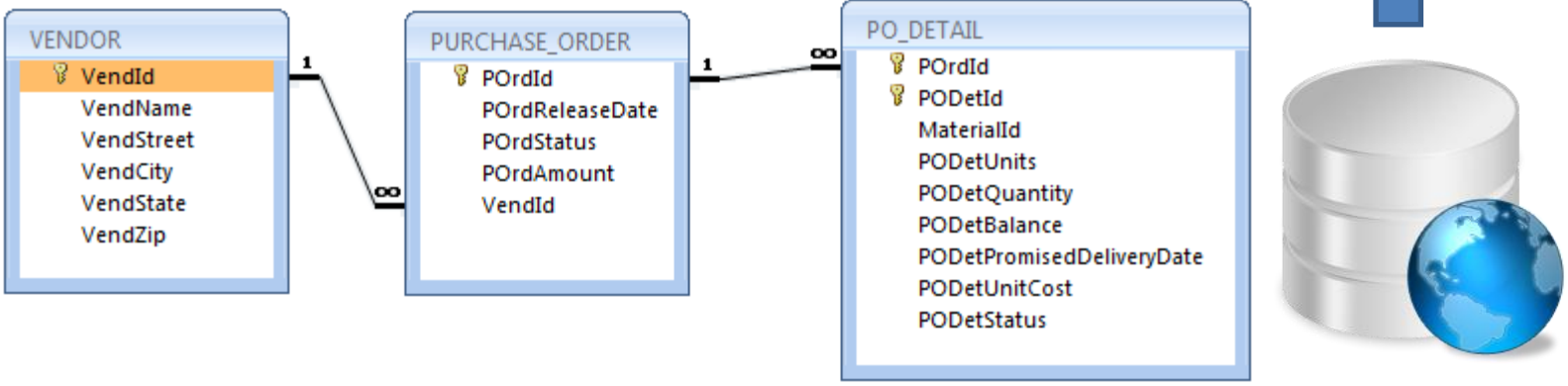
NOT BETWEEN

BETWEEN KEYWORD

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2593	2/11/2011	OPEN	$4,000.00	V110
2594	2/12/2011	OPEN	$3,280.00	V250

= equal to
!= not equal to
< less than
<= less than or equal to
> greater than
>= greater than or equal to

```
SELECT *
FROM PURCHASE_ORDER
WHERE ((POrdAmount BETWEEN 505 AND 4000)
AND (POrdReleaseDate BETWEEN #2/11/11# AND #2/13/11#));
```



Managing The Data In database Table

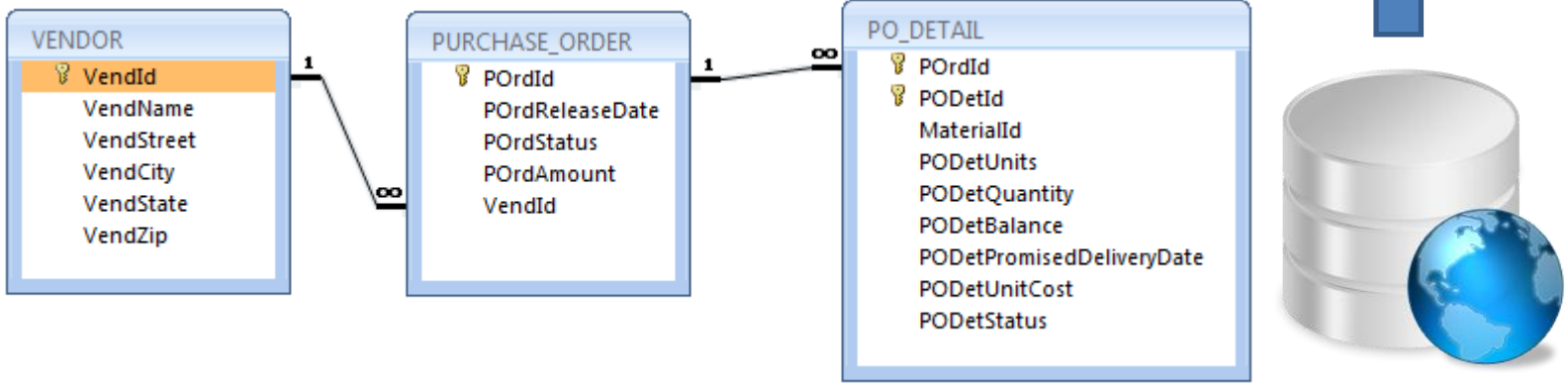
INSERT – SELECT – UPDATE – DELETE

LIKE
NOT LIKE KEYWORD

VenfId	VendName	VendStreet	VendCity	VendState	VendZip
V250	Spices Unlimited	25 Salty Lane	East Hampton	NY	10027

= equal to
!= not equal to
< less than
<= less than or equal to
> greater than
>= greater than or equal to

```
SELECT *
FROM VENDOR
WHERE VendName LIKE "Spice*";
```



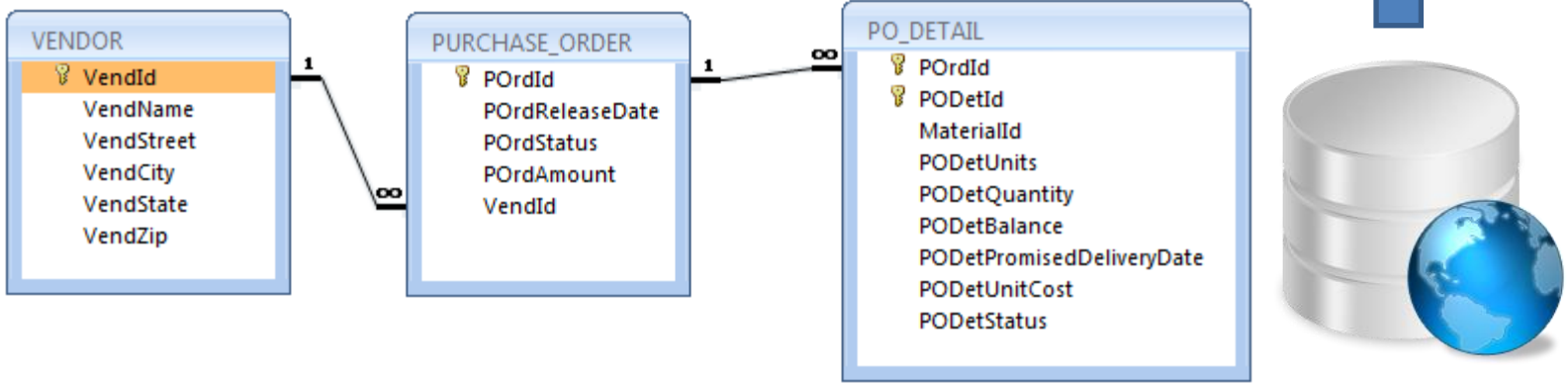
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

NULL KEYWORD

POrdId	POrdReleaseD	POrdStatus	POrdAmour	VendId
2596		HOLD	$1,000.00	V75

```
SELECT *
FROM PURCHASE_ORDER
WHERE POrdReleaseDate IS NULL;
```



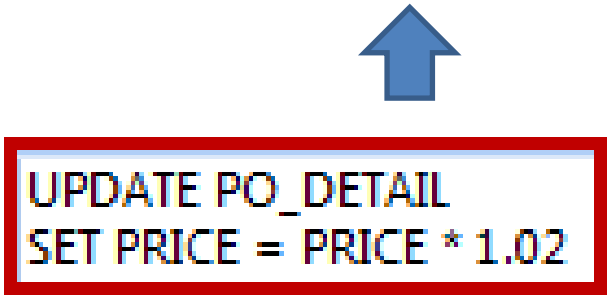
Managing The Data In database Table

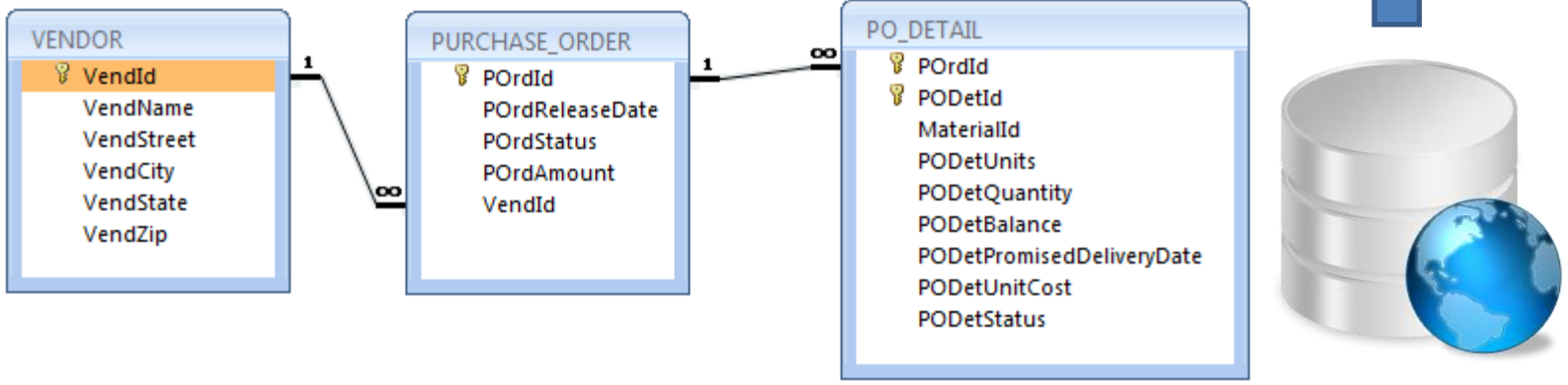
INSERT – SELECT – UPDATE – DELETE

UPDATE <table name>
SET <attribute name> = <value/expression> [...]
[WHERE <condition>];

Will not work

WHY ?





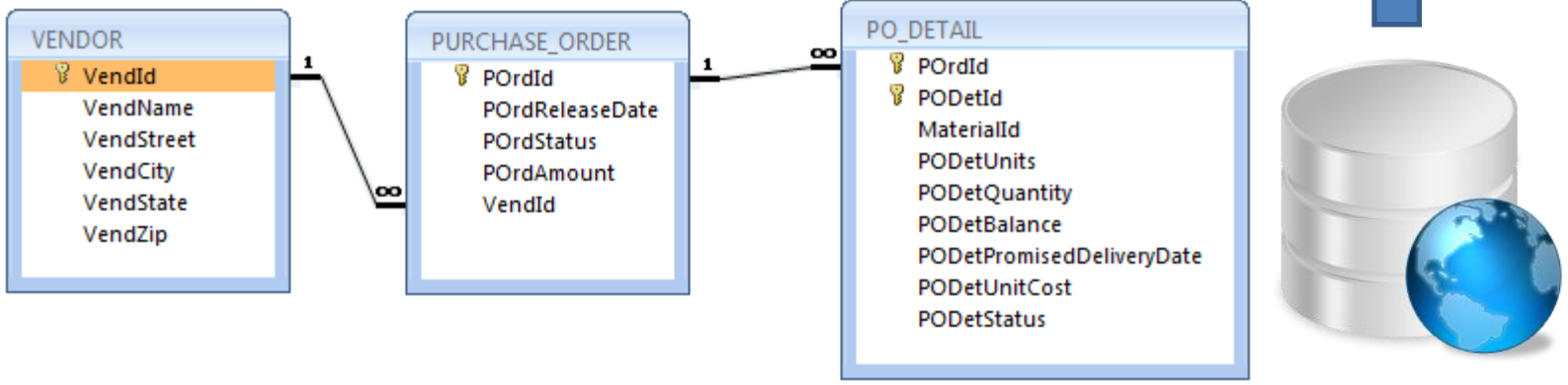
Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

UPDATE <table name>
SET <attribute name> = <value/expression> [...]
[WHERE <condition>];

???

```
UPDATE PO_DETAIL
SET       PODetPromisedDeliveryDate = "03/22/06"
WHERE POrdId=2394 ;
```



Managing The Data In database Table

INSERT – SELECT – UPDATE – DELETE

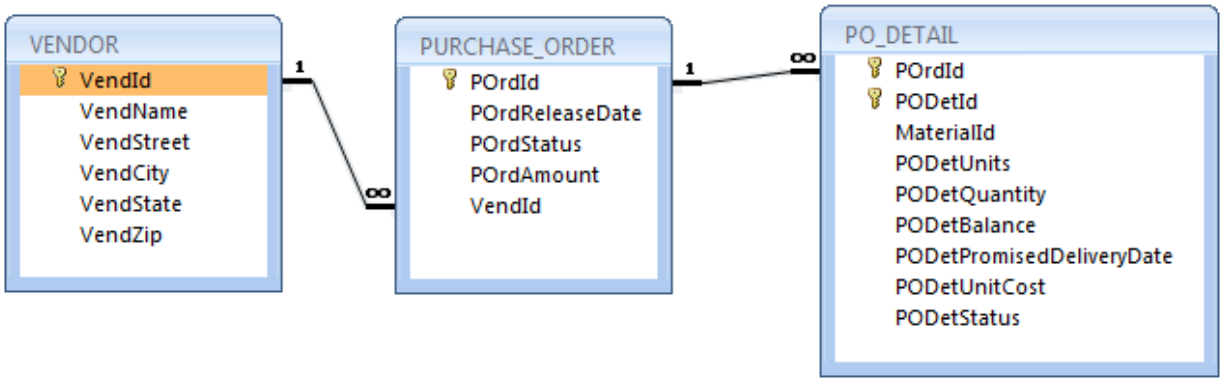
DELETE FROM <table name>
WHERE <condition>;

???

Start from last side to delete
Otherwise Referential Integrity prevents deleting.

```
DELETE FROM PO_DETAIL
WHERE POrdId = 2596;

DELETE FROM PURCHASE_ORDER
WHERE POrdId = 2596;
```



Converting Data Into Information

Forms and Reports

Converting Data Into Information

Aggregate Functions in SQL

```
SELECT AGGREGATE FUNCTION ([DISTINCT] <attribute name>)
FROM <table name>
WHERE <condition>
GROUP BY <attribute name> [HAVING <condition>];
```

Converting Data Into Information

Aggregate Functions in SQL

Expr1000
$2,071.38

```
SELECT AVG(POrdAmount)
FROM PURCHASE_ORDER
WHERE POrdStatus="Open";
```

Converting Data Into Information

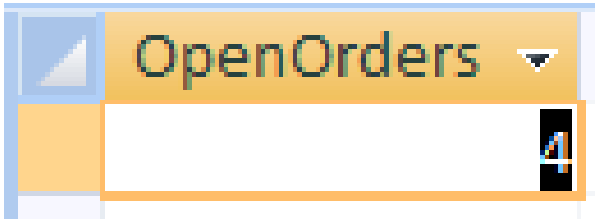
Aggregate Functions in SQL

MinimumAmount
$500.00

```
SELECT MIN(POrdAmount) AS MinimumAmount
FROM PURCHASE_ORDER;
```

Converting Data Into Information

Aggregate Functions in SQL



```
SELECT COUNT(POrdStatus) AS OpenOrders
FROM PURCHASE_ORDER
WHERE POrdStatus ="Open";
```

Converting Data Into Information

Aggregate Functions in SQL

Minimum	Maximum	Expr1002
$500.00	$4,000.00	$8,285.50

```
SELECT MIN(POrdAmount) AS Minimum, MAX(POrdAmount) AS Maximum, SUM(POrdAmount)
FROM PURCHASE_ORDER
WHERE POrdStatus = "OPEN";
```

Converting Data Into Information

Aggregate Functions in SQL

SumInclFreight
4158

```
SELECT SUM(POrdAmount) * 1.1 AS SumInclFreight
FROM PURCHASE_ORDER
WHERE VendId = "V250"
AND (POrdId=2594 OR POrdId=2595);
```

Converting Data Into Information

Grouping Data

POrdId	Expr1001	Expr1002
2591	300	1000
2592	210.5	800.5
2593	1000	2000
2594	560	4000
2595	400	1200
2596	5000	5000

```
SELECT POrdId, Min(PODetQuantity) , Max(PODetQuantity)
FROM PO_DETAIL
GROUP BY PO_DETAIL.POrdId;
```

Converting Data Into Information

GROUPBY - HAVING

POrdId	Expr1001	Expr1002
2593	1000	2000
2594	560	4000
2595	400	1200
2596	5000	5000

```
SELECT POrdId, Min(PODetQuantity) , Max(PODetQuantity)
FROM PO_DETAIL
GROUP BY PO_DETAIL.POrdId HAVING MAX(PODetQuantity)>1000;
```

Converting Data Into Information

Subqueries in SQL

```
MAIN QUERY    SELECT      <attribute name(s)>
              FROM        <TABLE NAME>
              WHERE       <column name> <criterion> / IN>

    SUBQUERY          (SELECT      <column name>
                      FROM         <table name>
                      [WHERE       <condition>]);
```

POrdId	POrdAmour
2591	$4,300.00
2593	$4,000.00
2594	$3,280.00

```
SELECT POrdId, POrdAmount
FROM PURCHASE_ORDER
WHERE POrdAmount > (SELECT AVG(POrdAmount) FROM PURCHASE_ORDER)
```

Converting Data Into Information

Subqueries in SQL – Multi Table

POrdId	PODetId	MaterialId	PODetUnits	PODetQuan	PODetBalan	PODetPromisedDel	PODetUnitCost	PODetStatu
2594	1	RM310	LB	4000	4000	3/12/2001	$0.50	OPEN
2594	2	RM311	LB	2000	2000	3/12/2001	$0.25	OPEN
2594	3	RM318	LB	2000	2000	3/12/2001	$0.25	OPEN
2594	4	RM340	LB	560	560	3/20/2001	$0.50	OPEN
2595	1	RM305	LB	400	400	2/27/2001	$0.50	OPEN
2595	2	RM308	LB	1200	1200	2/27/2001	$0.25	OPEN

table name
Main: PO_DETAIL
Sub: PURCHASE ORDER
Link: PO_NUMBER

```
SELECT *
FROM PO_DETAIL
WHERE POrdId IN (SELECT POrdId
                 FROM PURCHASE_ORDER
                 WHERE VendId = "V250");
```

The **IN** operator states that the main query is conditioned on the PO_NUMBER(s) that are returned *in* the subquery.

Converting Data Into Information

Appending Tables - JOINS

VendId	POrdRelease	PURC	PO_DE	PC	Materia	POI	PODet	PODet	PODetPro	PODet	PODets
V250	2/12/2001	2594	2594	1	RM310	LB	4000	4000	3/12/2001	$0.50	OPEN
V250	2/12/2001	2594	2594	2	RM311	LB	2000	2000	3/12/2001	$0.25	OPEN
V250	2/12/2001	2594	2594	3	RM318	LB	2000	2000	3/12/2001	$0.25	OPEN
V250	2/12/2001	2594	2594	4	RM340	LB	560	560	3/20/2001	$0.50	OPEN
V250	2/15/2001	2595	2595	1	RM305	LB	400	400	2/27/2001	$0.50	OPEN
V250	2/15/2001	2595	2595	2	RM308	LB	1200	1200	2/27/2001	$0.25	OPEN

```
SELECT PURCHASE_ORDER.VendId,
       PURCHASE_ORDER.POrdReleaseDate,
       PURCHASE_ORDER.POrdId,
       PO_DETAIL.*
FROM PURCHASE_ORDER, PO_DETAIL
WHERE PURCHASE_ORDER.POrdId = PO_DETAIL.POrdId AND
       PURCHASE_ORDER.VendId = "V250"
```

Converting Data Into Information

Appending Tables - JOINS

VendId	VendName	POrdI	POrdRelease	PC	Materi	PODet
V250	Spices Unlimited	2594	2/12/2001	1	RM310	4000
V250	Spices Unlimited	2594	2/12/2001	2	RM311	2000
V250	Spices Unlimited	2594	2/12/2001	3	RM318	2000
V250	Spices Unlimited	2594	2/12/2001	4	RM340	560
V250	Spices Unlimited	2595	2/15/2001	1	RM305	400
V250	Spices Unlimited	2595	2/15/2001	2	RM308	1200

```
SELECT VENDOR.VendId, VENDOR.VendName,
        PURCHASE_ORDER.POrdId, PURCHASE_ORDER.POrdReleaseDate,
        PO_DETAIL.PODetId, PO_DETAIL.MaterialId, PO_DETAIL.PODetQuantity
FROM VENDOR, PURCHASE_ORDER, PO_DETAIL
WHERE VENDOR.VendId = PURCHASE_ORDER.VendId
AND PURCHASE_ORDER.POrdId = PO_DETAIL.POrdId
AND VENDOR.VendId = "V250";
```

Classwork - Homework

- In Lab:
  - 2.4

- Homework:
  - 2.4

Questions

Questions?

532 2877127

hposaci@quiztechnology.com