

## About T-SQL

- Structured Query Language (SQL)
  - Developed by IBM in 1970s
  - Adopted as a standard by ANSI and ISO standards bodies
  - Widely used in industry
    - PL/SQL, SQL Procedural Language, Transact-SQL
- Microsoft's implementation is Transact-SQL
  - Referred to as T-SQL
  - Query language for SQL Server 2012, 2014, 2016, 2017

# About T-SQL

Display all customers whose city is Portland.

## Procedural approach: Loop through each row in the data

- 1) Open a cursor to consume rows, one at a time
- 2) Fetch the first cursor record.
- 3) Examine first row.
- 4) If the city is Portland, return the row.
- 5) Move to next row.
- 6) If the city is Portland, return the row.
- 7) Fetch the next record
- 8) (Repeat until end of table is reached).

- SQL is declarative, not procedural
  - Describe **what** you want, don't specify steps

```
SELECT *
FROM Sales.Customers
WHERE city = 'Portland'
```

# Categories of T-SQL Statements

## Data Manipulation Language (DML\*)

- Statements for querying and modifying data
- SELECT, INSERT, UPDATE, DELETE

## Data Definition Language (DDL)

- Statements for object definitions
- CREATE, ALTER, DROP

## Data Control Language (DCL)

- Statements for security permissions
- GRANT, REVOKE, DENY

```
<SELECT statement> ::=  
  [WITH <common_table_expression> [ ,...n ]]  
  <query_expression>  
    [ ORDER BY { order_by_expression | column_position [ ASC | DESC ] }  
    [ ,...n ] ]  
    [ <FOR Clause>]  
    [ OPTION ( <query_hint> [ ,...n ] ) ]
```

```
<query_expression> ::=  
  { <query_specification> | ( <query_expression> ) }  
  [ { UNION [ ALL ] | EXCEPT | INTERSECT }  
    <query_specification> | ( <query_expression> ) [ ...n ] ]
```

```
<query_specification> ::=  
  SELECT [ ALL | DISTINCT ]  
    [TOP ( expression ) [PERCENT] [ WITH TIES ] ]  
    < select_list > → { * | column_name | expression } [ ,...n ]  
    [ INTO new_table ]  
    [ FROM { <table_source> } [ ,...n ] ]  
    [ WHERE <search_condition> ]  
    [ <GROUP BY> ]  
    [ HAVING < search_condition > ]
```

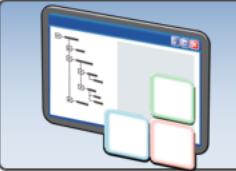
**SELECT \***  
**FROM Sales.Customers**  
**WHERE city = 'Portland'**

# T-SQL Language Elements

Predicates and Operators



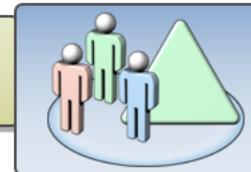
Batch Separators



Functions



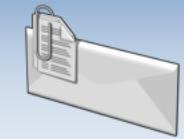
Control of Flow



Variables



Comments



Expressions



# T-SQL Language Elements: Predicates and Operators

<b>Elements:</b>	<b>Predicates and Operators:</b>
Predicates	IN, BETWEEN, LIKE
Comparison Operators	=, >, <, >=, <=, <>, !=, !>, !<
Logical Operators	AND, OR, NOT
Arithmetic Operators	+, -, *, /, %
Concatenation	+

## T-SQL Language Elements: Predictors

```
SELECT empid, firstname, lastname  
FROM HR.Employees ;
```

	empid	firstname	lastname
1	1	Sara	Davis
2	2	Don	Funk
3	3	Judy	Lew
4	4	Yael	Peled
5	5	Sven	Buck
6	6	Paul	Suurs
7	7	Russell	King
8	8	Maria	Cameron
9	9	Zoya	Dolgopyatova

```
SELECT empid,  
firstname + ' ' + lastname AS FullName  
FROM HR.Employees ;
```

alias

	empid	FullName
1	1	Sara Davis
2	2	Don Funk
3	3	Judy Lew
4	4	Yael Peled
5	5	Sven Buck
6	6	Paul Suurs
7	7	Russell King
8	8	Maria Cameron
9	9	Zoya Dolgop...

Concatenation

+

# T-SQL Language Elements: Predicates and Operators

Elements:	Predicates and Operators:
Predicates	IN, BETWEEN, LIKE
Comparison Operators	=, >, <, >=, <=, <>, !=, !>, !<
Logical Operators	AND, OR, NOT
Arithmetic Operators	+ , - , * , / , %
Concatenation	+

$$\begin{array}{rcl} 12 * 3 & = & 36 \\ 12 / 3 & = & 4 \\ 12 \% 3 & = & 0 \end{array}$$

$$\begin{array}{rcl} 24 - 16 / 4 & = & 20 \\ (24 - 16) / 4 & = & 2 \\ 12 \% 3 * 2 & = & 0 \end{array}$$

**T-SQL enforces operator precedence**

Order of Evaluation	Operators
1	( ) Parentheses
2	*, /, % (Multiply, Divide, Modulo)
3	+, - (Add/Positive/Concatenate, Subtract/Negative)
4	=, <, >, >=, <=, !=, !>, !< (Comparison)

```
SELECT *
FROM Sales.OrderDetails
WHERE unitprice * qty < 25;
```



Results    Messages

	orderid	productid	unitprice	qty	discount
1	10281	19	7.30	1	0.000
2	10420	13	4.80	2	0.100
3	10462	13	4.80	1	0.000
4	10623	24	4.50	3	0.000
5	10782	31	12.50	1	0.000
6	10850	33	2.50	4	0.150
7	11038	52	7.00	2	0.000
8	11077	52	7.00	2	0.000

Order of Evaluation	Operators
1	( ) Parentheses
2	*, /, % (Multiply, Divide, Modulo)
3	+, - (Add/Positive/Concatenate, Subtract/Negative)
4	=, <, >, >=, <=, !=, !>, !< (Comparison)
5	NOT

```
SELECT lastname, city, titleofcourtesy
FROM HR.Employees
WHERE NOT city = 'London' ;
```

lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Buck	London	Mr.
Suurs	London	Mr.
King	London	Mr.
Cameron	Seattle	Ms.
Dolgopyatova	London	Ms.



```
SELECT lastname, city, titleofcourtesy
FROM HR.Employees
WHERE city <> 'London' ;
```

lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Cameron	Seattle	Ms.

Order of Evaluation	Operators
1	( ) Parentheses
2	*, /, % (Multiply, Divide, Modulo)
3	+, - (Add/Positive/Concatenate, Subtract/Negative)
4	=, <, >, >=, <=, !=, !>, !< (Comparison)
5	NOT
6	AND

```
SELECT lastname, city, titleofcourtesy
FROM HR.Employees
WHERE NOT city = 'London'
      AND titleofcourtesy LIKE 'Ms%';
```

lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Buck	London	Mr.
Suurs	London	Mr.
King	London	Mr.
Cameron	Seattle	Ms.
Dolgopyatova	London	Ms.



lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Cameron	Seattle	Ms.



lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Lew	Kirkland	Ms.
Cameron	Seattle	Ms.

## Order of Evaluation

1

## Operators

( ) Parentheses

2

\*, /, % (Multiply, Divide, Modulo)

3

+, - (Add/Positive/Concatenate, Subtract/Negative)

4

=, <, >, >=, <=, !=, !>, !< (Comparison)

5 NOT

`SELECT lastname, city, titleofcourtesy`

`FROM HR.Employees`

6 AND

`WHERE NOT city = 'London'`

`AND titleofcourtesy LIKE 'Ms%'`

7 IN, OR

`OR city = 'Tacoma';`

lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Buck	London	Mr.
Suurs	London	Mr.
King	London	Mr.
Cameron	Seattle	Ms.
Dolgopyatova	London	Ms.



lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Peled	Redmond	Mrs.
Cameron	Seattle	Ms.



lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Funk	Tacoma	Dr.
Lew	Kirkland	Ms.
Cameron	Seattle	Ms.



lastname	city	titleofcourtesy
Davis	Seattle	Ms.
Lew	Kirkland	Ms.
Cameron	Seattle	Ms.



Order of Evaluation	Operators
1	( ) Parentheses
2	*, /, % (Multiply, Divide, Modulo)
3	+, - (Add/Positive/Concatenate, Subtract/Negative)
4	=, <, >, >=, <=, !=, !>, !< (Comparison)
5	NOT
6	AND
7	IN, OR
8	= (Assignment)

## String Functions

- SUBSTRING
- LEFT, RIGHT
- LEN
- DATALENGTH
- REPLACE
- REPLICATE
- UPPER, LOWER
- RTRIM, LTRIM

```
SELECT empid, UPPER(firstname),  
lastname  
FROM HR.Employees
```

	empid	(No column name)	lastname
1	1	SARA	Davis
2	2	DON	Funk
3	3	JUDY	Lew
4	4	Yael	Peled
5	5	SVEN	Buck
6	6	PAUL	Suurs
7	7	RUSSELL	King
8	8	MARIA	Cameron
9	9	ZOYA	Dolgopyatova

```
SELECT empid, firstname, lastname  
FROM HR.Employees  
WHERE LEFT(lastname,1) = 'D';
```

	empid	firstname	lastname
1	1	Sara	Davis
2	9	Zoya	Dolgopyatova

```
SELECT empid, lastname, birthdate  
FROM HR.Employees ;
```

empid	lastname	birthdate
1	Davis	1958-12-08 00:00:00.000
2	Funk	1962-02-19 00:00:00.000
3	Lew	1973-08-30 00:00:00.000
4	Peled	1947-09-19 00:00:00.000
5	Buck	1965-03-04 00:00:00.000
6	Suurs	1973-07-02 00:00:00.000
7	King	1970-05-29 00:00:00.000
8	Cameron	1968-01-09 00:00:00.000
9	Dolgopyatova	1976-01-27 00:00:00.000

```
SELECT empid, lastname,  
YEAR(birthdate) AS [Year of Birth]  
FROM HR.Employees  
WHERE YEAR(birthdate) BETWEEN 1970 AND 1980;
```

empid	lastname	Year of Birth
3	Lew	1973
6	Suurs	1973
7	King	1970
9	Dolgopyatova	1976

```
SELECT COUNT(country) AS UK_Customers  
FROM Sales.Customers  
WHERE country = 'UK' ;
```

UK_Customers
7

```
SELECT country,  
       COUNT(*) AS [# of Customers]  
  FROM Sales.Customers  
 GROUP BY country  
 ORDER BY COUNT(*) DESC;
```

```
SELECT country,  
       COUNT(*) AS [# of Customers]  
  FROM Sales.Customers  
 GROUP BY country  
 HAVING COUNT(*) > 10  
 ORDER BY COUNT(*) DESC;
```

country	# of Customers
USA	13
France	11
Germany	11

country	# of Customers
USA	13
France	11
Germany	11
Brazil	9
UK	7
Spain	5
Mexico	5
Venezuela	4
Italy	3
Canada	3
Argentina	3
Austria	2
Belgium	2
Denmark	2
Finland	2
Portugal	2
Sweden	2
Switzerland	2
Norway	1
Poland	1
Ireland	1

```
SELECT COUNT(*) AS [# of Orders],  
       SUM(freight) AS TotalFreight,  
       AVG(freight) AS AverageFreight,  
       MAX(freight) AS MaxFreight,  
       MIN(freight) AS MinFreight  
FROM Sales.Orders ;
```

# of Orders	TotalFreight	AverageFreight	MaxFreight	MinFreight
830	64942,69	78,2442	1007,64	0,02

S

## Aggregate Functions

- SUM
- MIN
- MAX
- AVG
- COUNT

## T-SQL Language Elements: Variables

- Local variables in T-SQL temporarily store a value of a specific data type
- Name begins with single @ sign
  - @@ reserved for system functions
- Assigned a data type
- Must be declared and used within the same batch
- In SQL Server 2008 and later, you can declare and initialize in the same statement

@MyVar

**DECLARE @MyVar int = 30;**

```
DECLARE @MyDate date;
SET @MyDate = '20120215';
```

## T-SQL Language Elements: Expressions

- Combination of identifiers, values, and operators evaluated to obtain a single result
- Can be used in SELECT statements
  - SELECT clause
  - WHERE clause
- Can be single constant, single-valued function, or variable
- Can be combined if expressions have the same data type

```
SELECT YEAR(orderdate) + 1 ...
```

```
SELECT qty * unitprice ...
```

## T-SQL Language Elements: Expressions

- Combination of identifiers, values, and operators evaluated to obtain a single result
- Can be used in SELECT statements
  - SELECT clause
  - WHERE clause
- Can be single constant, single-valued function, or variable
- Can be combined if expressions have the same data type

```
SELECT YEAR(orderdate) + 1 ...
```

```
SELECT qty * unitprice ...
```

# T-SQL Language Elements: Comments

- Marks T-SQL code as a comment:
  - For a block, enclose it between /\* and \*/ characters

```
/*
    This is a block
    of commented code
*/
```

- For inline text, precede the comments with --

```
-- This line of text will be ignored
```

- T-SQL Editors such as SSMS will typically color-code comments, as shown above

## T-SQL Language Elements: Batch Separators

- Batches are sets of commands sent to SQL Server as a unit
- Batches determine variable scope

```
DECLARE @custid int = 5 ;
GO
SELECT custid, companynname, contactname
FROM Sales.Customers
WHERE custid = @custid;
```

It returns the following error:

```
Msg 137, Level 15, State 2, Line 3
Must declare the scalar variable "@custid".
```

## T-SQL Language Elements: Batch Separators

- Batches are sets of commands sent to SQL Server as a unit
- Batches determine variable scope
- To separate statements into batches, use a separator:
  - SQL Server tools use the GO keyword
  - GO is not a SQL Server T-SQL command!

## Predicate Logic and SQL Server

- Predicate logic is a mathematical basis for the relational database model.
- In theory, a predicate is a property or expression that is either true or false.
- Predicate is also referred to as a Boolean expression.

# Predicate Logic Applied to SQL Server Queries

- In SQL Server, a predicate is a property or expression that evaluates to true, false, or unknown (NULL).

## Uses for Predicates

- Filtering data in queries (WHERE and HAVING clauses)

```
SELECT empid, YEAR(orderdate) AS orderyear  
FROM Sales.Orders  
WHERE custid =71;
```

```
SELECT empid, YEAR(orderdate) AS orderyear  
FROM Sales.Orders  
WHERE custid =71  
GROUP BY empid, YEAR(orderdate)  
HAVING COUNT(*) > 1
```

# Predicate Logic Applied to SQL Server Queries

- In SQL Server, a predicate is a property or expression that evaluates to true, false, or unknown (NULL).

## Uses for Predicates

- Filtering data in queries (WHERE and HAVING clauses)
- Providing conditional logic to CASE expressions
- Joining tables (ON filter)
- Defining subqueries
- Enforcing data integrity (CHECK constraints)
- Control of flow (IF statement)

# Elements of a SELECT Statement

Element	Expression	Role
SELECT	<select list>	Defines which columns to return
FROM	<table source>	Defines table(s) to query
WHERE	<search condition>	Filters rows using a predicate
GROUP BY	<group by list>	Arranges rows by groups
HAVING	<search condition>	Filters groups using a predicate
ORDER BY	<order by list>	Sorts the output

```
SELECT empid, YEAR(orderdate) AS orderyear  
FROM Sales.Orders  
WHERE custid =71  
GROUP BY empid, YEAR(orderdate)  
HAVING COUNT(*) > 1  
ORDER BY empid, orderyear;
```

# Logical Query Processing

- The order in which a query is written is not the order in which it is evaluated by SQL Server.

5: SELECT <select list>

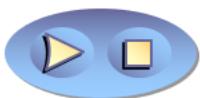
1: FROM <table source>

2: WHERE <search condition>

3: GROUP BY <group by list>

4: HAVING <search condition>

6: ORDER BY <order by list>



# Applying the Logical Order of Operations to Writing SELECT Statements

```
USE TSQL2012;
```

```
SELECT empid, YEAR(orderdate) AS orderyear  
FROM Sales.Orders  
WHERE custid =71  
GROUP BY empid, YEAR(orderdate)  
HAVING COUNT(*) > 1  
ORDER BY empid, orderyear;
```



USE TSQL2012;

**FROM**  
**Sales.Orders**

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight
10248	85	5	2006-07-04 00:00:00.000	2006-08-01 00:00:00.000	2006-07-16 00:00:00.000	3	10.24
10249	79	6	2006-07-05 00:00:00.000	2006-08-16 00:00:00.000	2006-07-10 00:00:00.000	1	126.56
10250	34	4	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-12 00:00:00.000	2	89.16
10251	84	3	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-15 00:00:00.000	1	214.27
10252	76	4	2006-07-09 00:00:00.000	2006-08-06 00:00:00.000	2006-07-11 00:00:00.000	2	86.53
10253	34	3	2006-07-10 00:00:00.000	2006-07-24 00:00:00.000	2006-07-16 00:00:00.000	2	367.63
10254	14	5	2006-07-11 00:00:00.000	2006-08-08 00:00:00.000	2006-07-23 00:00:00.000	2	252.49
10255	68	9	2006-07-12 00:00:00.000	2006-08-09 00:00:00.000	2006-07-15 00:00:00.000	3	48.77
10256	88	3	2006-07-15 00:00:00.000	2006-08-12 00:00:00.000	2006-07-17 00:00:00.000	2	200.24
10257	35	4	2006-07-16 00:00:00.000	2006-08-13 00:00:00.000	2006-07-22 00:00:00.000	3	544.09

(830 rows affected)

**WHERE custid = 71**

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight
10324	71	9	2006-10-08 00:00:00.000	2006-11-05 00:00:00.000	2006-10-10 00:00:00.000	1	214.27
10393	71	1	2006-12-25 00:00:00.000	2007-01-22 00:00:00.000	2007-01-03 00:00:00.000	3	126.56
10398	71	2	2006-12-30 00:00:00.000	2007-01-27 00:00:00.000	2007-01-09 00:00:00.000	3	89.16
10440	71	4	2007-02-10 00:00:00.000	2007-03-10 00:00:00.000	2007-02-28 00:00:00.000	2	86.53
10452	71	8	2007-02-20 00:00:00.000	2007-03-20 00:00:00.000	2007-02-26 00:00:00.000	1	140.26
10510	71	6	2007-04-18 00:00:00.000	2007-05-16 00:00:00.000	2007-04-28 00:00:00.000	3	367.63
10555	71	6	2007-06-02 00:00:00.000	2007-06-30 00:00:00.000	2007-06-04 00:00:00.000	3	252.49
10603	71	8	2007-07-18 00:00:00.000	2007-08-15 00:00:00.000	2007-08-08 00:00:00.000	2	48.77
10607	71	5	2007-07-22 00:00:00.000	2007-08-19 00:00:00.000	2007-07-25 00:00:00.000	1	200.24
10612	71	1	2007-07-28 00:00:00.000	2007-08-25 00:00:00.000	2007-08-01 00:00:00.000	2	544.09

(31 rows affected)

**GROUP BY empid, YEAR(orderdate)**

## GROUP BY empid, YEAR(orderdate)

**16 groups**

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight	shipname	shipaddress	shipcity	shipregion	shippostalcode	shipcountry
10393	71	1	2006-12-25 00:00:00.000	2007-01-22 00:00:00.000	2007-01-03 00:00:00.000	3	126,56	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10612	71	1	2007-07-28 00:00:00.000	2007-08-25 00:00:00.000	2007-08-01 00:00:00.000	2	544,08	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10713	71	1	2007-10-22 00:00:00.000	2007-11-19 00:00:00.000	2007-10-24 00:00:00.000	1	167,05	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10894	71	1	2008-02-18 00:00:00.000	2008-03-18 00:00:00.000	2008-02-20 00:00:00.000	1	116,13	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10984	71	1	2008-03-30 00:00:00.000	2008-04-27 00:00:00.000	2008-04-03 00:00:00.000	3	211,22	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11064	71	1	2008-05-01 00:00:00.000	2008-05-29 00:00:00.000	2008-05-04 00:00:00.000	1	30,09	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10398	71	2	2006-12-30 00:00:00.000	2007-01-27 00:00:00.000	2007-01-09 00:00:00.000	3	89,16	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10657	71	2	2007-09-04 00:00:00.000	2007-10-02 00:00:00.000	2007-09-15 00:00:00.000	2	352,69	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10983	71	2	2008-03-27 00:00:00.000	2008-04-24 00:00:00.000	2008-04-06 00:00:00.000	2	657,54	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10815	71	2	2008-01-05 00:00:00.000	2008-02-02 00:00:00.000	2008-01-14 00:00:00.000	3	14,62	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10748	71	3	2007-11-20 00:00:00.000	2007-12-18 00:00:00.000	2007-11-28 00:00:00.000	1	232,55	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10700	71	3	2007-10-10 00:00:00.000	2007-11-07 00:00:00.000	2007-10-16 00:00:00.000	1	65,10	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10440	71	4	2007-02-10 00:00:00.000	2007-03-10 00:00:00.000	2007-02-28 00:00:00.000	2	86,53	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10847	71	4	2008-01-22 00:00:00.000	2008-02-05 00:00:00.000	2008-02-10 00:00:00.000	3	487,57	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10882	71	4	2008-02-11 00:00:00.000	2008-03-11 00:00:00.000	2008-02-20 00:00:00.000	3	23,10	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11002	71	4	2008-04-06 00:00:00.000	2008-05-04 00:00:00.000	2008-04-16 00:00:00.000	1	141,16	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10607	71	5	2007-07-22 00:00:00.000	2007-08-19 00:00:00.000	2007-07-25 00:00:00.000	1	200,24	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10711	71	5	2007-10-21 00:00:00.000	2007-12-02 00:00:00.000	2007-10-29 00:00:00.000	2	52,41	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10714	71	5	2007-10-22 00:00:00.000	2007-11-19 00:00:00.000	2007-10-27 00:00:00.000	3	24,49	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10510	71	6	2007-04-18 00:00:00.000	2007-05-16 00:00:00.000	2007-04-28 00:00:00.000	3	367,63	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10555	71	6	2007-06-02 00:00:00.000	2007-06-30 00:00:00.000	2007-06-04 00:00:00.000	3	252,49	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10757	71	6	2007-11-27 00:00:00.000	2007-12-25 00:00:00.000	2007-12-15 00:00:00.000	1	8,19	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11031	71	6	2008-04-17 00:00:00.000	2008-05-15 00:00:00.000	2008-04-24 00:00:00.000	2	227,22	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10678	71	7	2007-09-23 00:00:00.000	2007-10-21 00:00:00.000	2007-10-16 00:00:00.000	3	388,98	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
11030	71	7	2008-04-17 00:00:00.000	2008-05-15 00:00:00.000	2008-04-27 00:00:00.000	2	830,75	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10941	71	7	2008-03-11 00:00:00.000	2008-04-08 00:00:00.000	2008-03-20 00:00:00.000	2	400,81	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10627	71	8	2007-08-11 00:00:00.000	2007-09-22 00:00:00.000	2007-08-21 00:00:00.000	3	107,46	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10722	71	8	2007-10-29 00:00:00.000	2007-12-10 00:00:00.000	2007-11-04 00:00:00.000	1	74,58	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10603	71	8	2007-07-18 00:00:00.000	2007-08-15 00:00:00.000	2007-08-08 00:00:00.000	2	48,77	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10452	71	8	2007-02-20 00:00:00.000	2007-03-20 00:00:00.000	2007-02-26 00:00:00.000	1	140,26	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10324	71	9	2006-10-08 00:00:00.000	2006-11-05 00:00:00.000	2006-10-10 00:00:00.000	1	214,27	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA

(31 rows affected)

HAVING COUNT(\*) > 1

# HAVING COUNT(\*) > 1

**16-7 groups = 9**

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight	shipname	shipaddress	shipcity	shipregion	shippostalcode	shipcountry
10393	71	1	2006-12-25 00:00:00.000	2007-01-22 00:00:00.000	2007-01-03 00:00:00.000	3	126,56	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10612	71	1	2007-07-28 00:00:00.000	2007-08-25 00:00:00.000	2007-08-01 00:00:00.000	2	544,08	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10713	71	1	2007-10-22 00:00:00.000	2007-11-19 00:00:00.000	2007-10-24 00:00:00.000	1	167,05	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10894	71	1	2008-02-18 00:00:00.000	2008-03-18 00:00:00.000	2008-02-20 00:00:00.000	1	116,13	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10984	71	1	2008-03-30 00:00:00.000	2008-04-27 00:00:00.000	2008-04-03 00:00:00.000	3	211,22	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11064	71	1	2008-05-01 00:00:00.000	2008-05-29 00:00:00.000	2008-05-04 00:00:00.000	1	30,09	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10398	71	2	2006-12-30 00:00:00.000	2007-01-27 00:00:00.000	2007-01-09 00:00:00.000	3	89,16	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10657	71	2	2007-09-04 00:00:00.000	2007-10-02 00:00:00.000	2007-09-15 00:00:00.000	2	352,69	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10983	71	2	2008-03-27 00:00:00.000	2008-04-24 00:00:00.000	2008-04-06 00:00:00.000	2	657,54	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10815	71	2	2008-01-05 00:00:00.000	2008-02-02 00:00:00.000	2008-01-14 00:00:00.000	3	14,62	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10748	71	3	2007-11-20 00:00:00.000	2007-12-18 00:00:00.000	2007-11-28 00:00:00.000	1	232,55	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10700	71	3	2007-10-10 00:00:00.000	2007-11-07 00:00:00.000	2007-10-16 00:00:00.000	1	65,10	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10440	71	4	2007-02-10 00:00:00.000	2007-03-10 00:00:00.000	2007-02-28 00:00:00.000	2	86,53	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10847	71	4	2008-01-22 00:00:00.000	2008-02-05 00:00:00.000	2008-02-10 00:00:00.000	3	487,57	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10882	71	4	2008-02-11 00:00:00.000	2008-03-11 00:00:00.000	2008-02-20 00:00:00.000	3	23,10	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11002	71	4	2008-04-06 00:00:00.000	2008-05-04 00:00:00.000	2008-04-16 00:00:00.000	1	141,16	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10607	71	5	2007-07-22 00:00:00.000	2007-08-19 00:00:00.000	2007-07-25 00:00:00.000	1	200,24	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10711	71	5	2007-10-21 00:00:00.000	2007-12-02 00:00:00.000	2007-10-29 00:00:00.000	2	52,41	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10714	71	5	2007-10-22 00:00:00.000	2007-11-19 00:00:00.000	2007-10-27 00:00:00.000	3	24,49	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10510	71	6	2007-04-18 00:00:00.000	2007-05-16 00:00:00.000	2007-04-28 00:00:00.000	3	367,63	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10555	71	6	2007-06-02 00:00:00.000	2007-06-30 00:00:00.000	2007-06-04 00:00:00.000	3	252,49	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10757	71	6	2007-11-27 00:00:00.000	2007-12-25 00:00:00.000	2007-12-15 00:00:00.000	1	8,19	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
11031	71	6	2008-04-17 00:00:00.000	2008-05-15 00:00:00.000	2008-04-24 00:00:00.000	2	227,22	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10678	71	7	2007-09-23 00:00:00.000	2007-10-21 00:00:00.000	2007-10-16 00:00:00.000	3	388,98	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
11030	71	7	2008-04-17 00:00:00.000	2008-05-15 00:00:00.000	2008-04-27 00:00:00.000	2	830,75	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10941	71	7	2008-03-11 00:00:00.000	2008-04-08 00:00:00.000	2008-03-20 00:00:00.000	2	400,81	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10627	71	8	2007-08-11 00:00:00.000	2007-09-22 00:00:00.000	2007-08-21 00:00:00.000	3	107,46	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10722	71	8	2007-10-29 00:00:00.000	2007-12-10 00:00:00.000	2007-11-04 00:00:00.000	1	74,58	Ship to 71-A	7890 Suffolk Ln.	Boise	ID	10305	USA
10603	71	8	2007-07-18 00:00:00.000	2007-08-15 00:00:00.000	2007-08-08 00:00:00.000	2	48,77	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA
10452	71	8	2007-02-20 00:00:00.000	2007-03-20 00:00:00.000	2007-02-26 00:00:00.000	1	140,26	Ship to 71-B	8901 Suffolk Ln.	Boise	ID	10306	USA
10324	71	9	2006-10-08 00:00:00.000	2006-11-05 00:00:00.000	2006-10-10 00:00:00.000	1	214,27	Ship to 71-C	9012 Suffolk Ln.	Boise	ID	10307	USA

(31 rows affected)

HAVING COUNT(\*) > 1

9 groups

orderid	custid	empid	orderdate
10393	71	1	2006-12-25 00:00:00.000
10612	71	1	2007-07-28 00:00:00.000
10713	71	1	2007-10-22 00:00:00.000
10894	71	1	2008-02-18 00:00:00.000
10984	71	1	2008-03-30 00:00:00.000
11064	71	1	2008-05-01 00:00:00.000
10398	71	2	2006-12-30 00:00:00.000
10657	71	2	2007-09-04 00:00:00.000
10983	71	2	2008-03-27 00:00:00.000
10815	71	2	2008-01-05 00:00:00.000
10748	71	3	2007-11-20 00:00:00.000
10700	71	3	2007-10-10 00:00:00.000
10440	71	4	2007-02-10 00:00:00.000
10847	71	4	2008-01-22 00:00:00.000
10882	71	4	2008-02-11 00:00:00.000
11002	71	4	2008-04-06 00:00:00.000
10607	71	5	2007-07-22 00:00:00.000
10711	71	5	2007-10-21 00:00:00.000
10714	71	5	2007-10-22 00:00:00.000
10510	71	6	2007-04-18 00:00:00.000
10555	71	6	2007-06-02 00:00:00.000
10757	71	6	2007-11-27 00:00:00.000
11031	71	6	2008-04-17 00:00:00.000
10678	71	7	2007-09-23 00:00:00.000
11030	71	7	2008-04-17 00:00:00.000
10941	71	7	2008-03-11 00:00:00.000
10627	71	8	2007-08-11 00:00:00.000
10722	71	8	2007-10-29 00:00:00.000
10603	71	8	2007-07-18 00:00:00.000
10452	71	8	2007-02-20 00:00:00.000
10324	71	9	2006-10-08 00:00:00.000

SELECT empid,  
YEAR(orderdate) AS orderyear

1	2007
1	2008
2	2008
3	2007
4	2008
5	2007
6	2007
7	2007
8	2007

empid	orderyear
1	2007
1	2008
2	2008
3	2007
4	2008
5	2007
6	2007
7	2008
8	2007

ORDER BY empid, orderyear;

empid	orderyear
1	2007
1	2008
2	2008
3	2007
4	2008
5	2007
6	2007
7	2008
8	2007