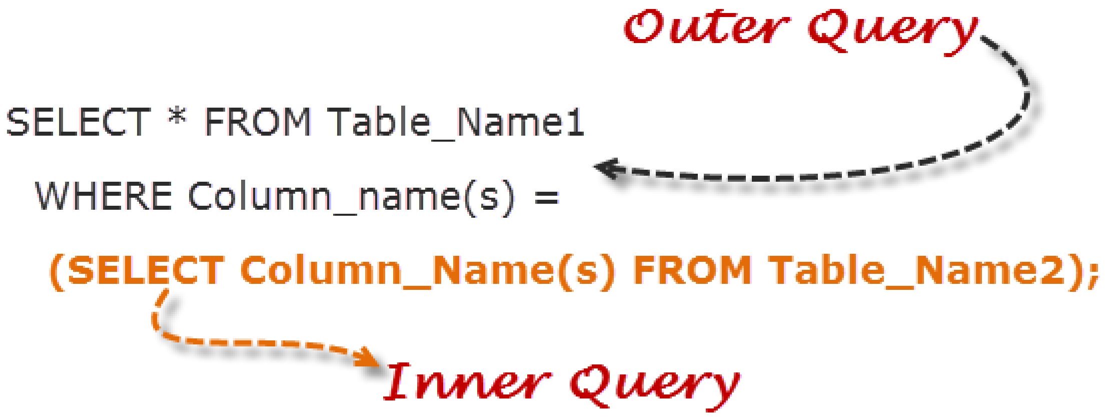
SUBQUERY

LEARNING GOALS

By the end of this lecture students should be able to:

- ✓ Understand subqueries in SQL Server
- ✓ Use smoothly subqueries and apply to real projects



```
SELECT * FROM Table_Name1
  WHERE Column_name(s) =
   (SELECT Column_Name(s) FROM Table_Name2);
```

Lesson Agenda

- What is a subquery?
- Type of Subqueries
- Common cases use subquery
- Rules that subqueries must follow

Section 1

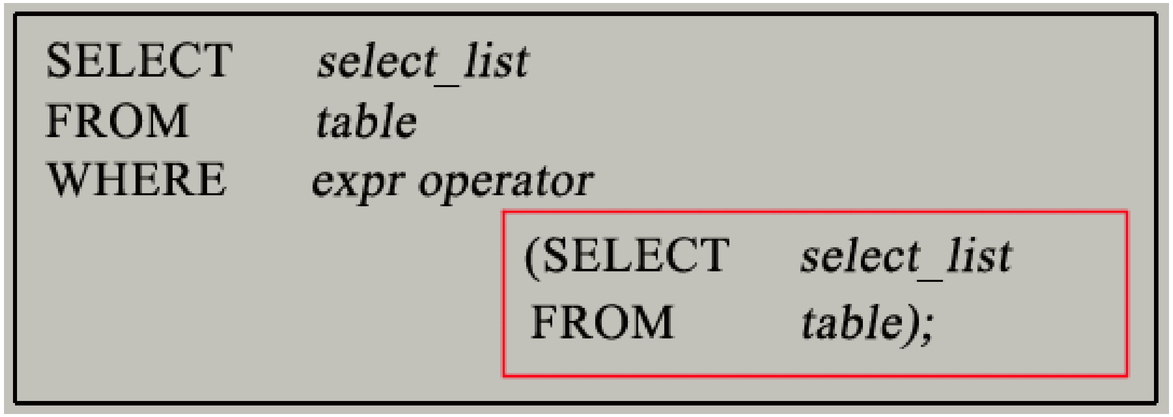
WHAT’S A SUBQUERY?

What Is A subquery?

- A sub-query, also called an inner query, is a SQL query nested inside a larger query.
- The subquery can be nested inside a SELECT, INSERT, UPDATE, or DELETE statement or inside another subquery.
- You can use the comparison operators, such as >, <, or =. The comparison operator can also be a multiple-row operator, such as IN, ANY, or ALL.

What Is A subquery?

- **Syntax (example: subquery within the Where)** :



```
SELECT    select_list
FROM      table
WHERE     expr operator
                      (SELECT    select_list
                       FROM      table);
```

Exam:

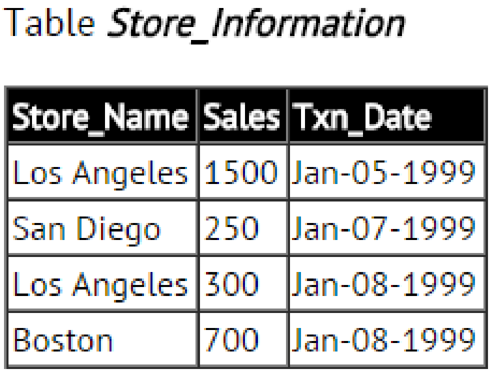


Table *Store_Information*

Store_Name	Sales	Txn_Date
Los Angeles	1500	Jan-05-1999
San Diego	250	Jan-07-1999
Los Angeles	300	Jan-08-1999
Boston	700	Jan-08-1999

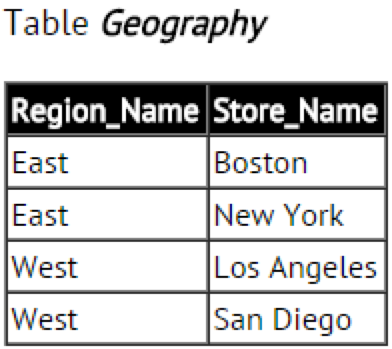


Table *Geography*

Region_Name	Store_Name
East	Boston
East	New York
West	Los Angeles
West	San Diego

Sale_Sum
2050

```
SELECT SUM (Sales) AS Sale_Sum FROM Store_Information
WHERE Store_Name IN
        (SELECT Store_Name FROM Geography WHERE Region_Name = 'West');
```

What Is A subquery? (3/3)

- **How to work?**:
  - ✓ Inner query is independent of outer query.
  - ✓ Inner query is executed first and the results are stored.
  - ✓ Outer query then runs on the stored results.
- *Note about specific type*: **Correlated** subqueries (be mentioned in the next slides)

Section 2

TYPE OF SUBQUERIES

Subquery Types

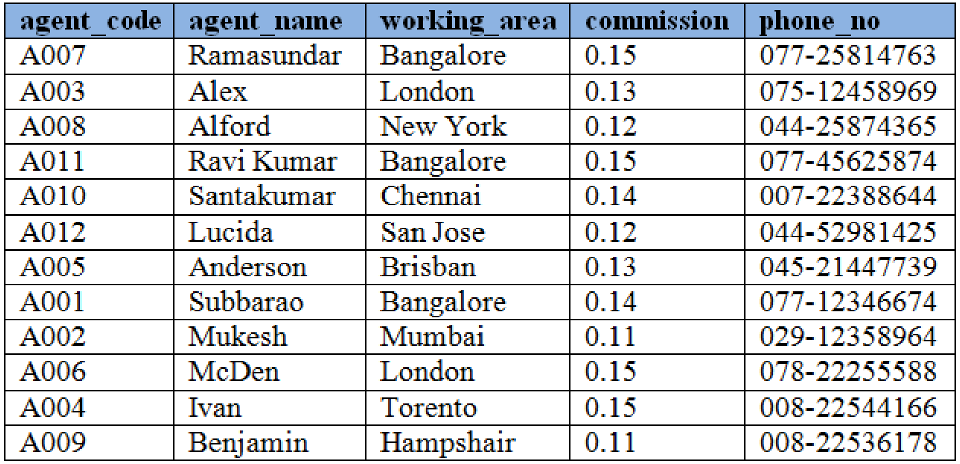
- **Single row** subquery
- **Multiple row** subquery
- **Multiple column** subquery
- **Correlated** subquery
- **Nested** subquery

Single row subquery

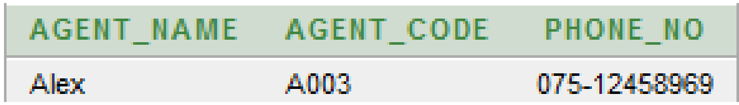
- A single row subquery returns zero or one row to the outer SQL statement. You can place a subquery in a **WHERE** clause, a **HAVING** clause, or a **FROM** clause of a **SELECT** statement.
- **Exam**: Single Row subqueries in WHERE clause

```
SELECT agent_name, agent_code, phone_no
FROM agents
WHERE agent_code =
      (SELECT agent_code FROM agents WHERE agent_name = 'Alex')
```

agents



agent_code	agent_name	working_area	commission	phone_no
A007	Ramasundar	Bangalore	0.15	077-25814763
A003	Alex	London	0.13	075-12458969
A008	Alford	New York	0.12	044-25874365
A011	Ravi Kumar	Bangalore	0.15	077-45625874
A010	Santakumar	Chennai	0.14	007-22388644
A012	Lucida	San Jose	0.12	044-52981425
A005	Anderson	Brisban	0.13	045-21447739
A001	Subbarao	Bangalore	0.14	077-12346674
A002	Mukesh	Mumbai	0.11	029-12358964
A006	McDen	London	0.15	078-22255588
A004	Ivan	Torento	0.15	008-22544166
A009	Benjamin	Hampshair	0.11	008-22536178

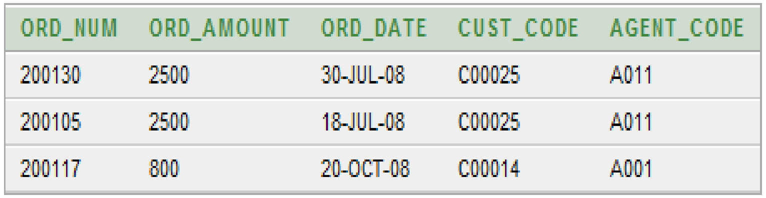


AGENT_NAME	AGENT_CODE	PHONE_NO
Alex	A003	075-12458969

Multiple row subquery

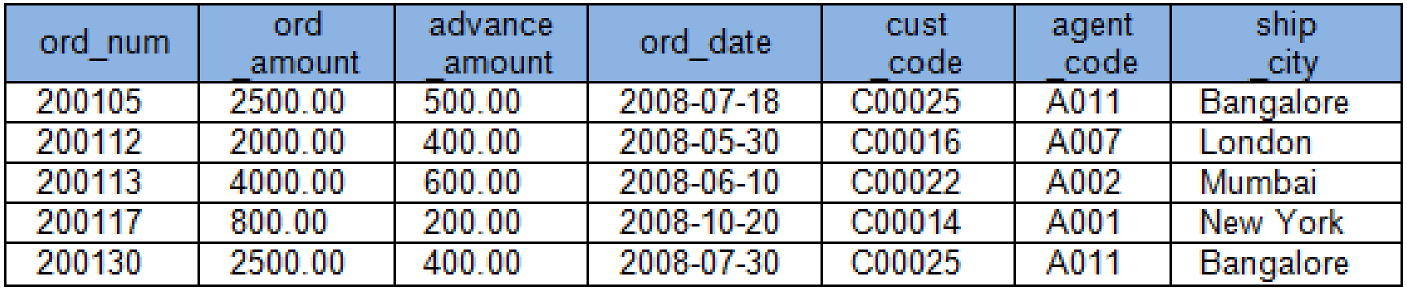
- Multiple row subquery returns one or more rows to the outer SQL statement. You may use the **IN**, **ANY**, or **ALL** operator in outer query to handle a subquery that returns multiple rows.
- **Ex**: Multiple row Subquery in a WHERE clause

```
SELECT ord_num, ord_amount, ord_date, cust_code, agent_code
FROM orders
WHERE agent_code IN (
          SELECT agent_code FROM agents
          WHERE working_area='Bangalore'
)
```



ORD_NUM	ORD_AMOUNT	ORD_DATE	CUST_CODE	AGENT_CODE
200130	2500	30-JUL-08	C00025	A011
200105	2500	18-JUL-08	C00025	A011
200117	800	20-OCT-08	C00014	A001

orders

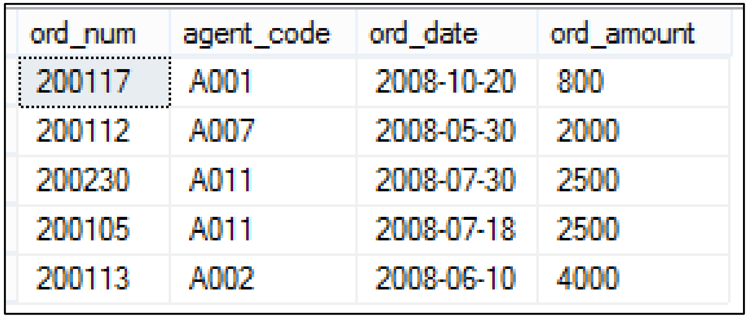


ord_num	ord _amount	advance _amount	ord_date	cust _code	agent _code	ship _city
200105	2500.00	500.00	2008-07-18	C00025	A011	Bangalore
200112	2000.00	400.00	2008-05-30	C00016	A007	London
200113	4000.00	600.00	2008-06-10	C00022	A002	Mumbai
200117	800.00	200.00	2008-10-20	C00014	A001	New York
200130	2500.00	400.00	2008-07-30	C00025	A011	Bangalore

Multiple column subquery

- You can write subqueries that return multiple columns.
- **Ex**: Multiple column Subquery in a FROM clause

```
SELECT ord_num, agent_code, ord_date, ord_amount
FROM orders o1
WHERE EXISTS(
 SELECT agent_code, ord_amount
 FROM orders o2
 WHERE  o1.agent_code = o2.agent_code
     AND o1.ord_amount = o2.ord_amount
)
ORDER BY ord_amount ASC
```



ord_num	agent_code	ord_date	ord_amount
200117	A001	2008-10-20	800
200112	A007	2008-05-30	2000
200230	A011	2008-07-30	2500
200105	A011	2008-07-18	2500
200113	A002	2008-06-10	4000

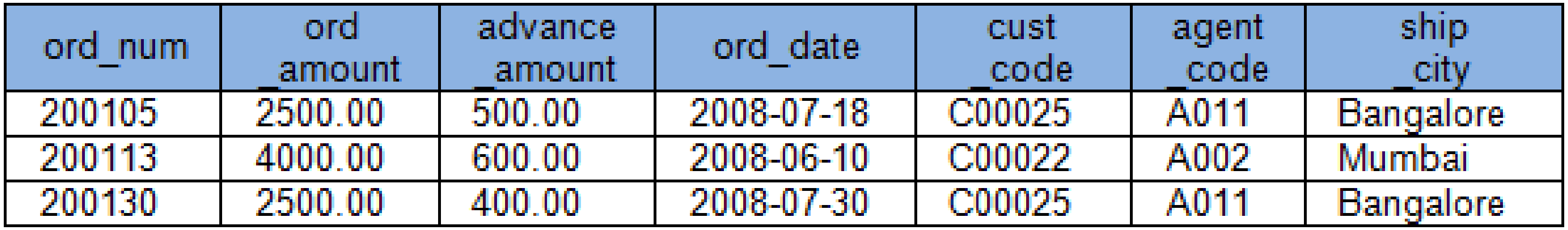
Correlated subquery

- Reference one or more columns in the outer SQL statement. The subquery is known as a correlated subquery because the subquery is related to the outer SQL statement.

- **Ex**: Correlated Subquery in a FROM clause

```
SELECT * FROM orders o
WHERE agent_code IN (
      SELECT agent_code FROM agents a
      WHERE o.ship_city = a.working_area
)
```

- **Result:**



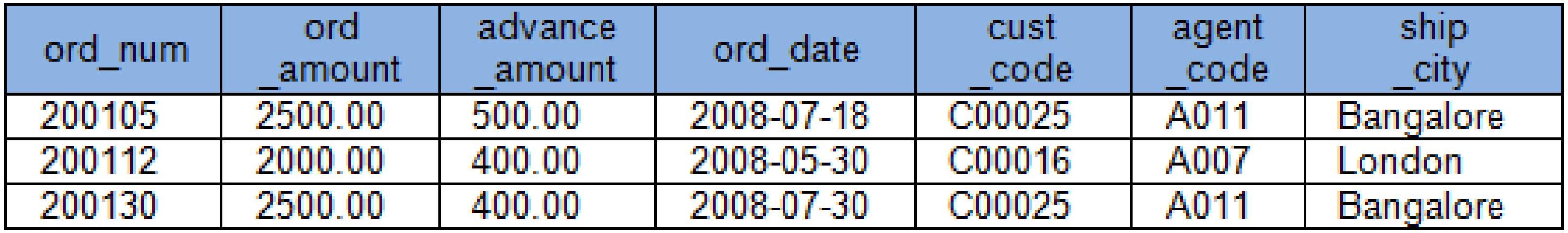
ord_num	ord_amount	advance_amount	ord_date	cust_code	agent_code	ship_city
200105	2500.00	500.00	2008-07-18	C00025	A011	Bangalore
200113	4000.00	600.00	2008-06-10	C00022	A002	Mumbai
200130	2500.00	400.00	2008-07-30	C00025	A011	Bangalore

Nested subquery

- A subquery can be nested inside other subqueries.
- **Ex**: Nested Subquery in a WHERE clause

```
SELECT *
FROM orders
WHERE ship_city IN (
          SELECT DISTINCT working_area FROM agents
          WHERE agent_code IN  (
                    SELECT agent_code FROM agents
                    WHERE commission >= 0.14
          )
)
```

- **Result:**



ord_num	ord_amount	advance_amount	ord_date	cust_code	agent_code	ship_city
200105	2500.00	500.00	2008-07-18	C00025	A011	Bangalore
200112	2000.00	400.00	2008-05-30	C00016	A007	London
200130	2500.00	400.00	2008-07-30	C00025	A011	Bangalore

Common caseS use subquery

- We focus on some typical usecases for Subquery:
  - ✓ *Subqueries with Aliases:* Many statements in which the subquery and the outer query refer to the same table
  - ✓ *Subqueries with IN / NOT IN:* The result of a subquery introduced with IN (or with NOT IN) is a list of zero or more values. After the subquery returns results, the outer query makes use of them
  - ✓ *Subqueries with EXISTS / NOT EXISTS:* The subquery functions as an existence test.
  - ✓ *Subqueries in UPDATE, DELETE, INSERT, SELECT*

Rules that subqueries must follow

- You must enclose a subquery in parenthesis.
- A subquery must include a SELECT clause and a FROM clause.
- Subqueries that return more than one row can only be used with multiple value operators, such as the IN operator.
- A subquery can include optional WHERE, GROUP BY, and HAVING clauses.
- A subquery cannot include COMPUTE or FOR BROWSE clauses.
- You can include an ORDER BY clause only when a TOP clause is included.
- You can nest subqueries up to 32 levels.

SUMMARY

- What is a subquery?
- Type of Subqueries
- Common cases use subquery
- Rules that subqueries must follow

Q&A
Thank you