IDS Lecture 15: Nested Queries 3 (Subqueries)

Examples with Exists / Not Exists

- universal quantification expressed with `NOT EXISTS` *Question 1. Customers living in a city without a branch (repeat question)?*

```
SELECT *
FROM Customer C
WHERE NOT EXISTS ( SELECT *
                   FROM Account A
                   WHERE A.brach = c.city ) ;
```

Scoping

A subquery has - a local scope (its `FROM` clause) - an outer scope

Question 2. Branches with a total balance (across accounts) of at least 500?

```
SELECT subquery.branch
FROM ( SELECT A.branch, SUM(A.balance) AS total
       FROM   Account A
       GROUP BY A.branch ) AS subquery
WHERE subquery.total >= 500 ;
```

Question 3: Average the total balances across each customer's accounts?

Strategy 1. find the total balance across each customers accounts 2. take the average of the totals

```
SELECT AVG(subquery.tot)
FROM   ( SELECT   A.custid, SUM(A.balance) AS tot
         FROM     Account A
         GROUP BY A.custid ) AS subquery ;
```

Ordering

Syntax: `ORDER BY` $\langle$column$_1\rangle$ `[DESC]` $,\ldots,$ $\langle$column$_n\rangle$ `[DESC]`

Example 1

```
SELECT *
FROM Accounts
ORDER BY custid ASC, balance DESC;
```

Casting

syntax: `CAST(`term `AS` $\langle$ type $\rangle$

Conditional Expressions

```
CASE WHEN (bool-exp)
     THEN (value-exp)
     ...
     WHEN (bool-exp)
     THEN (value-exp)
     ELSE (value-exp)
END
```

- ELSE is optional -> NULL if no match

Pattern Matching

Syntax: `term LIKE pattern`

Question 4: Customers with a name that begins with 'K' and has at least 5 characters?

```
SELECT *
FROM Customer
WHERE name LIKE 'K____%' ;
```