IE220
Introduction to Database Systems

Assist. Prof. Dr. Fırat YÜCEL
Akdeniz University
Informatics Department

Alias

- In queries, field names and table names can be renamed by **AS** statement.

```
SELECT C.CustomerID, S.Amount
FROM Customers AS C, Sales AS S
WHERE C.CustomerID = S.CustomerID;
```

- Another example

```
SELECT C.CustomerID AS ‘Müşteri Numarası’, S.Amount AS ‘Satış Tutarı’
FROM Customers AS C, Sales AS S
WHERE C.CustomerID = S.CustomerID;
```

Undetermined WHERE Statement & Using Asterisk

- If WHERE is not used, all records is returned.

  SELECT Name, Surname FROM Customers

- If asterisk is used after SELECT, all fields of records is returned.

  SELECT * FROM Customers

  Another example;

  SELECT * FROM Customers WHERE Name = ‘Ahmet’;

ALL & DISTINCT Statements

- To return repetitive data in a table **multiple times**

  SELECT ALL Name FROM Customers;

- *Using* ALL *statement is optional*.

- To return repetitive data in a table **only one time**

  SELECT DISTINCT Name FROM Customers;

Searching in a Text

- It is possible to search at beginning, middle and end of a text in a query by using LIKE comparing operator.
  - LIKE ‘ABC%’ Find the records contained ‘ABC’ at beginning of text.
  - LIKE ‘%ABC’ Find the records contained ‘ABC’ at end of text.
  - LIKE ‘%ABC%’ Find the records contained ‘ABC’ in text.
- An example;

```
SELECT * FROM Customers WHERE Name LIKE ‘%AL%’;
```

Arithmetical Operations

- It can be performed arithmetical operations in queries by using +, -, *, / operators.

```
SELECT ProductName AS 'Product Name',
       Quantity – 1 AS 'Remaining',
       Price * 1.18 AS 'Price incl. VAT'
FROM   Products;
```

BETWEEN Operator

- BETWEEN operator is used to define a range of numbers.

```
SELECT * FROM Orders
WHERE (Amount BETWEEN 3000 AND 4000) AND ProductName = 'HDD'
```

Amount is between 3000 and 4000.

Sorting of Results

- It is performed by ORDER BY statement. There are two choices:
  - ASC: Ascending Ordering (A-Z, 0-9)
  - DESC: Descending Ordering (Z-A, 9-0)

```
SELECT * FROM Customers ORDER BY Name ASC, Surname DESC;
```

Sort in ascending name, and then descending surname.

Basic Statistical Functions

- To find number of records, COUNT() function can be used.

  SELECT COUNT(*) FROM Students WHERE Department = ‘IE’;

- To find average of record values, AVG() function can be used.

  SELECT AVG(Grade) FROM Grades WHERE CourseID = ‘IE220’;

- To find sum of record values, SUM() function can be used.

  SELECT SUM(Amount) FROM Orders WHERE Category = ‘1’;

Grouping

- Grouping by a field; COUNT, SUM, MAX, MIN, AVG functions are retrieved data.

Students

StudentID	Name	Surname

Courses

CourseID	CourseName

Exams

StudentID	CourseID	Grade	Semester

- For example; average grades for each course in Courses table,

```
SELECT CourseName, AVG(Grade)
FROM Courses, Exams
WHERE Courses.CourseID = Exams.CourseID
GROUP BY CourseName;
```