

# Data Modeling and Databases

## Lab 3: Introduction to SQL

Bulat Gabbasov, Albina Sayfeeva

Innopolis University

2016

# Basic SQL query structure

- Basic SQL query structure consists of **SELECT**, **FROM**, **WHERE**, **GROUP BY** and **ORDER BY** clauses.
- **SELECT** [**ALL** | **DISTINCT**] *expressions*
  - specifies the columns to appear in the result
  - distinct keyword can be used to eliminate duplicates
- **FROM** *from items*
  - specifies the relations to be used
- **WHERE** *condition*
  - filters the tuples
- **GROUP BY** *expression*
  - groups rows with the same column values
  - the HAVING construct can be used to further filter the groups
- **ORDER BY** *expression*
  - defines the order of the resulting tuples

# Data manipulation

- **INSERT**

- Inserts a tuple into the specified table
- `INSERT INTO tablename (list of columns)  
VALUES (list of values), ...`

- **UPDATE**

- Updates all tuples that match specified condition
- `UPDATE tablename SET column = newvalue, ...  
WHERE condition`

- **DELETE**

- Deletes all tuples that match specified condition
- `DELETE FROM tablename WHERE condition`

# Inserting

- Create a new student Harvey Specter:

```
INSERT INTO students
(student_id, firstname, lastname) VALUES
(1, 'Harvey', 'Specter')
```

| <b>student_id</b><br>integer | <b>firstname</b><br>character varying(100) | <b>lastname</b><br>character varying(100) |
|------------------------------|--|---|
| 1                            | Harvey                                     | Specter                                   |

# Updating

Change firstname of all students having student\_id = 1 to 'John':

```
UPDATE students  
SET firstname = 'John'  
WHERE student_id = 1
```

| <b>student_id</b><br>integer | <b>firstname</b><br>character varying(100) | <b>lastname</b><br>character varying(100) |
|------------------------------|--|---|
| 1                            | John                                       | Specter                                   |

# Deleting

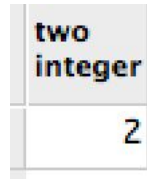
- Delete student having student\_id = 1 from table students:

```
DELETE FROM students WHERE student_id = 1
```

# Expressions

- Calculate expression  $1 + 1$  and name it as two:

```
SELECT 1 + 1 AS two
```



| two<br>integer |
|----------------|
| 2              |

# Tables

- Return list of all students:

```
SELECT * FROM students
```

| <b>student_id</b><br>integer | <b>firstname</b><br>character varying(100) | <b>lastname</b><br>character varying(100) | <b>address</b><br>character varying(100) | <b>gender</b><br>character(1) | <b>birthdate</b><br>date |
|------------------------------|--|---|--|-------------------------------|--------------------------|
| 1                            | Harvey                                     | Specter                                   | New York                                 | m                             | 1970-01-01               |
| 2                            | Michael                                    | Ross                                      | New York                                 | m                             | 1988-09-11               |
| 3                            | Rachel                                     | Zane                                      | Kazan                                    | f                             | 1989-06-23               |
| 4                            | Louis                                      | Litt                                      | Moscow                                   | m                             | 1973-07-04               |
| 5                            | Jessica                                    | Pearson                                   | Moscow                                   | f                             | 1975-08-25               |



# Exercise

- Insert a new department named 'Machine Learning' and led by professor identified by professor\_id = 1

```
INSERT INTO departments  
VALUES (4, 'Machine Learning', 1)
```

- Change name of the newly created department to 'Advanced Machine Learning'

```
UPDATE departments  
SET name = 'Advanced Machine Learning'  
WHERE name = 'Machine Learning'
```

- Delete new newly created department

```
DELETE FROM departments  
WHERE name = 'Advanced Machine Learning'
```

# Exercise

- Find the address of the student with first name "Donna"

```
SELECT address FROM students  
WHERE firstname = 'Donna'
```

|  |
|--|
| <b>address</b><br>character varying(100) |
| Kazan                                    |

# Exercise

- Find all students who are either male or are from Kazan

```
SELECT * FROM students
```

```
WHERE gender = 'm' or address = 'Kazan'
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 1                     | Harvey                              | Specter                            | New York                          | m                      | 1970-01-01        |
| 2                     | Michael                             | Ross                               | New York                          | m                      | 1988-09-11        |
| 3                     | Rachel                              | Zane                               | Kazan                             | f                      | 1989-06-23        |
| 4                     | Louie                               | Litt                               | Moscow                            | m                      | 1973-07-04        |

# Exercise

- Find all courses that worth at least 9 credits and are given by MSIT department
  - Hint: department\_id for MSIT-SE is 1.

```
SELECT * FROM courses
```

```
WHERE credits >= 9 AND department_id = 1
```

| course_id<br>integer | name<br>character varying(100) | credits<br>integer | professor_id<br>integer | department_id<br>integer |
|----------------------|--------------------------------|--------------------|-------------------------|--------------------------|
| 2                    | Models                         | 12                 | 4                       | 1                        |
| 3                    | Methods                        | 12                 | 5                       | 1                        |
| 4                    | Management                     | 9                  | 5                       | 1                        |
| 6                    | Analysis                       | 12                 | 5                       | 1                        |

# Exercise

- Find names and salaries of professors who earn less than 15 000

```
SELECT firstname, lastname, salary  
FROM professors WHERE salary < 15000
```

| firstname<br>character varying(100) | lastname<br>character varying(100) | salary<br>integer |
|-------------------------------------|------------------------------------|-------------------|
| Eric                                | Foreman                            | 10000             |
| Lisa                                | Cuddy                              | 5000              |

# Exercise

- Find students born earlier than 1980

```
SELECT * FROM students
```

```
WHERE birthdate < '1980-01-01'
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 1                     | Harvey                              | Specter                            | New York                          | m                      | 1970-01-01        |
| 4                     | Louis                               | Litt                               | Moscow                            | m                      | 1973-07-04        |
| 5                     | Jessica                             | Pearson                            | Moscow                            | f                      | 1975-08-25        |
| 8                     | Rose                                | Bukater                            | Pittsburgh                        | f                      | 1967-10-13        |

# Exercise

- List full names of all students living in Moscow
  - Hint: concatenation operator a || b

```
SELECT
```

```
    firstname || ' ' || lastname AS fullname  
, address  
FROM Students WHERE address = 'Moscow'
```

| fullname<br>text | address<br>character varying(100) |
|------------------|-----------------------------------|
| Louis Litt       | Moscow                            |
| Jessica Pearson  | Moscow                            |

# Exercise

- Find students who's address contains "k" letter

```
SELECT * FROM students WHERE address LIKE '%k%'
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 1                     | Harvey                              | Specter                            | New York                          | m                      | 1970-01-01        |
| 2                     | Michael                             | Ross                               | New York                          | m                      | 1988-09-11        |



# Exercise

- Find students whose lastname consists of 7 letters and ends with "n"

```
SELECT * FROM students
WHERE lastname LIKE '_____n'
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 1                     | Harvey                              | Specter                            | New York                          | m                      | 1970-01-01        |
| 2                     | Michael                             | Ross                               | New York                          | m                      | 1988-09-11        |

# Exercise

- Order and display students by lastname (alphabetically)

```
SELECT * FROM students  
ORDER BY lastname
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 9                     | Molly                               | Brown                              | Kazan                             | f                      | 1995-02-17        |
| 8                     | Rose                                | Bukater                            | Pittsburgh                        | f                      | 1967-10-13        |
| 7                     | Jack                                | Dawson                             | Kazan                             | m                      | 1985-10-17        |
| 10                    | Bruce                               | Ismay                              | Kazan                             | m                      | 1990-05-01        |
| 4                     | Louis                               | Litt                               | Moscow                            | m                      | 1973-07-04        |

# Exercise

- Order and display students by lastname and then by firstname (alphabetically)

```
SELECT * FROM students  
ORDER BY lastname, firstname
```

| student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| 9                     | Molly                               | Brown                              | Kazan                             | f                      | 1995-02-17        |
| 8                     | Rose                                | Bukater                            | Pittsburgh                        | f                      | 1967-10-13        |
| 7                     | Jack                                | Dawson                             | Kazan                             | m                      | 1985-10-17        |
| 10                    | Bruce                               | Ismay                              | Kazan                             | m                      | 1990-05-01        |
| 4                     | Louis                               | Litt                               | Moscow                            | m                      | 1973-07-04        |

# Exercise

- Order by login : first letter of firstname + full lastname in descending order
  - Hint: use SUBSTRING(column from begin for length)

```
SELECT SUBSTRING(firstname from 1 for 1)
       || lastname AS login, *
FROM students
ORDER BY 1 DESC
```

| login<br>text | student_id<br>integer | firstname<br>character varying(100) | lastname<br>character varying(100) | address<br>character varying(100) | gender<br>character(1) | birthdate<br>date |
|---------------|-----------------------|-------------------------------------|------------------------------------|-----------------------------------|------------------------|-------------------|
| RZane         | 3                     | Rachel                              | Zane                               | Kazan                             | f                      | 1989-06-23        |
| RBukater      | 8                     | Rose                                | Bukater                            | Pittsburgh                        | f                      | 1967-10-13        |
| MRoss         | 2                     | Michael                             | Ross                               | New York                          | m                      | 1988-09-11        |

# Exercise

- Find names of male students who got more than 50 for any course

```
SELECT s.firstname, s.lastname
FROM students s
WHERE gender = 'm' AND
EXISTS(SELECT 1 FROM enrollment e
WHERE e.student_id = s.student_id
AND e.grade > 50)
```

# Exercise

- Which students are enrolled in DMD course?

```
SELECT s.* FROM students s
NATURAL JOIN enrollment e
NATURAL JOIN courses c
WHERE c.name = 'DMD'
```

QA?