Lab work 1

Database Design. Introduction to SQL.

Full Name: Daniil Kalts.

Group: IT2-2404SE.

Goal: Perform a subject area analysis for a University Database.

Task:

1. **Describe at least 7 tables to store in the Database.**
2. **Describe attributes for each table in the Database.**
3. **Make sure that the Database has 5 constrained attributes.**
4. **Describe relations between entities in the Database.**
5. **Describe 2 access rights groups in the Database.**
6. **Provide 10-15 queries for the Database.**

---

1. What tables does the Database store?

1. The Database includes the following tables:

1. **faculties**
2. **deans**
3. **departments**
4. **programs**
5. **program_subject**
6. **subjects**
7. **contact_info**
8. **personal_info**
9. **students**
10. **teachers**
11. **teacher_subject**
12. **groups**
13. **buildings**
14. **classrooms**
15. **schedules**

16. **assignments**
17. **schedule_assignments**
18. **student_assignment**
19. **clubs**
20. **student_club**
21. **rector**

2. What attributes does the Database store?

2. The Database includes the attributes tables:

1. **faculties:**
    - faculty_id (PK)
    - name
    - description
2. **deans:**
    - dean_id (PK)
    - faculty_id (FK to faculties.faculty_id)
    - personal_info_id (FK to personal_info.personal_info_id)
3. **departments:**
    - department_id (PK)
    - faculty_id (FK to faculties.faculty_id)
    - name
    - description
4. **programs:**
    - program_id (PK)
    - department_id (FK to departments.department_id)
    - name
    - description
5. **program_subject:**
    - program_id (FK to programs.program_id)
    - subject_id (FK to subjects.subject_id)
    - PRIMARY KEY (program_id, subject_id)
6. **subjects:**
    - subject_id (PK)
    - name
    - description
    - credits
7. **contact_info:**
    - contact_info_id (PK)
    - email

- phone_number
- address

8. **personal_info:**
    - personal_info_id (PK)
    - first_name
    - last_name
    - patronymic
    - gender
    - birth_date
    - nationality
    - citizenship
    - iin
    - education
    - contact_info_id (FK to contact_info.contact_info_id)
9. **students:**
    - student_id (PK)
    - group_id (FK to groups.group_id)
    - personal_info_id (FK to personal_info.personal_info_id)
10. **teachers:**
    - teacher_id (PK)
    - personal_info_id (FK to personal_info.personal_info_id)
11. **teacher_subject:**
    - teacher_id (FK to teachers.teacher_id)
    - subject_id (FK to subjects.subject_id)
    - PRIMARY KEY (teacher_id, subject_id)
12. **groups:**
    - group_id (PK)
    - head_student_id (FK to students.student_id)
    - name
13. **buildings:**
    - building_id (PK)
    - name
    - description
    - address
    - floor_count
14. **classrooms:**
    - classroom_id (PK)
    - building_id (FK to buildings.building_id)
    - name
    - description
    - floor_number
    - capacity
15. **schedules:**
    - schedule_id (PK)

- group_id (FK to groups.group_id)
- subject_id (FK to subjects.subject_id)
- teacher_id (FK to teachers.teacher_id)
- classroom_id (FK to classrooms.classroom_id)
- date
- weekday
- start_time
- end_time

16. **assignments:**
    - assignment_id (PK)
    - subject_id (FK to subjects.subject_id)
    - name
    - description
    - deadline
17. **schedule_assignments:**
    - schedule_id (FK to schedules.schedule_id)
    - assignment_id (FK to assignments.assignment_id)
    - PRIMARY KEY (schedule_id, assignment_id)
18. **student_assignment:**
    - student_id (FK to students.student_id)
    - assignment_id (FK to assignments.assignment_id)
    - score
    - PRIMARY KEY (student_id, assignment_id)
19. **clubs:**
    - club_id (PK)
    - name
    - description
20. **student_club:**
    - student_id (FK to students.student_id)
    - club_id (FK to clubs.club_id)
    - PRIMARY KEY (student_id, club_id)
21. **rector:**
    - rector_id (PK)
    - personal_info_id (FK to personal_info.personal_info_id)

3. What constrained attributes does the Database have?

3. Constrained attributes of each table:

1. **faculties:**
    - **name:** 5-100 characters.

- **description:** 20-500 characters.

2. **deans:**
    - **personal_info_id:** Valid foreign key to personal_info.
    - **faculty_id:** Valid foreign key to faculties.
3. **departments:**
    - **name:** 5-100 characters.
    - **description:** 20-500 characters.
    - **faculty_id:** Valid foreign key to faculties.
4. **programs:**
    - **name:** 5-100 characters.
    - **description:** 20-500 characters.
    - **department_id:** Valid foreign key to departments.
5. **program_subject:**
    - **program_id:** Valid foreign key to programs.
    - **subject_id:** Valid foreign key to subjects.
6. **subjects:**
    - **name:** 5-50 characters.
    - **description:** 20-250 characters.
    - **credits:** Must be between 1 and 10.
7. **contact_info:**
    - **email:** Must be in valid email format.
    - **phone_number:** Must follow the Kazakhstan format: +7 followed by 10 digits.
    - **address:** 10-100 characters.
8. **personal_info:**
    - **first_name:** 2-100 characters.
    - **last_name:** 2-100 characters.
    - **patronymic:** 2-100 characters.
    - **gender:** Must be either MALE, FEMALE, or OTHER.
    - **birth_date:** Valid date; age must be between 17 and 70.
    - **iin:** Exactly 12 digits.
    - **education:** 20-200 characters.
    - **contact_info_id:** Valid foreign key to contact_info.
9. **students:**
    - **group_id:** Valid foreign key to groups.
    - **personal_info_id:** Valid foreign key to personal_info.
10. **teachers:**
    - **personal_info_id:** Valid foreign key to personal_info.
11. **teacher_subject:**
    - **teacher_id:** Valid foreign key to teachers.
    - **subject_id:** Valid foreign key to subjects.
12. **groups:**
    - **name:** 6-20 characters.
    - **head_student_id:** Valid foreign key to students.
13. **buildings:**

- **name:** 5-100 characters.
- **description:** 20-500 characters.
- **address:** 10-100 characters.
- **floor_count:** Must be between 1 and 50.

14. **classrooms:**
    - **name:** 5-100 characters.
    - **description:** 20-200 characters.
    - **floor_number:** Must be between 1 and the total number of floors in the building.
    - **capacity:** Must be between 5 and 100.
15. **schedules:**
    - **group_id:** Valid foreign key to groups.
    - **subject_id:** Valid foreign key to subjects.
    - **teacher_id:** Valid foreign key to teachers.
    - **classroom_id:** Valid foreign key to classrooms.
    - **date:** Must be a valid date.
    - **weekday:** Must be one of the days Monday to Sunday.
    - **start_time:** Must be in valid 24-hour time format (HH:MM:SS).
    - **end_time:** Must be in valid 24-hour time format (HH:MM:SS) and must occur after **start_time** on the same day.
16. **assignments:**
    - **name:** 5-40 characters.
    - **description:** 15-100 characters.
    - **deadline:** Must be a valid date.
    - **subject_id:** Valid foreign key to subjects.
17. **schedule_assignments:**
    - **schedule_id:** Valid foreign key to schedules.
    - **assignment_id:** Valid foreign key to assignments.
18. **student_assignment:**
    - **student_id:** Valid foreign key to students.
    - **assignment_id:** Valid foreign key to assignments.
    - **score:** Must be between 0 and 100.
19. **clubs:**
    - **name:** 5-100 characters.
    - **description:** 20-500 characters.
20. **student_club:**
    - **student_id:** Valid foreign key to students.
    - **club_id:** Valid foreign key to clubs.
21. **rector:**
    - **personal_info_id:** Valid foreign key to personal_info.

4. What relations between tables does the Database have?

4. Relations between tables:

1. **faculties:**
    - **faculties - departments** (one-to-many)
    - **faculties - deans** (one-to-one)
2. **deans:**
    - **deans - faculties** (one-to-one)
    - **deans - personal_info** (one-to-one)
3. **departments:**
    - **departments - programs** (one-to-many)
4. **programs:**
    - **programs - program_subject** (one-to-many)
    - **programs - subjects** (many-to-many through program_subject)
5. **program_subject:**
    - **program_subject - programs** (many-to-one)
    - **program_subject - subjects** (many-to-one)
6. **subjects:**
    - **subjects - program_subject** (one-to-many)
    - **subjects - assignments** (one-to-many)
7. **contact_info:**
    - **contact_info - personal_info** (one-to-one)
8. **personal_info:**
    - **personal_info - students** (one-to-one)
    - **personal_info - teachers** (one-to-one)
    - **personal_info - deans** (one-to-one)
9. **students:**
    - **students - groups** (many-to-one)
    - **students - clubs** (many-to-many through student_club)
    - **students - assignments** (many-to-many through student_assignment)
    - **students - personal_info** (one-to-one)
10. **teachers:**
    - **teachers - schedules** (one-to-many)
    - **teachers - subjects** (many-to-many through teacher_subject)
    - **teachers - personal_info** (one-to-one)
11. **teacher_subject:**
    - **teacher_subject - teachers** (many-to-one)
    - **teacher_subject - subjects** (many-to-one)
12. **deans:**
    - **deans - faculties** (one-to-one)
    - **deans - personal_info** (one-to-one)
13. **groups:**
    - **groups - students** (one-to-many)
    - **groups - schedules** (one-to-many)
14. **buildings:**

    - **buildings - classrooms** (one-to-many)
15. **classrooms:**
    - **classrooms - schedules** (one-to-many)
    - **classrooms - buildings** (many-to-one)
16. **schedules:**
    - **schedules - schedule_assignments** (one-to-many)
    - **schedules - classrooms** (many-to-one)
    - **schedules - teachers** (many-to-one)
    - **schedules - groups** (many-to-one)
17. **assignments:**
    - **assignments - schedule_assignments** (one-to-many)
    - **assignments - student_assignment** (one-to-many)
    - **assignments - subjects** (many-to-one)
18. **student_assignment:**
    - **student_assignment - students** (many-to-one)
    - **student_assignment - assignments** (many-to-one)
19. **clubs:**
    - **clubs - students** (many-to-many through student_club)
20. **student_club:**
    - **student_club - students** (many-to-one)
    - **student_club - clubs** (many-to-one)
21. **rector:**
    - **rector - personal_info:** (one-to-one)

5. What access rights groups does the Database have?

5. Access rights groups of the Database:

- **User Group 1:** Students
- **User Group 2:** Teachers
- **User Group 3:** Deans
- **User Group 4:** Rector

---

1. **faculties:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - rw (read-write for faculties they manage)
    - Rector - rw (read-write)
2. **deans:**
    - Students - ro (read-only)

- Teachers - ro (read-only)
- Deans - rw (read-write for their own data)
- Rector - ro (read-only)

3. **departments:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - rw (read-write for departments related to their faculties)
    - Rector - rw (read-write)
4. **programs:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - rw (read-write for programs related to their departments)
    - Rector - rw (read-write)
5. **program_subject:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - rw (read-write for subjects related to their programs)
    - Rector - rw (read-write)
6. **subjects:**
    - Students - ro (read-only)
    - Teachers - rw (read-write for subjects they teach)
    - Deans - rw (read-write for subjects related to their faculties)
    - Rector - rw (read-write)
7. **contact_info:**
    - Students - rw (read-write for their own contact data)
    - Teachers - rw (read-write for their own contact data)
    - Deans - rw (read-write for their own contact data)
    - Rector - rw (read-write for their own contact data)
8. **personal_info:**
    - Students - rw (read-write for their own data)
    - Teachers - rw (read-write for their own data)
    - Deans - rw (read-write for their own data)
    - Rector - rw (read-write for their own data)
9. **students:**
    - Students - rw (read-write for their own data)
    - Teachers - ro (read-only)
    - Deans - ro (read-only)
    - Rector - ro (read-only)
10. **teachers:**
    - Students - ro (read-only)
    - Teachers - rw (read-write for their own data)
    - Deans - ro (read-only)
    - Rector - ro (read-only)
11. **teacher_subject:**

- Students - ro (read-only)
- Teachers - rw (read-write for subjects they teach)
- Deans - rw (read-write for subjects related to their faculties)
- Rector - rw (read-write)

12. **groups:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - rw (read-write for groups related to their faculties)
    - Rector - rw (read-write)
13. **buildings:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - ro (read-only)
    - Rector - rw (read-write)
14. **classrooms:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - ro (read-only)
    - Rector - rw (read-write)
15. **schedules:**
    - Students - ro (read-only)
    - Teachers - rw (read-write for schedules related to their subjects)
    - Deans - rw (read-write for schedules related to their faculties)
    - Rector - rw (read-write)
16. **assignments:**
    - Students - ro (read-only)
    - Teachers - rw (read-write for assignments they manage)
    - Deans - rw (read-write for assignments related to their faculties)
    - Rector - rw (read-write)
17. **schedule_assignments:**
    - Students - ro (read-only)
    - Teachers - rw (read-write for assignments related to their schedules)
    - Deans - rw (read-write for assignments related to their faculties)
    - Rector - rw (read-write)
18. **student_assignment:**
    - Students - rw (read-write for their own data)
    - Teachers - rw (read-write for assignments they manage)
    - Deans - rw (read-write for assignments related to their faculties)
    - Rector - rw (read-write)
19. **clubs:**
    - Students - ro (read-only)
    - Teachers - ro (read-only)
    - Deans - ro (read-only)
    - Rector - rw (read-write)

20. **student_club:**
    - Students - rw (read-write for their own data)
    - Teachers - ro (read-only)
    - Deans - ro (read-only)
    - Rector - rw (read-write)
21. **rector:**
    - Rector - rw (read-write for their own data)

6. What are potential queries for the Database?

6. The Database may have the following queries:

1. **List all faculties along with their departments.**
2. **List the number of students in each program.**
3. **Find the subject with the highest average student score.**
4. **List all teachers assigned to the Database Design subject.**
5. **List all teachers who are teaching more than 3 subjects.**
6. **Find the teacher with the highest number of students across all their classes.**
7. **List all groups with their corresponding head students.**
8. **List all classrooms that have a capacity greater than 30.**
9. **List all assignments due next week.**
10. **List all assignments related to a specific subject.**
11. **List all students who have completed all assignments in a subject.**
12. **List the average score of students for each assignment.**
13. **List all students with a GPA below 50%.**
14. **List all students who have not joined any clubs.**

Thank you for your time!