Conceptual Answers of Weak 3 Assigment

Submited by: Shivam Gupta

Part A

Q: Explain why you choosed the data types for `student_id`, `email`, and `marks_obtained`.

- **student_id:**
  I choosed `INT` for `student_id` becuase in most colleges and universties, student roll numbers or enrolement numbers are numeric. Using `INT` ensure efficiant storage and fast retreival for these unique identifers.

- **email:**
  I used `VARCHAR` for the `email` feild becaus email addresses can contain alphabets, numbers, and spacial characters (like `@`, `.`, etc.). `VARCHAR` is perfact for such mixed strings and also provides dynamic memmory alocation, saveing space for shorter emails.

- **marks_obtained:**
  For `marks_obtained`, I selcted `INT` as marks are always numeric values. Useing an integer data type ensure that we can efficently perform mathematical operations and comparisions on marks data.

Part C

1. List all students enrolled in "Database Managment Systems" along with their emails.

```sql
SELECT s.student_id, s.name, s.email, c.title
FROM Students s
JOIN Enrollments e ON s.student_id = e.student_id
JOIN Courses c ON e.course_id = c.course_id
WHERE c.title = 'Database Management Systems';
```

Smaple Output:

student_id	name	email	title
1	Shivam Gupta	shivam.gupta@email.com	Database Management Systems
2	Aarav Sharma	aarav.sharma@email.com	Database Management Systems
3	Priya Verma	priya.verma@email.com	Database Management Systems

Explenation:
This query finds al students who are enroled in the course "Database Management Systems". Acorrding to the sample data, students with IDs 1, 2, and 3 (Shivam Gupta, Aarav Sharma, and Priya Verma) are enrolled in this course. Thier email addreses are also shown in the result.

2. Show the avrage marks obtained in each course.

```
SELECT c.title,
       ROUND(AVG(g.marks_obtained),2) AS average_marks
FROM Courses c
JOIN Enrollments e ON c.course_id = e.course_id
JOIN Grades g ON e.enrollment_id = g.enrollment_id
GROUP BY c.course_id, c.title;
```

Sample Output:

title	average_marks
Database Management Systems	61.00
Data Structures and Algorithms	54.33
Operating Systems	81.00
Computer Networks	85.00

Explenation:
This query calculates the avrage marks recived by students for each course. For instence, the average marks for "Database Management Systems" is 61.00. This helps to get a overview of how students are performig in diferent courses, based on the given grades in the databse.

Part D

Why are transactions important in real systems such as banking, education platforms, or e-comerce?

Answar:

Transactions are criticaly important in real-world systems becuase they insure the reliablity and integrity of data when mutiple operations need to be perfomed as a single logical unit. They achieve this by providing the follwing ACID propertis:

- **Atomicity:** Gaurentees that all operations within a transaction are compeleted; if any operation fails, the entier transaction is rolled back.
  Example: In banking, during a tranfer, the debit and credit must both occure together. If one fails, neithar should be applyed.

- **Consistency:** Ensures that the databse moves from one vailid state to anothar.
  Example: In an education platform, when a student enrolls in a course and makes a payment, the enrollment shold only be succesful if the payment is succesful too.

- **Isolation:** Prevents concurrent transactions from interfering with each other and causing data anomalys.
  Example: In e-comerce, simultaneus orders for the same item shouldn't allow the item count to become negative.

- **Durability:** Once a transaction is commited, its changes persit even in the event of a system falure. *Example: A completed purchese in an onlne store should never dissapear, even if the platform crases right afterward.*

Summery:
Transactions ensure the corectness and trustworthyness of critical operations in banking, education, and e-commerce systems, where data erors or inconsistencies can lead to finacial loses, user dissatisfacton, or secuirty breaches.

Part E

Q1: Explain the diference between DELETE and TRUNCATE.

Answar:

Both `DELETE` and `TRUNCATE` are used to remove data from a table, but they work a little differently:

- `DELETE` lets you remove specific rows from a table. You can chose which rows to delete by using a condition. For exmple, you can delete just one student from the student list. `DELETE` is also a bitt slower becuse it removes rows one at a time.
    - *Example:*
      `DELETE FROM Students WHERE student_id = 5;` (This deletes only the student with ID 5.)
- `TRUNCATE` quickly removes **all** rows from a table. You can't chose specific rows—it wipes the table cleen! It's usally much faster than `DELETE` for clearing a table. It also usaly resets things like auto-increment counters.
    - *Example:*
      `TRUNCATE TABLE Students;` (This deletes every student from the table—all at once.)

In short:
Use `DELETE` when you want to remove speciffic data; use `TRUNCATE` when you want to quickly empty the hole table.

Q2: Why are forgien keys important for data integraty? Give a real-world example.

Answar:

Foreign keys make sure that relashionships between tables stay acurate. They prevent data that doesn’t match from being entered, keeping your data consitent.

- *Example:*
  If there’s an `Enrollments` table that links students to courses, a foreign key ensures you can’t enroll a student who dosen't exist in the `Students` table.

Q3: What is the advatage of normalization in this schema?

Answar:

Normalization keeps the data tidy and avoids repetion. In this schema, each peice of info—like students or courses—is stored just once, so there’s no confsion or extra coppies to update. This makes the databse more acurate and easyer to manage.