Quiz 10

True or False (1 pt each)

1. In Discretionary Access Control, the owner of a resource has the authority to determine who can access that resource and what permissions they have.
2. Role-Based Access Control assigns permissions directly to individual users rather than to roles or job functions.
3. Discretionary Access Control is generally considered more scalable and easier to manage in large organizations compared to Role-Based Access Control because it allows each user to control their own resources independently.
4. Hashing is typically applied to encrypting credit card information.
5. A salt is a randomly generated string that is often added to each password before hashing. The same salt is typically used for all data that needs to be hashed.
6. The same key is used for both encryption and decryption for symmetric encryption. The key values are typically stored along with encrypted data in the same table.
7. The CVV/CVC values of credit cards are typically not stored in databases at all.
8. When a column is indexed, an extra data structure is built to speed up searching, sorting, and ordering of the indexed column.
9. When a column needs to be indexed, a full-text index is needed as long as its data type is string (e.g., CHAR or VARCHAR).
10. Denormalization is often considered the first step to optimize a database design.

Short Answer (2 pts each)

1. Think about a database for an online bookstore. Someone designed a table named books in this database. books is supposed to store the information of books as multiple attributes, including:

- title - `VARCHAR(200)`
- ISBN - `CHAR(13)`

- publisher - `VARCHAR(255)`
- description - `VARCHAR(500)`
- categories - `TEXT`
- authors - `TEXT`

The table has only one candidate/primary key - ISBN, and all columns are dependent on ISBN. However, as testing data is used to populate the table, it looks as follows:

Title	ISBN	Publisher	Description	Categories	Authors
The Art of Computer Programming	9780201896831	Addison-Wesley	A comprehensive monograph written by Donald Knuth that covers many kinds of programming algorithms.	Computer Science, Programming, Algorithms	Donald E. Knuth
Design Patterns: Elements of Reusable Object-Oriented Software	9780201633612	Addison-Wesley	Describes simple and elegant solutions to specific problems in object-oriented software design.	Computer Science Programming Software Engineering	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
Clean Code: A Handbook of Agile Software Craftsmanship	9780132350884	Prentice Hall	A handbook that teaches the principles of writing clean, readable, and maintainable code.	Programming, Software Development, Agile	Robert C. Martin
Introduction to Algorithms	9780262033848	MIT Press	A comprehensive textbook covering the full spectrum of modern algorithms.	Computer Science,Algorithms,Mathematics	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein
The Pragmatic Programmer	9780201616224	Addison-Wesley	Your Journey to Mastery. A guide to becoming a better programmer.	Programming Software Development Career	David Thomas,Andrew Hunt
Structure and Interpretation of Computer Programs	9780262510875	MIT Press	A textbook aiming to teach the principles of computer programming.	Computer Science, Programming, LISP	Harold Abelson, Gerald Jay Sussman, Julie Sussman
Refactoring: Improving the Design of Existing Code	9780201485677	Addison-Wesley	A book about improving the design of existing code without changing its functionality.	Programming,Software Engineering,Code Quality	Martin Fowler, Kent Beck, John Brant, William Opdyke, Don Roberts
You Don't Know JS: Up & Going	9781491924464	O'Reilly Media	First book in the You Don't Know JS series about JavaScript fundamentals.	JavaScript, Web Development, Programming	Kyle Simpson

Does this table violate any normal forms? If so, how do you propose to fix the issues?

Note: All information should be retained in one or more tables. If your fix involves breaking the original design into multiple tables, please identify the primary key of each table.

2. Think about a database for a course system of a higher education institution. Someone designed a table named course_enrollment in this database. course_enrollment is supposed to store the information of books as multiple attributes, including:

- **enrollment_id:** Primary key (auto-increment)
- **student_id:** Foreign key to student table
- **course_code:** Course identifier (e.g., CS101, MATH200)
- **semester:** Semester when course is taken (e.g., Fall2023, Spring2024)
- **course_title:** Name of the course
- **credits:** Number of credits for the course
- **instructor_name:** Name of the instructor teaching the course
- **instructor_email:** Instructor's email address
- **grade:** Grade received by the student
- **enrollment_date:** Date when student enrolled

There are two valid candidate keys in the table:

- **enrollment_id**
- **student_id, course_code, semester**

enrollment_id is picked as the primary key. Beyond that, there are two other known functional dependencies:

1. course_code → course_title, credits
2. (course_code, semester) → instructor_name, instructor_email

When the table is populated with test data, it looks as follows:

enrollment_id	student_id	course_code	semester	course_title	credits	instructor_name	instructor_email	grade	enrollment_date
1001	S001	CS101	Fall2023	Introduction to Computer Science	3	Dr. Smith	smith@university.edu	A	2023-08-15
1002	S002	CS101	Fall2023	Introduction to Computer Science	3	Dr. Smith	smith@university.edu	B+	2023-08-16
1003	S001	MATH200	Fall2023	Calculus II	4	Dr. Johnson	johnson@university.edu	A-	2023-08-17
1004	S003	CS101	Spring2024	Introduction to Computer Science	3	Dr. Williams	williams@university.edu	B	2024-01-10
1005	S002	MATH200	Spring2024	Calculus II	4	Dr. Johnson	johnson@university.edu	B+	2024-01-12
1006	S004	ENG101	Fall2023	English Composition	3	Prof. Davis	davis@university.edu	A	2023-08-18
1007	S005	CS101	Fall2023	Introduction to Computer Science	3	Dr. Smith	smith@university.edu	C+	2023-08-19
1008	S001	ENG101	Spring2024	English Composition	3	Prof. Miller	miller@university.edu	A-	2024-01-15

Does this table violate any normal forms? If so, how do you propose to fix the issues?

Note: All information should be retained in one or more tables. If your fix involves breaking the original design into multiple tables, please identify the primary key of each table.