

Final Exam

MULTIPLE CHOICE QUESTIONS

Please select the correct answer to each question. (1 point each)

Question 1. What Python collection takes unique elements?

- a)
- b)
- c)
- d) Set

Question 2. What JOIN process is always reversible?

- a)
- b)
- c) Outer Join
- d)

Question 3. What SQL operator is applied after aggregation?

- a)
- b) HAVING
- c)
- d)

Question 4. What Python operator can take an "else:" block ?

- a)
- b)
- c)
- d) all three

Question 5. What part of a Django project manages the routing of the project?

- a)
- b) urls.py
- c)
- d)

SHORT ANSWER QUESTIONS

Please determine the truth of the statements below. You should explain why any of the statements is wrong then correct it.

Question 6. In PostgreSQL, we can store numbers in the format of 1234.56 in a NUMERIC(4,2) data type. False. It should be NUMERIC(6,2) because the first argument controls total number of digits and the second one controls the precision.

Question 7. The select operator (σ) in relational algebra, works the same as the SELECT command in PostgreSQL. False. SELECT deals with columns but (σ) deals with conditions to return rows.

Question 8. Django engine does not process (.html) files and shows them as is. False. Django engine looks for certain tags in HTML files and replaces them with data to provide the page content.

Question 9. ER diagrams show the tables that exist in a database. False. ER diagram shows the entities and relationships in the design process. They may change during implementation.

Question 10. Minimal candidate keys are called superkeys in database theory. False. Minimal superkeys are called candidate keys not the other way around.

Question 11. Many-to-Many relationships can involve more than two entity sets. False. All of these cardinality attributes such as many-to-one, many-to-many, ect. only involve two entity sets.

COMPREHENSIVE QUESTIONS

Please answer following questions in detail.

Question 12. What are the main components of the MVT design pattern, and how can using it help us develop a solution. (8 points)

- M stands for the model and it interacts with the database
- V stands for the view and it contains the functionalities (business logic) of a project.
- T stands for the template and it includes the representation layer of a web project.

Using MVT design pattern makes it easier to scale up and manage the web projects.

Question 13. What is the advantage of using normal forms in a schema design, and why would we decide to violate a normal form? (8 points) Using normal forms prevents redundancy. As a result the database would take less storage, and its consistency and stability would be maintained easier. However, sometimes we prefer violating the normal form to persist a functional dependency or to avoid performing multiple joins on frequent queries. That will make our database more responsive, and reduce the execution time of certain queries.

Question 14. Based on the table below, mention one lossless decomposition and one lossy decomposition. For each decomposition, you should try to rebuild the initial table to prove your point. (6 points) Fig. 1

A	B	C
a1	b1	c1
a1	b2	c2

TABLE 1. Table of $R=(A, B, C)$

shows a lossless decomposition and Fig. 2 shows a lossy decomposition. As you can see, it is possible to reconstruct the relation from only from a lossless decomposition.

Question 15. Explain how we perform aggregation in MongoDB mentioning three of its operators. You do not need to write precise syntax. (8 points) There is a *aggregate* function that takes an array of operators and executes those elements by order. We have *group*, *match*, and *unwind* as three of the main components of MongoDB aggregation. We can determine the attributes of aggregation via *group*, check for conditions before or after the aggregation using *match*, and extract the inner elements of an array using *unwind*.

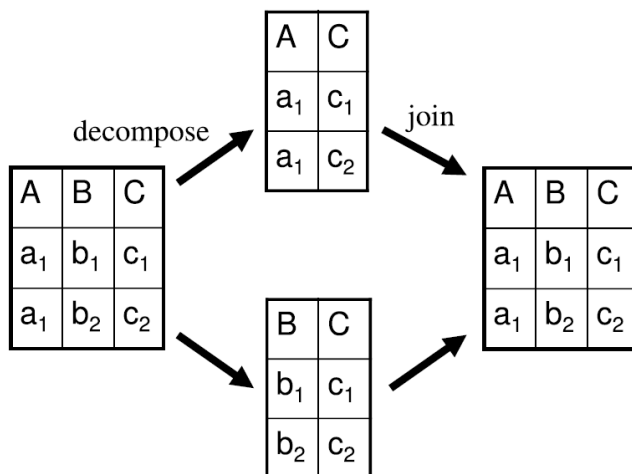


FIGURE 1. lossless decomposition.

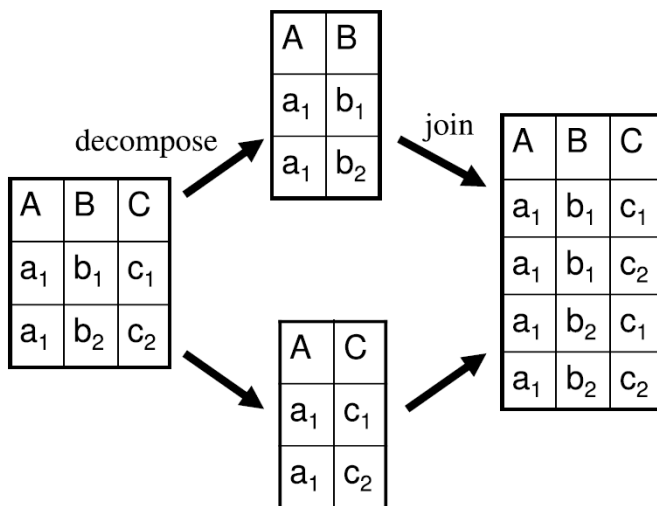


FIGURE 2. lossy decomposition.