## Types of Architecture:

1-Tier Architecture:

- What is it? In this setup, the user interacts directly with the database.
- How does it work? Any changes or requests are made directly to the database.
- Use case: It’s often used for local applications during development, where programmers need quick communication with the database.

2-Tier Architecture:

- What is it? This is like a basic client-server model.
- How does it work? Applications on the user’s computer directly communicate with the database on the server side. APIs like ODBC or JDBC are used for this communication.
- Responsibilities: The user’s machine handles the interface and application, while the server manages database functionalities like queries and transactions.

3-Tier Architecture:

- What is it? This involves an extra layer between the user and the server.
- How does it work? The user’s application communicates with an application server, which then communicates with the database. The user has no direct contact with the database, and the database is unaware of individual users.
- Use case: It’s often used for large web applications where separation between the user interface, application logic, and database management is crucial.

Visual Explanation:

- 1-Tier: User talks directly to the database.sqlCopy code`User <-----> Database`
- 2-Tier: User's application communicates with the database.arduinoCopy code`User <----> Application (Client) <-----> Database (Server)`
- 3-Tier: User's application talks to an application server, which then communicates with the database.rustCopy code`User <-----> Application (Client) <-----> Application Server <-----> Database`

## Related posts:

1. What is database management system (DBMS) ? What are the tasks performed by users in DBMS ?
2. What are the advantages and disadvantages of DBMS ?

3. What do you understand by database users ? Describe the different types of database users.
4. Who are data administrators ? What are the functions of database administrator ?OR Discuss the role of database adinistrator.
5. What is data abstraction ? Explain different levels ofabstraction.
6. Explain the differences between physical level,conceptual level and view level of data abstraction.
7. Explain the difference between database management system (DBMS) and file system.
8. What are data models ? Briefly explain different types of data models.
9. Describe data schema and instances.
10. Describe data independence with its types
11. Describe the classification of database language. Which type of language is SQL ?
12. Explain DBMS interfaces. What are the various DBMS interfaces ?
13. What is ER model ? What are the elements of ER model ? What are the notations of ER diagram ?
14. What do you understand by attributes and domain ?Explain various types of attributes used in conceptual data model.
15. Construct an ER diagram for University system.
16. Construct an ER diagram for the registrar’s office
17. Explain the primary key, super key, foreign key and candidate key with example. OR Define key. Explain various types of keys.
18. What do you mean by a key to the relation ? Explain the differences between super key, candidate key and primary key.
19. Explain generalization, specialization and aggregation. OR Compare generalization, specialization and aggregation with suitable examples.
20. What is Unified Modeling Language ? Explain different types of UML.
21. What is relational model ? Explain with example.

22. Explain constraints and its types.
23. Consider the following relations:
24. What are the additional operations in relational algebra ?
25. Explain integrity constraints.
26. Explain the following constraints : i. Entity integrity constraint. ii. Referential integrity constraint. iii. Domain constraint.
27. Describe mapping constraints with its types.
28. Explain how a database is modified in SQL. OR Explain database modification.
29. Discuss join and types with suitable example. Define join. Explain different types of join.
30. Describe the SQL set operations