1. What is Hibernate?

Interview answer:
“Hibernate is an ORM framework that maps Java objects to database tables and handles database operations automatically, reducing boilerplate JDBC code.”

---

2. Why Hibernate instead of JDBC?

Interview answer:
“Hibernate reduces boilerplate code, handles object–table mapping, manages transactions, supports caching, and makes applications database-independent.”

---

3. What is ORM?

Interview answer:
“ORM stands for Object Relational Mapping, where Java classes are mapped to database tables and objects are mapped to rows.”

---

4. What are the main components of Hibernate?

Interview answer:
“The main components are Configuration, SessionFactory, Session, Transaction, and Query.”

---

5. What is SessionFactory?

Interview answer:
“SessionFactory is a heavyweight, thread-safe object created once per application. It is used to create Session objects.”

---

6. What is Session?

Interview answer:
“Session is a lightweight, non-thread-safe object used to perform CRUD operations. It represents a single unit of work with the database.”

---

7. Difference between SessionFactory and Session?

Interview answer:
“SessionFactory is created once and shared across the application, while Session is created per request or transaction.”

8. What is Transaction in Hibernate?

Interview answer:
“Transaction represents a unit of work. It ensures data consistency using commit and rollback operations.”

---

9. What are the states of an object in Hibernate?

Interview answer:
“There are three main states:
Transient – object not associated with session
Persistent – object associated with session
Detached – object disconnected from session”

---

10. What is persistent state?

Interview answer:
“In persistent state, changes made to the object are automatically synchronized with the database.”

---

11. What is dirty checking?

Interview answer:
“Dirty checking is a mechanism where Hibernate automatically detects changes in persistent objects and updates the database without explicit update calls.”

---

12. What is first-level cache?

Interview answer:
“First-level cache is the session-level cache enabled by default. It stores objects within a session to avoid repeated database hits.”

---

13. What is second-level cache?

Interview answer:
“Second-level cache is optional and shared across sessions. It improves performance by reducing database access.”

---

14. Difference between first-level and second-level cache?

Interview answer:
“First-level cache is session-scoped and mandatory, while second-level cache is session-factory-scoped and optional.”

---

15. What is HQL?

Interview answer:
“HQL is Hibernate Query Language. It works on entities and properties, not on tables and columns.”

---

16. Difference between HQL and SQL?

Interview answer:
“HQL is database-independent and works with objects, while SQL is database-specific and works with tables.”

---

17. What is Criteria API?

Interview answer:
“Criteria API is used to build queries dynamically in a programmatic and type-safe way.”

---

18. Difference between `get()` and `load()`?

Interview answer:
“get() hits the database immediately and returns null if data is not found, while load() returns a proxy and throws exception if object doesn’t exist.”

---

19. What is lazy loading?

Interview answer:
“Lazy loading means data is fetched only when it is actually required, improving performance.”

---

20. What is eager loading?

Interview answer:
“Eager loading fetches associated data immediately along with the parent object.”

---

21. Difference between lazy and eager fetching?

Interview answer:
“Lazy loading improves performance by loading data on demand, while eager loading loads everything upfront and may impact performance.”

---

22. What is N+1 select problem?

Interview answer:
“N+1 problem occurs when Hibernate executes one query for parent records and N queries for child records, leading to performance issues.”

---

23. How do you solve N+1 select problem?

Interview answer:
“By using fetch joins, entity graphs, or changing fetch type appropriately.”

---

24. What is cascade?

Interview answer:
“Cascade defines how operations like save, update, or delete are propagated from parent to child entities.”

---

25. Common cascade types?

Interview answer:
“ALL, PERSIST, MERGE, REMOVE, REFRESH, and DETACH.”

---

26. Difference between save(), persist(), and saveOrUpdate()?

Interview answer:
“save() returns generated ID, persist() does not return ID and is JPA standard, saveOrUpdate() decides whether to insert or update.”

---

27. What is flush()?

Interview answer:
“flush() synchronizes session changes with the database but does not commit the transaction.”

---

28. What is clear() and evict()?

Interview answer:
“clear() removes all objects from session cache, while evict() removes a specific object.”

---

29. How do you handle one-to-many relationship?

Interview answer:
“Using @OneToMany and @ManyToOne annotations with proper mapping and cascade options.”

---

30. Real-time use of Hibernate in projects?

Interview answer:
“In real projects, Hibernate is used for CRUD operations, relationship mapping, pagination, caching, and transaction management.”