1. What is JPA?

Answer: JPA is a Java specification for accessing, persisting, and managing data between

Java objects/classes and a relational database. It's like a set of rules or guidelines

that Java developers can follow to interact with a database.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2. What are the advantages of JPA?

Answer: Advantages of JPA (Java Persistence API) include:

\* Object-Relational Mapping (ORM):

Facilitates mapping Java objects to database tables, eliminating the need for manual SQL.

\* Database Independence:

Enables database portability, allowing the application to work with different database systems.

\* Reduced Boilerplate Code:

Simplifies database operations, reducing the amount of repetitive code needed for CRUD operations.

\* Automatic Table Creation:

Can automatically generate database tables based on entity classes, easing database setup.

\* Query Language (JPQL):

Provides a powerful and object-oriented query language for database interactions.

\* Caching Mechanism:

Supports caching to improve performance by reducing database hits for frequently accessed data.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

3. What are the Differentiate Between Hibernate And Jpa?

Answer: JPA (Java Persistence API) is a specification for ORM (Object-Relational Mapping)

in Java, while Hibernate is an implementation of JPA.

In other words, JPA provides a standard set of interfaces and annotations for ORM,

while Hibernate is a concrete implementation of those interfaces and annotations.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4. Define entity ?

Answer: In JPA, an entity is a lightweight Java class that represents a persistent data object. Entities

are used to map Java objects to database tables, where each entity corresponds to a row in the table.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5. What Is An Entitymanager?

Answer: An EntityManager in Java Persistence API (JPA) is an interface that manages the

lifecycle of entities in a JPA application. It is responsible for persisting, merging, removing,

and querying entities in a relational database. The EntityManager acts as a bridge between

the application and the underlying database, providing a set of methods to interact with

persistent entities.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

6. What is JPQL and how is it used in JPA?

Answer: JPQL (Java Persistence Query Language) is a query language in Java used with the

Java Persistence API (JPA) to interact with databases. It provides a way to write database

queries in a platform-independent manner, allowing developers to query entities (objects)

rather than database tables.

Usage in JPA:

\* Developers write JPQL queries to retrieve, update, or delete entities stored in a relational database.

\* JPQL queries resemble SQL but focus on entities and their relationships rather than database tables.

\* JPQL queries are translated by the JPA provider into SQL queries specific to the underlying database.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

7. What is ORM Framework and how is JPA related to that?

Answer: An Object-Relational Mapping (ORM) framework is a software tool that allows

developers to map object-oriented programming language constructs to relational database

constructs. It provides a layer of abstraction between the application code and the database,

allowing developers to work with objects and classes rather than SQL queries.

JPA (Java Persistence API) is a Java EE standard that provides an ORM framework for

mapping Java objects to relational databases. It defines a set of interfaces and annotations

that allow developers to create persistent entities, query data, and manage relationships between entities.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8. What are some benefits of using an ORM framework like JPA?

Answer: Some of them are:

\* Increased Productivity:

- JPA provides a high level of abstraction that allows developers to focus on business logic

instead of writing SQL queries. This can lead to faster development cycles and fewer errors.

\* Portability:

- JPA abstracts away the details of the underlying database, which makes it possible to switch

databases without changing the application code. This can save a lot of time and effort when

porting applications between different databases.

\* Scalability:

- JPA provides a caching mechanism that can help improve application performance by reducing

the number of database queries needed to access data. This can help an application scale better

as the number of users and amount of data grows.

\* Maintainability:

- JPA provides a clear separation between application logic and persistence logic. This makes it

easier to maintain and modify an application over time.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9. What is Hibernate?

Answer: Hibernate is an open-source, object-relational mapping (ORM) framework for

Java. It simplifies linking object-oriented models to relational databases.

It simplifies the database interaction in Java applications by allowing

developers to work with Java objects directly.

Hibernate Query Language (HQL) is an object-oriented query language that

is similar to SQL but operates on the objects directly