1. **Что такое транзакция?**

Транзакция — это набор операций по работе с базой данных (БД), объединенных в одну атомарную пачку.

**2. В чем её отличие JPA от Hibernate?**

Hibernate одна из самых популярных открытых реализаций последней версии спецификации (JPA 2.1). Даже скорее самая популярная, почти стандарт де-факто. То есть JPA только описывает правила и API, а Hibernate реализует эти описания, впрочем, у Hibernate (как и у многих других реализаций JPA) есть дополнительные возможности, не описанные в JPA (и не переносимые на другие реализации JPA).

**3. Что такое Entity?**

**Entity** – это сущность которая является отображением в базе данных. Связь между Entity – это зависимость одной сущности от другой. Очень часто используются они в построении больших БД.

**4. Конфигурационный файл Hibernate?**

Hibernate требует заранее знать – где найти информацию о сопоставлении, которая определяет, как ваши классы Java связаны с таблицами базы данных. Hibernate также требует набора параметров конфигурации, связанных с базой данных и других связанных параметров. Вся такая информация обычно предоставляется в виде стандартного файла свойств Java с именем **hibernate.properties** или в виде файла XML с именем **hibernate.cfg.xml** .

**5. Аннотация @Autowired?**

Аннотацию @Autowired можно описать следующим образом Используя эту аннотацию, не нужно заботиться о том, как лучше всего передать классу или bean'у экземпляр другого bean'a. Фреймворк Spring сам найдет нужный bean и подставит его значение в свойство, которое отмечено аннотацией @Autowired.

Свойства класса с аннотацией @Autowired заполняются соответствующими значениями сразу после создания bean'а и перед тем, как любой из методов класса будет вызван.

**6. Что такое JPQL (Java Persistence query language)?**

Java Persistence Query Language (**JPQL**) — платформо-независимый объектно-ориентированный язык запросов являющийся частью спецификации JPA. JPQL используется для написания запросов к сущностям, хранящимся в реляционной базе данных.

**7. Интерфейс JpaRepository?**

**JpaRepository –** это интерфейс фреймворка Spring Data предоставляющий набор стандартных методов JPA для работы с БД. На основе этого интерфейса Spring Data предоставит реализации с методами, которые моггут быть использованы.  
8. Аннотация @Id, @GeneratedValue?

Аннотации @Id и @GeneratedValue предоставляют возожность не выставлять никакие параметры генерации и значения по умолчанию отработают нормально и в поле будет присваиваться уникальное значение каждый раз при сохранении.

Аннотация @Id наследуется от javax.persistence.Id, указывая, что поле члена ниже является первичным ключом текущего объекта. Следовательно, ваш Hibernate и spring рамки, а также вы можете сделать некоторые reflect работы, основанные на этой аннотации.

Аннотация @GeneratedValue заключается в настройке способа увеличения указанного столбца (поля). Например, при использовании Mysql вы можете указать auto\_increment в определении таблицы, чтобы сделать его самоинкрементным, а затем использовать в коде Java, чтобы обозначить, что вы также признали, что используете эту стратегию на стороне сервера базы данных. Кроме того, вы можете изменить значение в этой аннотации, чтобы соответствовать различным требованиям.

**9. Как проекте Maven подключить JPA?**

Нажимаем на название проекта правой кнопкой и выбираем Add Framework support. Отмечаем галочкой Spring, JavaEE Persistence и Hibernate. Если зависимости подгрузились, то будет предложено выбрать библиотеку maven. После этого в проект добавятся необходимые xml — META-INF\persistence.xml и sping-config.xml.

**10. Методы JpaRepository.**

<S extends T> S save(S entity); – сохраняет переданную сущность.

Optional<T> findById(ID primaryKey); – возвращает сущность, идентифицированную данным ID.

Iterable<T> findAll(); – возвращает все сущности.

long count(); – возвращает количество сущностей.

void delete(T entity); – удаляет переданную сущность.

boolean existsById(ID primaryKey); – указывает, существует ли сущность с данным ID.

*flush() –* производит запись сохраненных данных непосредственно в БД.