Md Rahman

1. What is Dependency Injection?

2. What is Dagger?

3. What if the difference in Dagger 1 vs Dagger 2?

4. What is a dependency graph?

5. What are the following annotations for in dagger 2:

Inject

Component

Module

Provides

6. What are the main types of dependency injection and what is the difference in them?

1. What is Dependency Injection?

Dependency Injection is a means of decoupling classes from what they depend on. This is important because it offers an inversion of control, which means that classes should configure its dependencies from the outside. Ultimately, it results in more reusable, independent classes and for easier testing. Dependencies can be injected thru constructors, getters and setter methods.

2. What is Dagger?

A dependency injection framework developed by Google for Java, Kotlin and Android. It is a compile time framework that uses no reflection. Dagger creates a dependency graph at build.

3. Dagger 1 vs Dagger 2?

Advantages of Dagger 2: No more reflection or runtime graph composition (improves performance). It also supports method injection, not just field and constructor injection. Modules are also easier to configure than in Dagger 1.

Advantages of Dagger 1: Inject() method has strong type association with injection target. We must now rebuild the entire project to see any new component implementation. Does not support overrides.

4. What is a dependency graph?

A component creates a dependency graph, which is a visual representation of the dependencies and the order they should be created

5. What are the following annotations in Dagger 2?

Inject - Inserting the dependency into an object

Component - Grouping of modules. Creates a dependency graph so it knows exactly which order to create the dependencies.

Module - Dependencies instantiated

Provides - Define classes and methods which provide dependencies

6. What are the main types of dependency injection and what is the difference in them?

Setter Injection - Most popular, injects the dependency via a setter method.

Constructor Injection - Inject dependency via a constructor.

Interface injection - Dependency provides an injector method that injects the dependency into whatever client was passed to it. The client must then implement an interface that exposes a setter method that accepts the dependency.