**Autowiring:**

Autowiring is a feature in Spring that allows Spring to automatically resolve and inject collaborating beans into your bean. It reduces the need for explicit property or constructor arguments in the bean definition. Spring supports several autowiring modes, each with its own use case.

**Autowiring Modes**

* **no (default):** No autowiring. Dependencies must be explicitly defined.
* **byName:** Autowires by property name. Spring will look for a bean with the same name as the property to be autowired.
* **byType:** Autowires by property type. Spring will look for a bean of the same type as the property to be autowired.
* **constructor:** Autowires by constructor. Spring will look for beans that match the constructor parameter types.
* **autodetect:** Deprecated as of Spring 3.0. It was a combination of constructor and byType. Spring first attempts to autowire using the constructor, then by type if no constructor is found.

**No Autowiring (default)**

<bean id="userService" class="com.example.UserService">

<property name="userRepository" ref="userRepository" />

</bean>

<bean id="userRepository" class="com.example.UserRepository" />

**Autowire by Name**

<bean id="userService" class="com.example.UserService" autowire="byName" />

<bean id="userRepository" class="com.example.UserRepository" />

**Autowire by Type**

<bean id="userService" class="com.example.UserService" autowire="byType" />

<bean id="userRepository" class="com.example.UserRepository" />

**Autowire by Constructor**

<bean id="userService" class="com.example.UserService" autowire="constructor" />

<bean id="userRepository" class="com.example.UserRepository" />