**Bean Scopes**

* Scope refers to the lifecycle of a bean
* How long does the bean live?
* How many instances are created?
* How is the bean shared?

The Default scope of a bean is singleton

**What is Singleton?**

* Spring container creates only one instance of the bean, by default
* It is cached in memory
* All requests for the bean will return a SHARED reference to the SAME beanA screenshot of a cell phone
  
  Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

* Prototype scope is good for keeping stateful data.

**Bean Lifecycle**

A close up of a sign

Description automatically generated

**Bean Lifecycle Methods / Hooks**

* You can add custom code during **Bean initialization**
* Calling custom business logic methods
* Setting up handles to resource (db, sockets, file etc)
* You can also add custom code during **bean destruction**

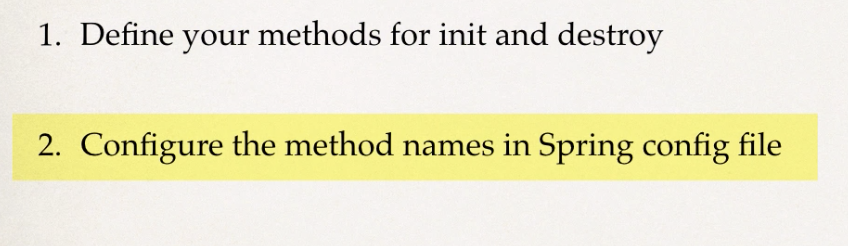
**INIT: method configuration**

A screenshot of a cell phone

Description automatically generated

**Destroy: Method configuration**

A screenshot of a cell phone

Description automatically generated

**Development Process:**

****

* These methods can have access modifiers
* Void is the most commonly used return type.
* The method should be a no-arg
* For the **prototype** scoped beans, Spring doesn’t call the destroy method, because spring doesn’t manage the complete lifecycle of the prototype bean.