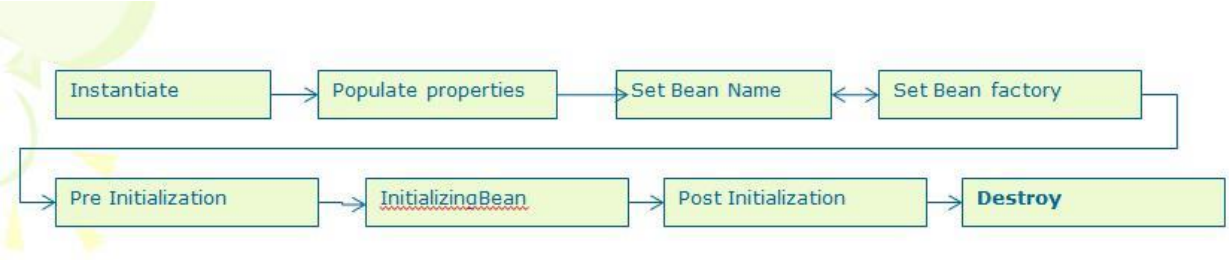
Spring Bean life cycle :



Spring's bean has a very elaborated life cycle which gives opportunities to hook up some custom code or logic in various life cycle stages. There are two distinct spring containers one is bean factory and another is application context. More precisely, only one additional phase is added in case of application context.

1. **Instantiate:** in this phase container finds the bean's definition and instantiates it.
2. **Populate properties:** in this phase using the dependency injection all the properties are populated which are specified in the bean definiton (in the xml file).
3. **Set Bean Name :** If BeanNameAware interface is implemented the factory calls the setBeanName() passing the Bean's Id.
4. **Set Bean factory:** if BeanFactoryAware interface is implemented setBeanFactory() method is called.
5. **ApplicationContextAware:** This step is valid only if the Application context container is used. In this phase if bean implements the ApplicationcontextAware then setApplicationContext() method is called.
6. **Pre-initialization:** If any BeanPostProcessors are associated with the bean then their postProcessBeforeInitialization() methods will be called.

7. **InitializingBean:** If an init-methodis specified for the bean then it is called.
8. **Post Initialization:** If there are any BeanPostProcessors are associated then postProcessAfterInitialization() is called.

9. D**estroy:** If bean implements the DisposableBean interface then destroy() method is called.