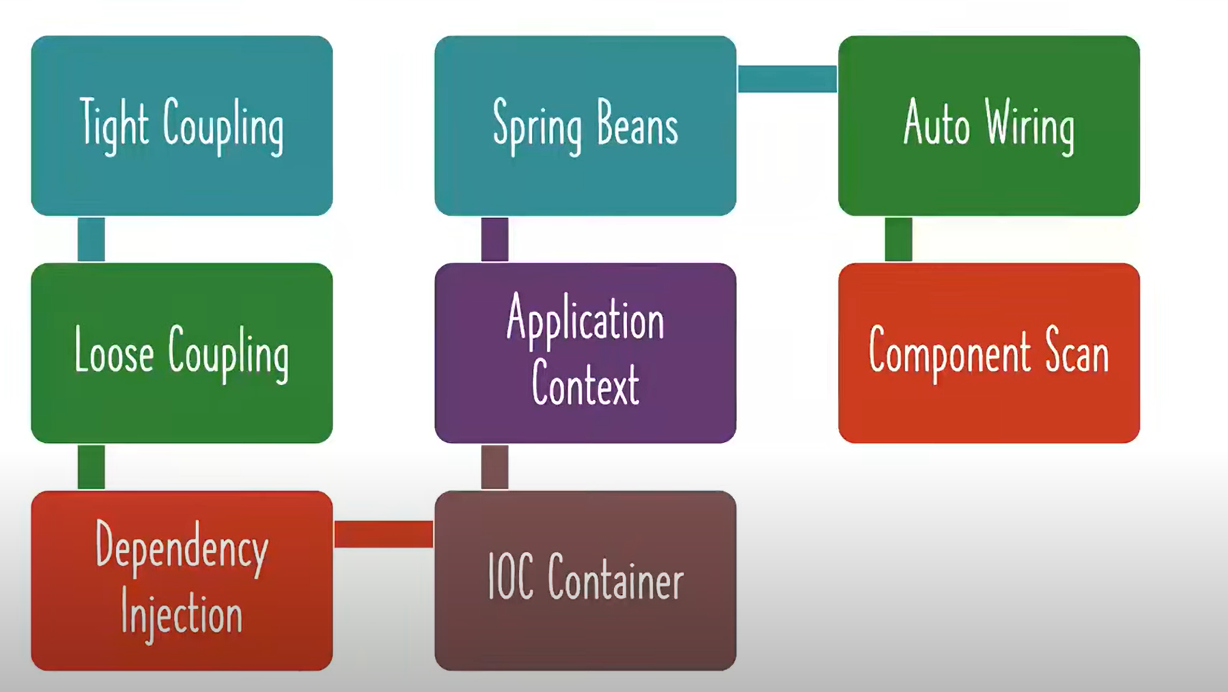
**This course is more about Spring , which is the base of SpringBoot**

**Url**:

<https://www.youtube.com/watch?v=f6DHAgL7FWc>

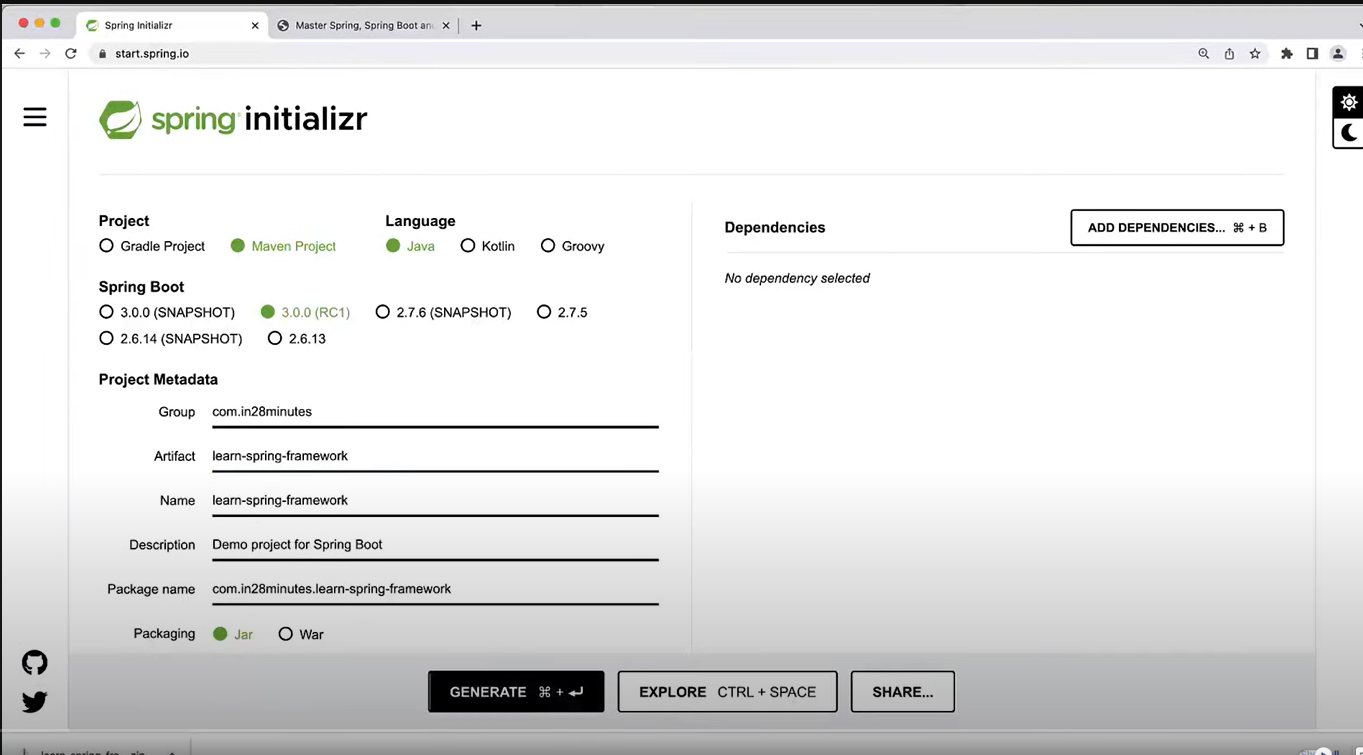
This are some of the concepts that we will learn:



|  |  |
| --- | --- |
|  | We are going to work this project , apparently there are different levels of coupling |

I stayed that i needed to install java 17, so I will install openJdk which is the open source reference implementation for jdk so it is just the source code , but apparently this is just the code or binaries so to install it i need a prebuilt , so a builds created from OpenJdk its call a OpenJdk binary, so for that is AdoptOpenJdk(there are others vendors like correto, azul etc) , but on top of that there is temurin who takes the prebuilt and test it and enhance it

1st we are going to create a project in spring initializer:

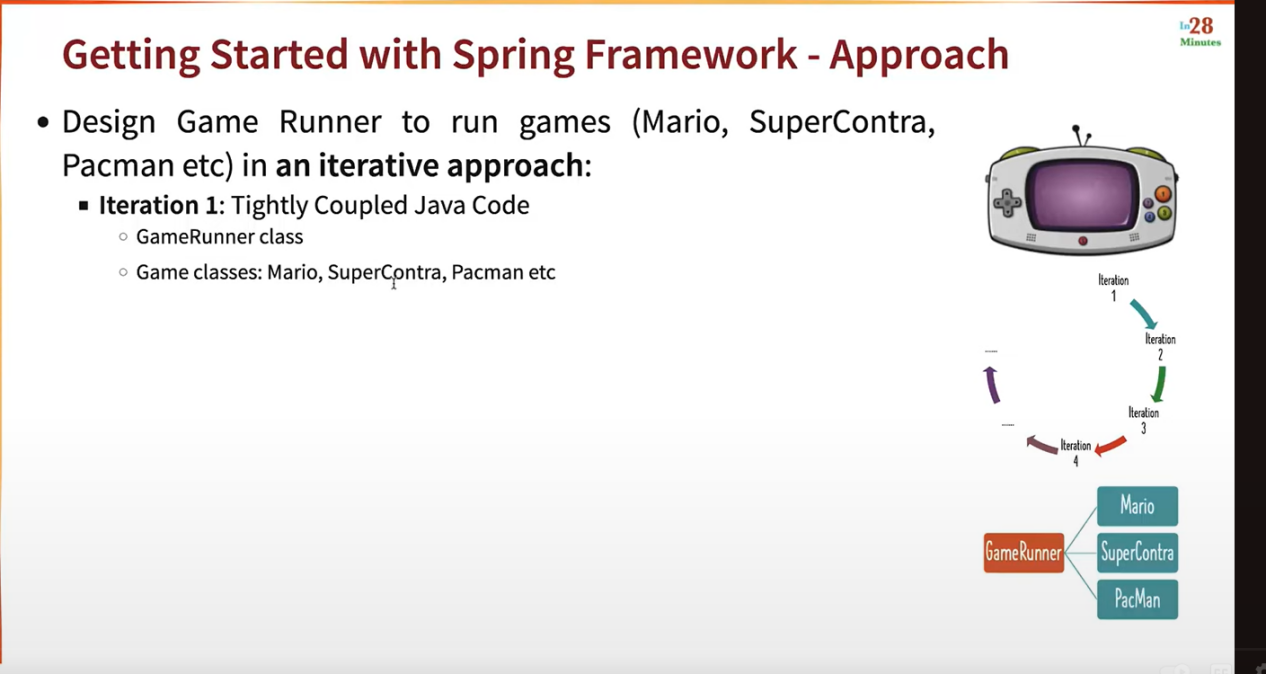


Some notes when you create a project here with the initializer:

- do not choose a snapshot spring boot version because they are currently under development and are not stable

- the group may be seen as the package name, and the artifactId as the class name

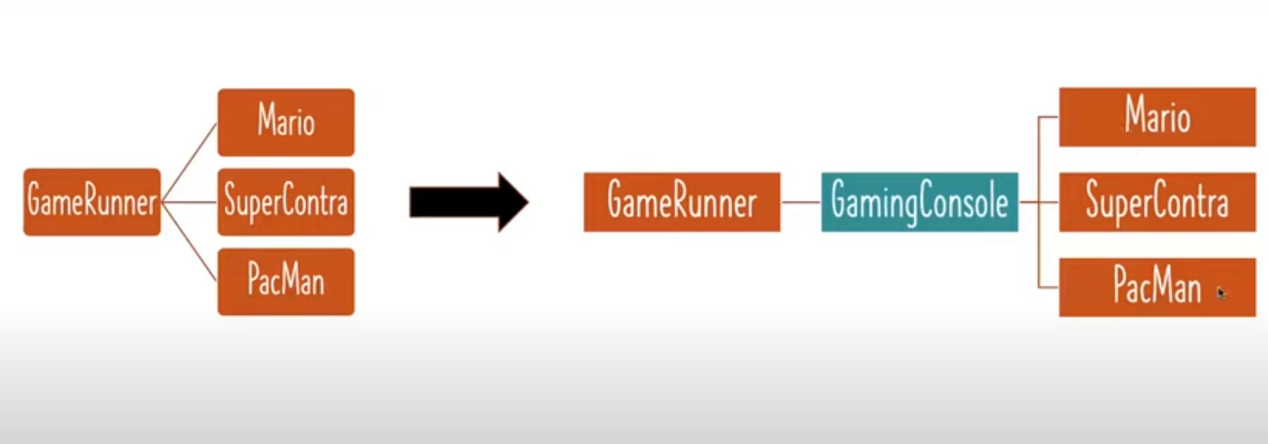
1. Tightly coupled java code: this is the case when is **tightly coupled,** this mean fuertemente acoplado, this mean that components are strongly interconnected and they have a high dependency on each other, the changes in a components will have a high impact in the others component



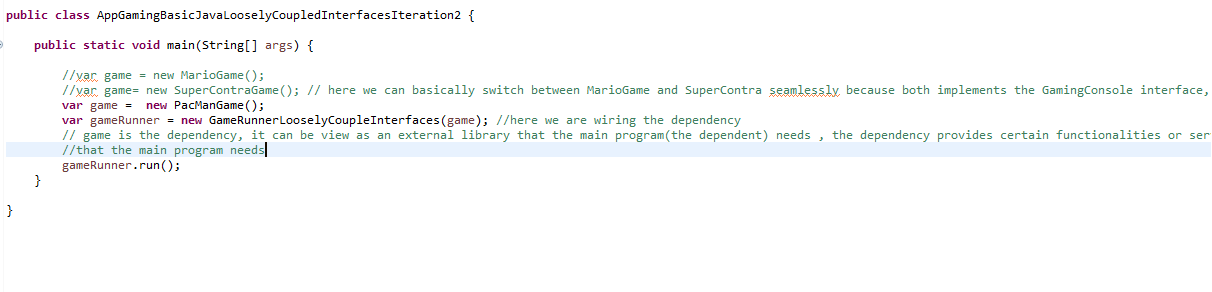
I stayed in the minute 21:50, I added documentation in the eclipse project

**Coupling**: is how much work is involved to change something

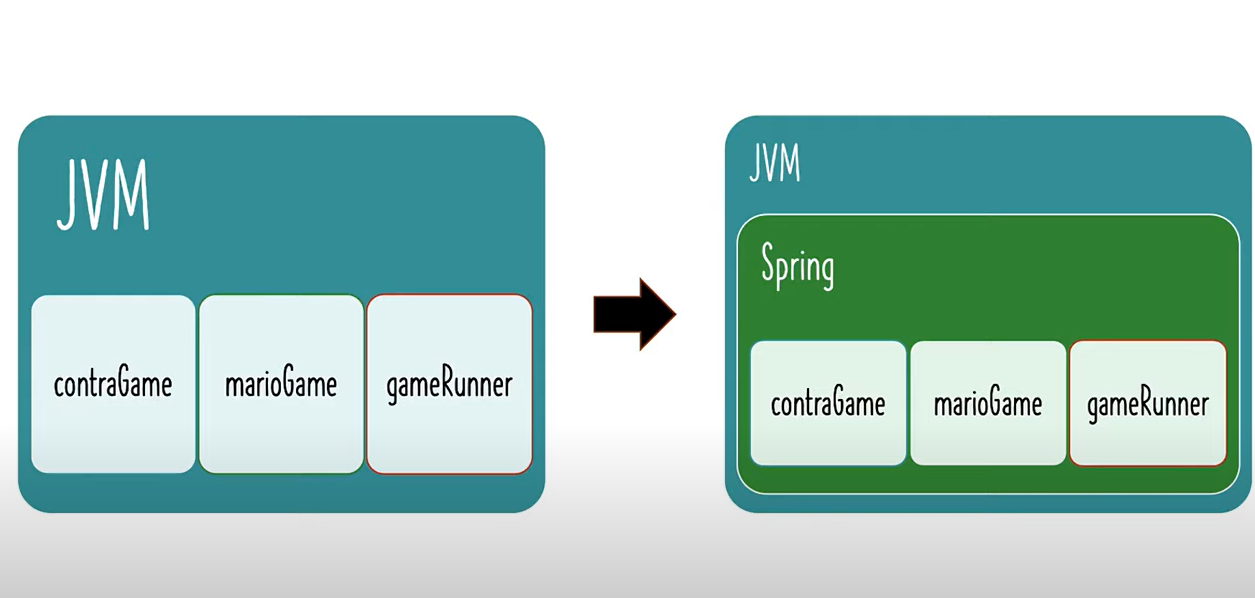
**Iteration 2**: loosely coupled with interfaces:



Now you see the games Mario, SuperContra etc are not directly connected to the GameRunner but now they are connected throug the GamingConsoleInterface , so any change in the games does not affect directly on the Runner.

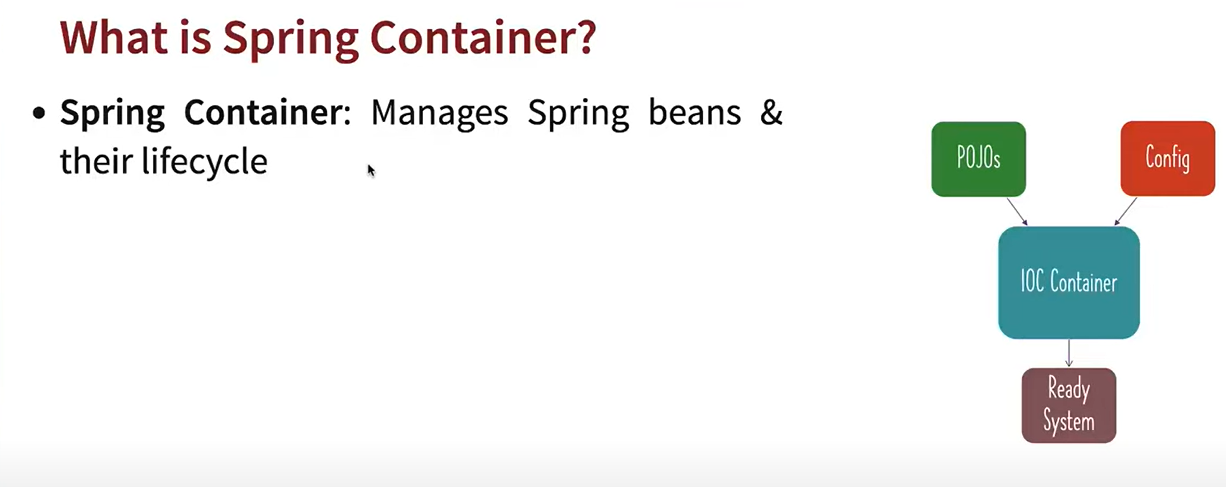


At this level we are creating the objects and wiring it ourserlves in the code, but in the next step what we wan is Spring wiring and managing the beans for us

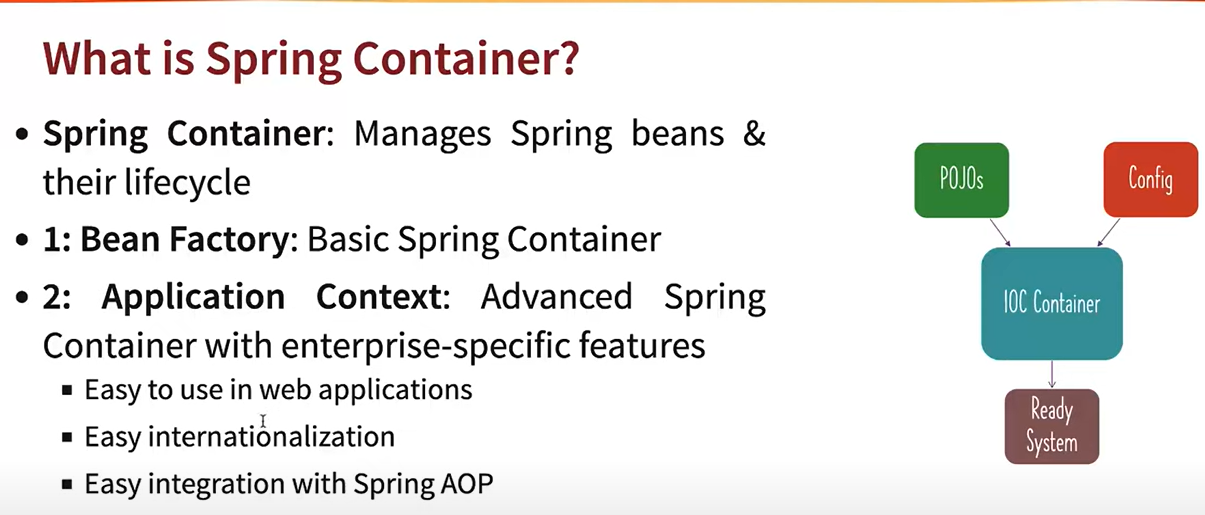


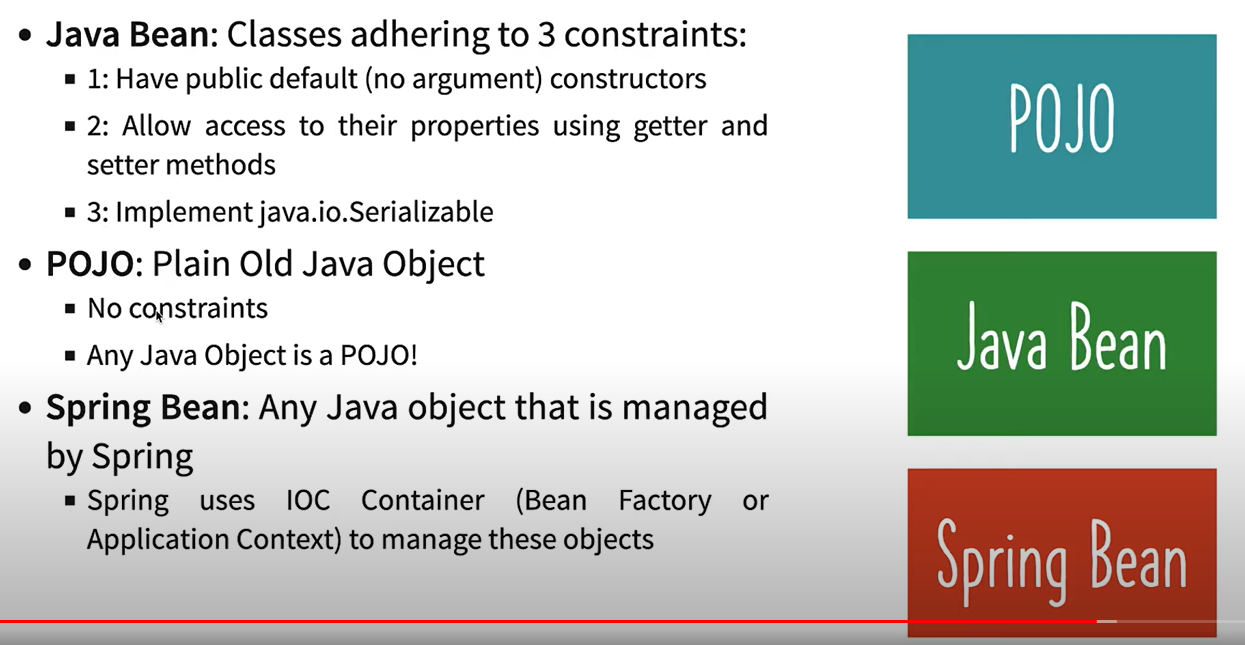
What is a spring bean: are the things that are managed by spring

the spring context is also called ioc container, spring container, application context , this all is refering to the same thing that takes some pojos and a config class and gives you a ready system



There are two types of spring containers, the bean factory is pretty basic, the other one application context allows more features , this is the one I already know, because is the enterprise spring containers , in which you can build microservices , web services etc.





siamti