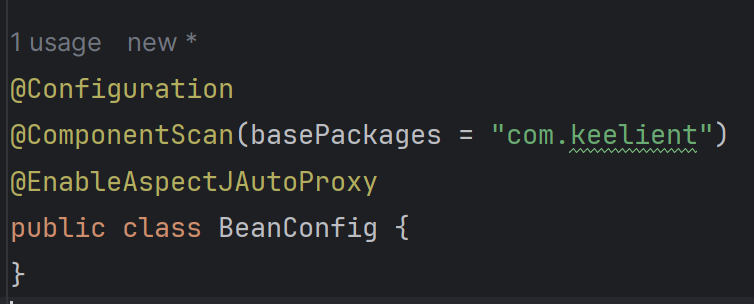
# Spring AOP

Aspect-oriented programming (AOP) is a programming paradigm that aims to modularize cross-cutting concerns, which are functionalities that affect multiple parts of a software application. It allows the separation of such concerns from the main application logic, promoting better modularity and reusability. AOP achieves this by creating entities called "aspects" that encapsulate these cross-cutting concerns and can be applied across different parts of the application.

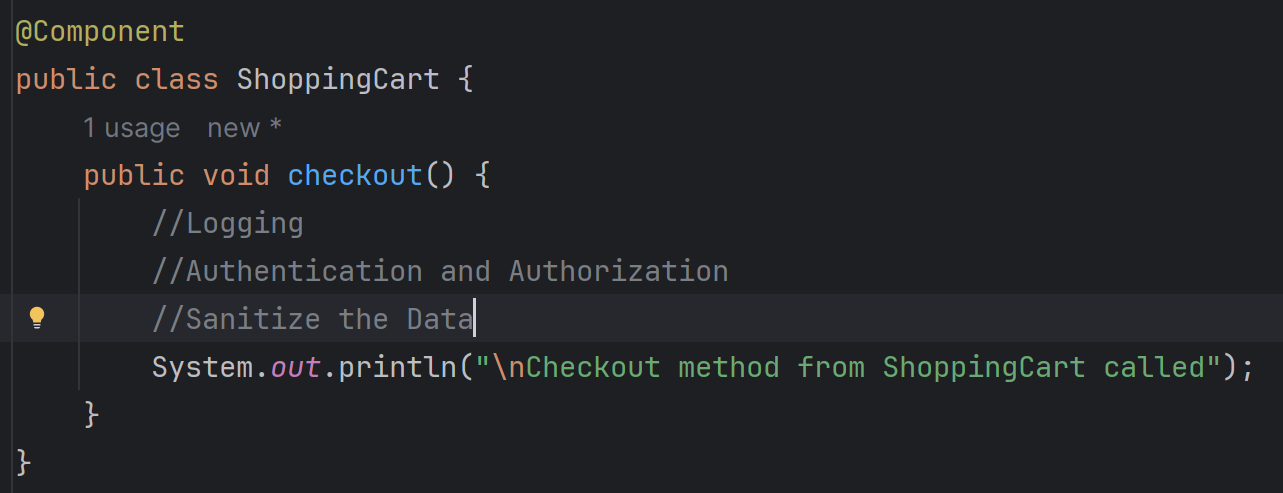
i.e.

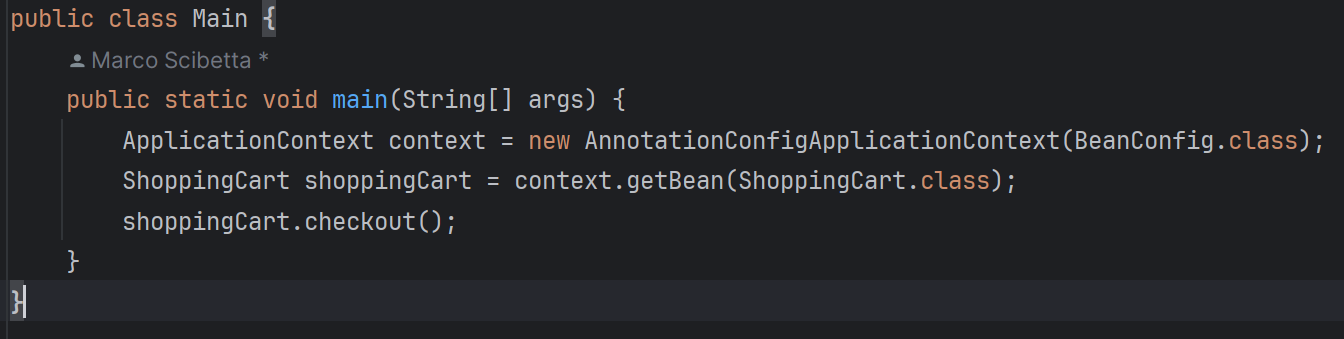
* Logging
* Authorization and authentication

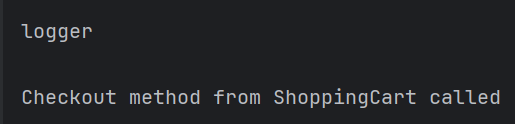
In aspect-oriented programming (AOP), a **joint point** is a specific point in the application's execution, such as method execution, exception handling, or changing object variable values. It is where you can apply additional functionality using AOP. On the other hand, a **pointcut** is a predicate that matches one or more joint points.











1.22.00