

In general things that CHANGE should be **hidden**, such as:

* Implementation details

Thing that should not change are **revealed** through INTERFACES, such as:

* Assumptions

You should just have to give what the other module needs, not all of the details

You are a software engineer at Amazon.com who works on the system used to ship products out to customers. It is important to the company that they restrict sensitive customer information on a need-to-know basis.

**Identify which parts of the customer’s information you would need to do your work on the shipping system, and which parts should be hidden because you do not need it.**



Full name

Correct

Great job! This is relevant information to the shipping company since it is needed to find a person, especially if there are problems with delivery.



Phone number

Correct

Great! Usually, a phone number is included with shipping, so that the shipping company can contact the customer if there are problems!



Payment method



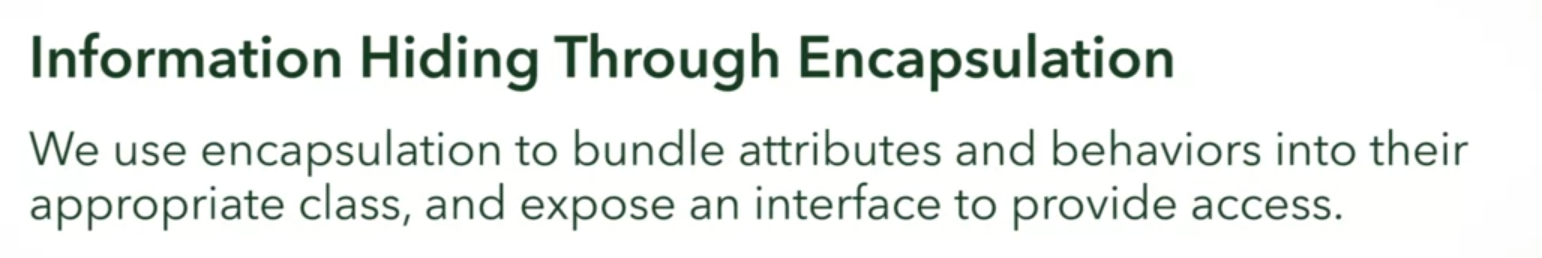
Email



Address

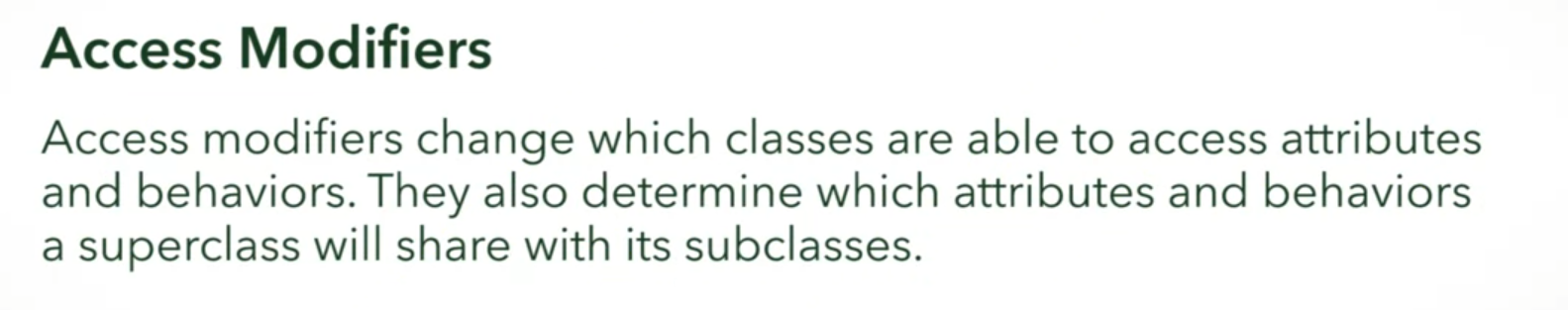
Correct

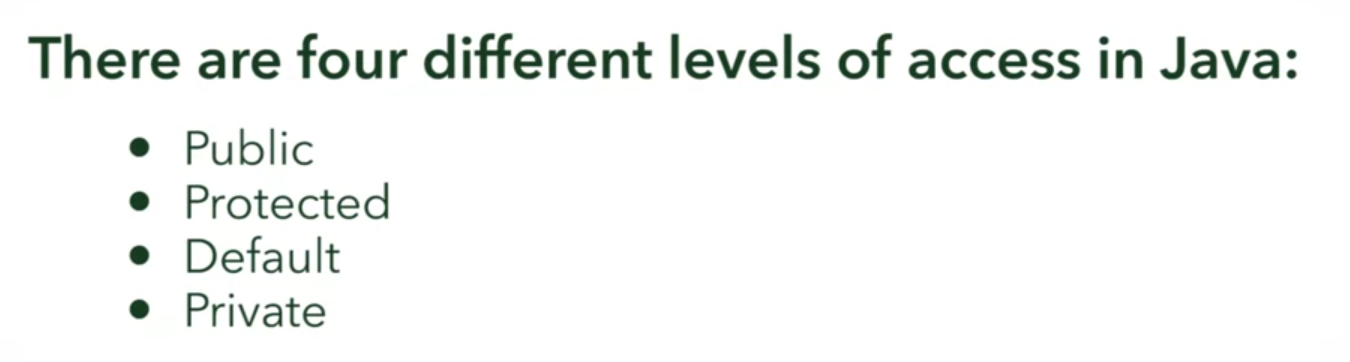
Absolutely! The shipping system is going to need the address of where the package will be sent.



Example:

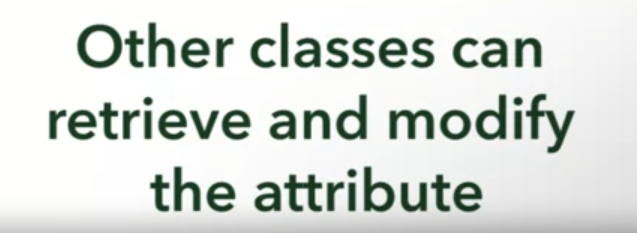




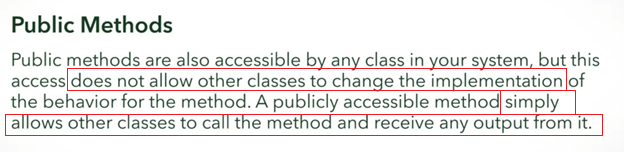


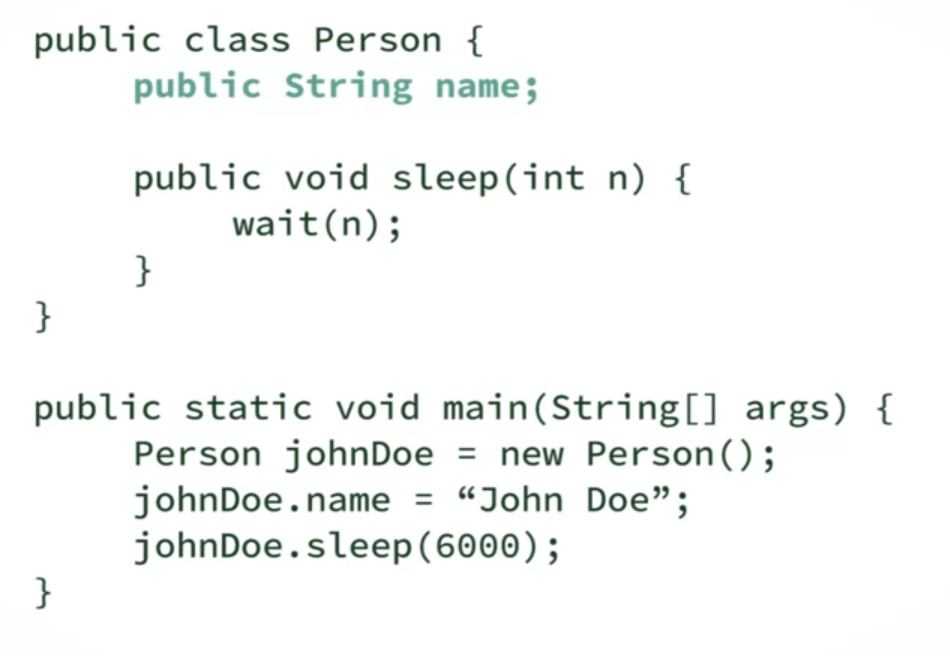
PUBLIC ACCESS MODIFIER

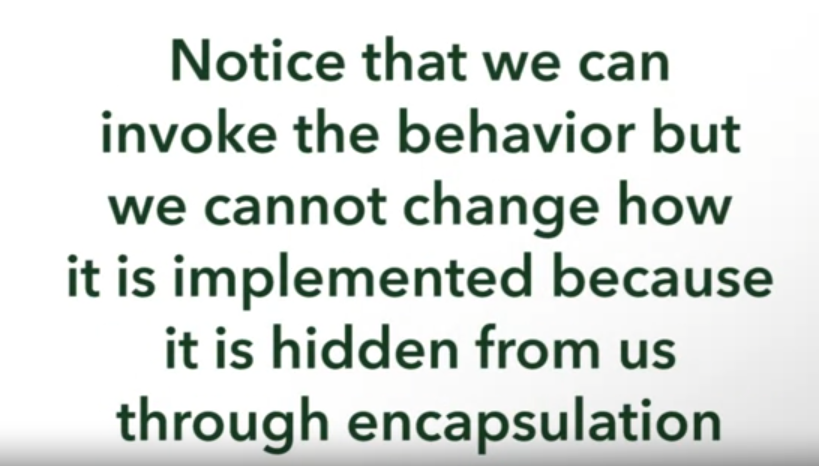
Public class



Public methods

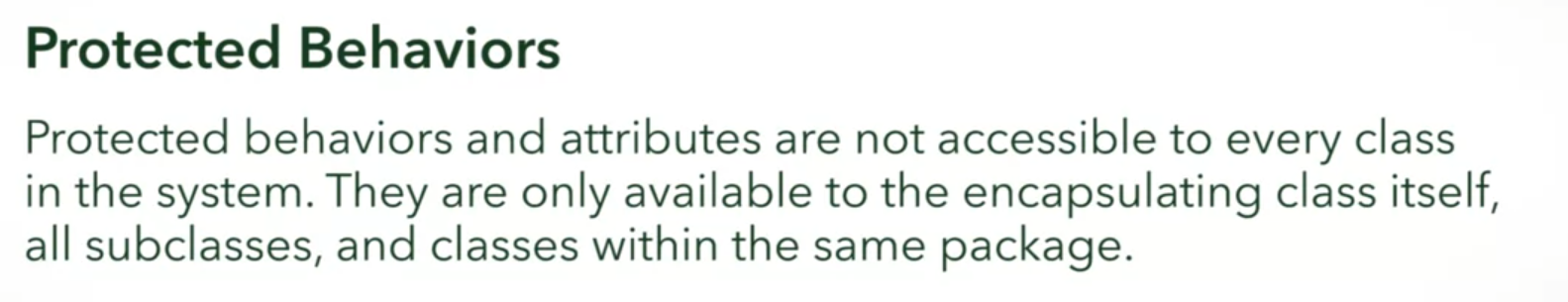






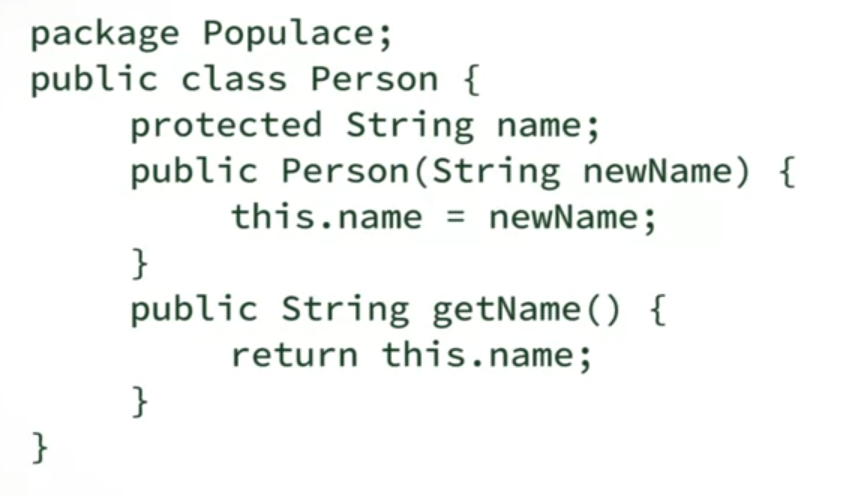
PROTECTED ACCESS MODIFIER

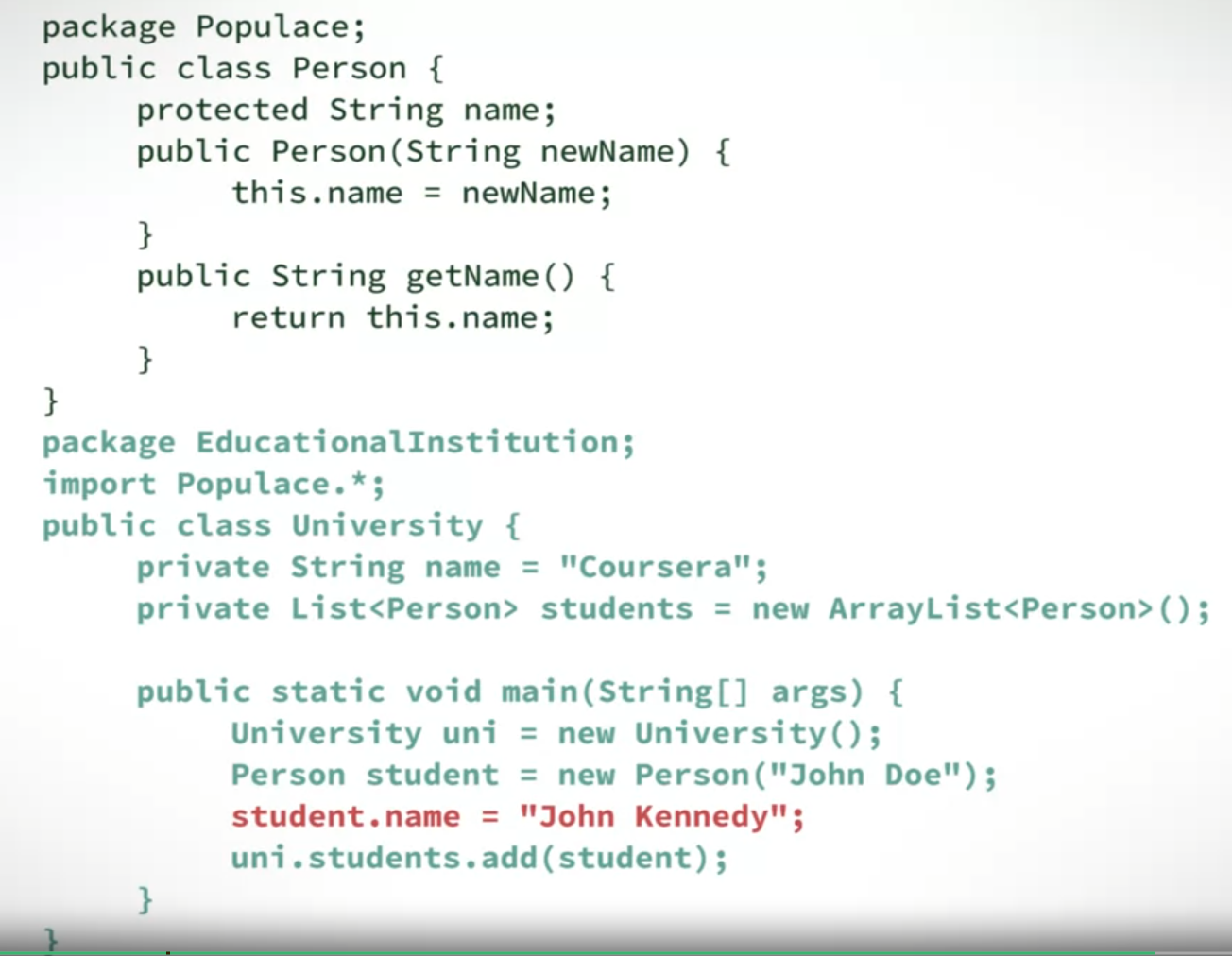
Protected behaviors (same with class and method)





Example of protected:

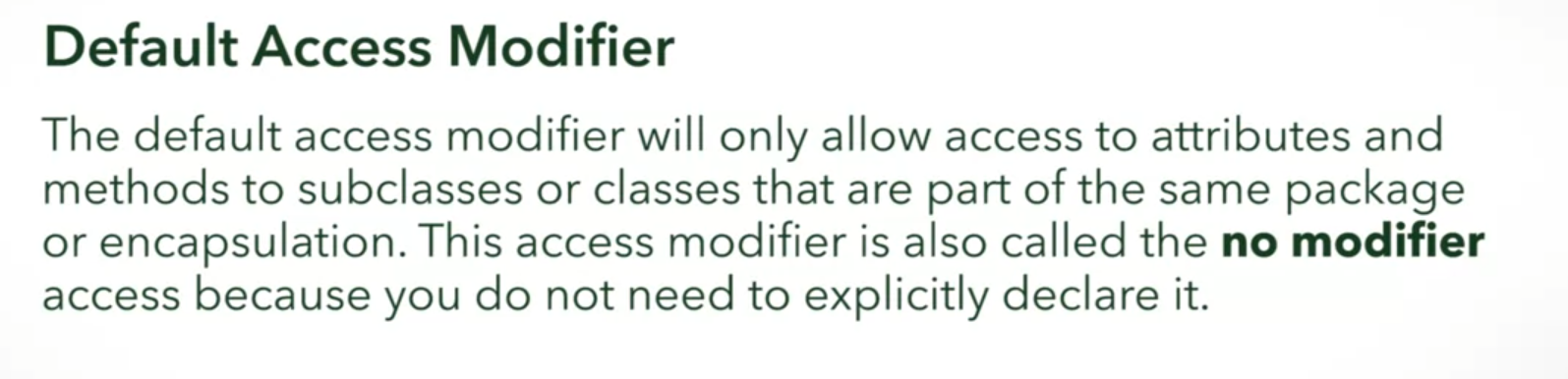


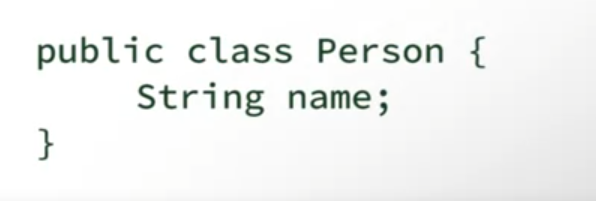


* This does not work



DEFAULT ACCESS MODIFIER

****



PRIVATE ACCESS MODIFIER

