

Day 07 – Java Inheritance

Classes and Objects in Java

What is a Car with special attributes and functionality? Think of things which may not be included in your Car class.

Hybrid or electric cars, have batteries and might not have gasoline.

Let's focus on a hybrid.

What are some attributes and functions they have in addition to the standard Car class?

Probably related to the battery and charge. What is the current charge level? That would be an **accessor** or “**getter**.” The ability to plug in and increase the charge seems like another.

So, our new class, Hybrid.java, will have additional attributes and functionality. But we actually call it the **sub-class**, because it derives from the **super-class**, Car.java.

Note: this always confuses me, until I look at it graphically, in a UML diagram, with classes and arrows. The sub and super don't imply functionality, but inheritance.

Also, in the sub-class, we can take advantage of what the super-class has already set up. Note the use of super() in constructors and toString()

Implement your own Hybrid.java and TDHybrid.java. Some code is provided as a help for you, but in a low-stakes environment, you should really try to code these on your own. That's how you'll learn.

Look at the True/False in Beginning Object Inheritance

Iterated development!

Cite your collaborators and sources!