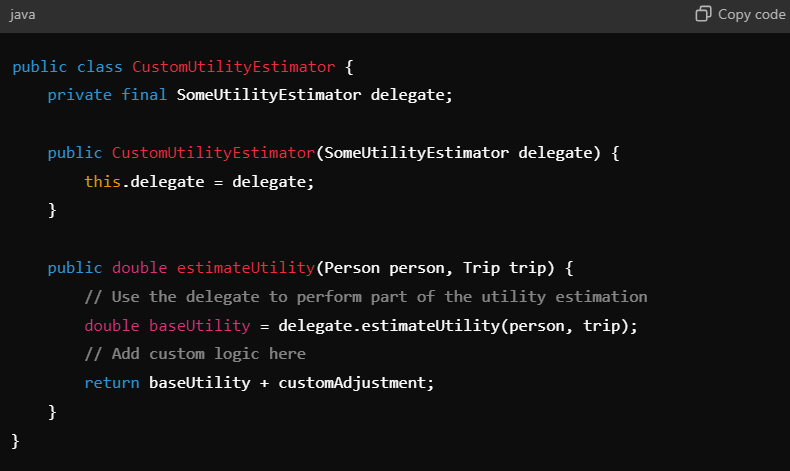
**How "delegate" is Used in MATSim**

In MATSim, "delegate" is often seen in the context of design patterns such as **delegation** or **composition**. The idea is that instead of inheriting behavior from a base class, an object delegates some of its responsibilities to another object. This is a flexible alternative to inheritance and is used to share or extend functionality.

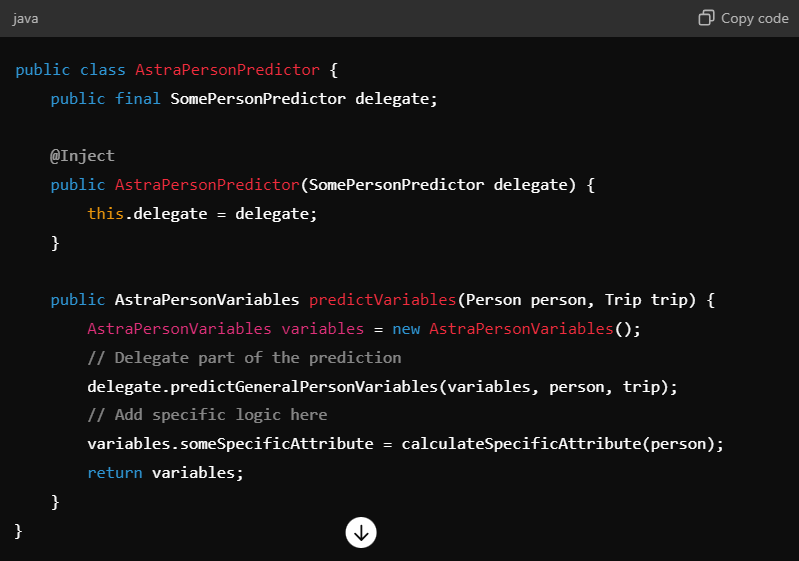
Here are a few ways "delegate" might be used in MATSim:

1. **Utility Estimation**: In utility estimators, a delegate might refer to another utility estimator or predictor that is used to calculate part of the utility or to carry out a task that the current object needs to perform.



 In this example, delegate refers to another SomeUtilityEstimator that the CustomUtilityEstimator relies on to perform part of the utility calculation. This is useful for creating layered or extended functionality without modifying the original class.

 **Predictors**: Predictors in mode choice models might use delegation to compute certain variables. For instance, an AstraPersonPredictor might have a delegate that is responsible for predicting more general person-related variables, while AstraPersonPredictor adds its own specific logic.



1. Here, delegate is another SomePersonPredictor that does some basic person-related predictions, while AstraPersonPredictor adds more specific logic.
2. **Routing and Plans**: In some cases, delegate might refer to a routing or plan handling component that does the main work of generating routes or handling plans, and the main object may add additional features or constraints.

**Why Use Delegation?**

* **Flexibility**: Delegation allows you to easily extend or modify functionality without changing the original class. This makes it easier to maintain and update code.
* **Reusability**: You can reuse components by delegating tasks to them, rather than duplicating logic in multiple places.
* **Decoupling**: Delegation reduces the dependencies between classes. The delegating object doesn’t need to know how a task is performed, only that the delegate can perform it.

In summary, in MATSim and similar systems, "delegate" usually refers to a pattern where an object defers a part of its behavior to another object. This approach is used to build flexible, reusable, and decoupled systems.