Contents

[***How many domain classes should we have in your application?*** 1](#_Toc450149643)

[***Should domain classes need to be related to each other?*** 1](#_Toc450149644)

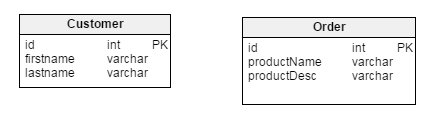
[***Is there a quick way to reverse engineer and visualize my classes in UML?*** 2](#_Toc450149645)

### **How many domain classes should we have in your application?**

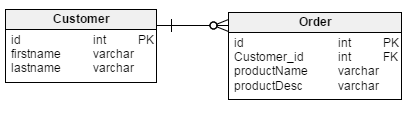
* + 1. No scientific formula – in the real world, depending upon an application, there could be a large number (tens to hundreds) of domain classes.
       1. For example, software for a convenience store would contain far fewer domain classes then say an FAA system.
    2. **For our academic purpose – you need to have at least 3 domain classes**

### **Should domain classes need to be related to each other?**

1. To understand the need for this, let’s digress and consider a database example.
   1. Consider the following tables. If the tables are not related (via foreign keys) then they provide very little value. We’d know of customers, and of Orders but we cannot determine what Orders are related to what Customer.

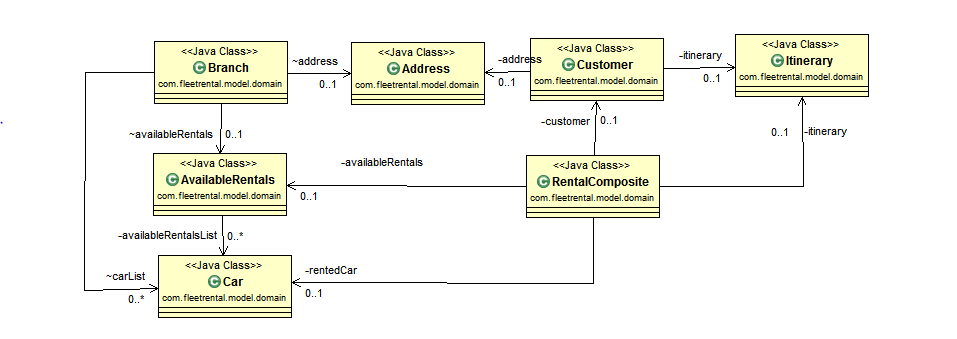


Tables below which are related to each other provide a more meaningful business value. We now know Customers and their associated Orders.



**Note: Tables designed using online tool** <http://www.vertabelo.com/>

1. The following domain model used in the FleetRental sample used in this course should validate the above understanding.
   1. For example, here we see that the Address class is related to both the Branch and Customer.
   2. Take away should be that no class is an island – they are all related/connected to one or more classes.



### ***Is there a quick way to reverse engineer and visualize my classes in UML?***

Good news is yes, there is a way.

Do follow instructions under Download link of the product to install the free plugin into Eclipse : <http://www.objectaid.com/home>

This tool allows one to drag and drop classes and it automatically builds and depicts the UML relationships.

**-END-**