
# Lab 01
# Software Design Patterns

Prepared for: Dr. Khalid AlHarbi

Prepared by: Faisal R. AlShehri
UID: 1935864

Faculty of Computing & Information Technology, Department of Information Technology

CPIT-252

A1: We fix the code by changing the quantity from being an instance variable to a static variable. Furthermore, it is illogical to access the quantity via an instance object of the class. Therefore, I also changed the getTotalQuantity() method to static, and adjusted the method call in the main accordingly.

A2: We were requested to skip this part as it requires a certain dependency that will be explained later on in the course.

A3: We can make the class abstract.

A4: All the instances applied polymorphism as demonstrated in the code, and each object was printed via a loop that iterates over the array.

A5: By making the method final.

A6: Because it will cause all other instances to crash when running, as well as require the changing of all subclasses. I'd fix it by iterating on all objects, saving their data in an array, nullify them, and instantiate them. I know this isn't ideal at all, but that's is what I thought of.

A7: Because it enables users to adjust or manipulate the weight by not applying encapsulation. It can be solved by making it private and adding a setter and getter to access it.