FACTORY PATTERN
SIMPLE FACTORY
SIMPLE FACTORY allows interfaces for creating objects without exposing the object creation logic to the client.
SimpleFactory

+ CreateProduct(enum type)

switch(type):
Case A: return new ProductA();
Case B: return new ProductB();

Client

ProductA

ProductB
FACTORY METHOD
FACTORY METHOD defines an interface for creating an object, but let subclasses decide which class to instantiate.

It lets a class defer instantiation to subclasses.
```java
SomeBehaviour() {
    product = MakeProductB();
    ... }
```
ABSTRACT FACTORY
**Abstract Factory** provides an interface for creating families of related or dependent objects without specifying their concrete classes.
ABSTRACT FACTORY does not specify how the objects should be created. It only enforces that objects are created in families.

To create the objects it is usually paired with either the factory method or the prototype pattern.
HOMEWORK: 26.04 - 10.05

Read Factory chapter from CGLearn
https://cglearn.eu/student/materials

Do the Factory task from CGLearn