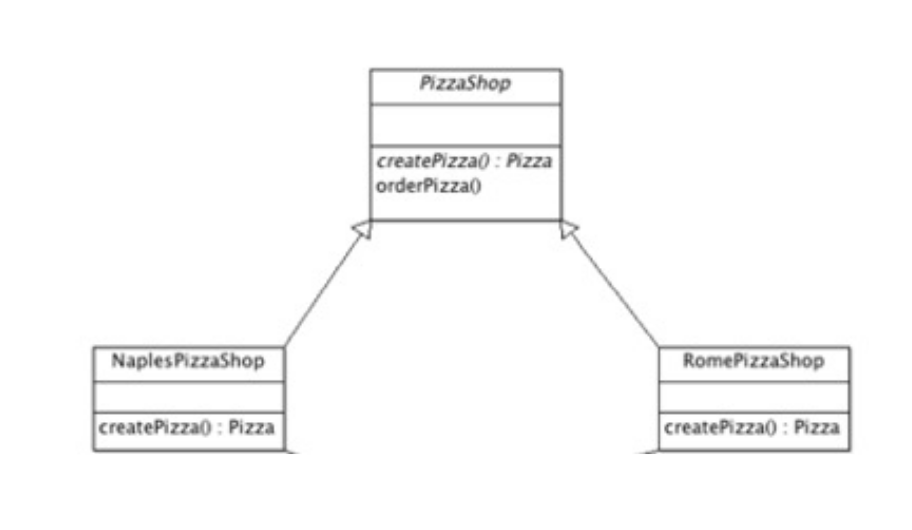
Design pattern provides a framework to solve problems from

* Provide a reusable solution or path to solving a specific type of problem
* Based on some principles of OO design
  + Separate what varies and what stays the same
  + Program to an interface, not an implementation
  + Favour object composition over class inheritance

3 types

* Need to know patterns introduced in lectures
* Creational patterns
  + Object creation
  + Factory pattern
  + Defines an interface for creating an object but lets subclasses decide which class to instantiate and allows subclasses to do said instantation
    - Example making pizza



Diagram

Description automatically generated

* + Abstract factory pattern
    - Provides an interface for creating groups or families of related objects without specifying their concrete classes
    - Supports creation of different groups
      * E.g. different OS or different pizza factories

Diagram

Description automatically generated

* Both abstract pattern and factory pattern are used for object creation
* Factory pattern uses inheritance to allows a class to be extended and methods overridden and provides an object in one ‘shot’
* Abstract factory pattern provides an abstract class and uses composition to create a concrete group of objects and classes, that determine how an object is produced. It returns a group of related classes
* Structural patterns
  + Composition of classes and objects, in order to form larger structure
* Behavioural patterns
  + Interaction between objects