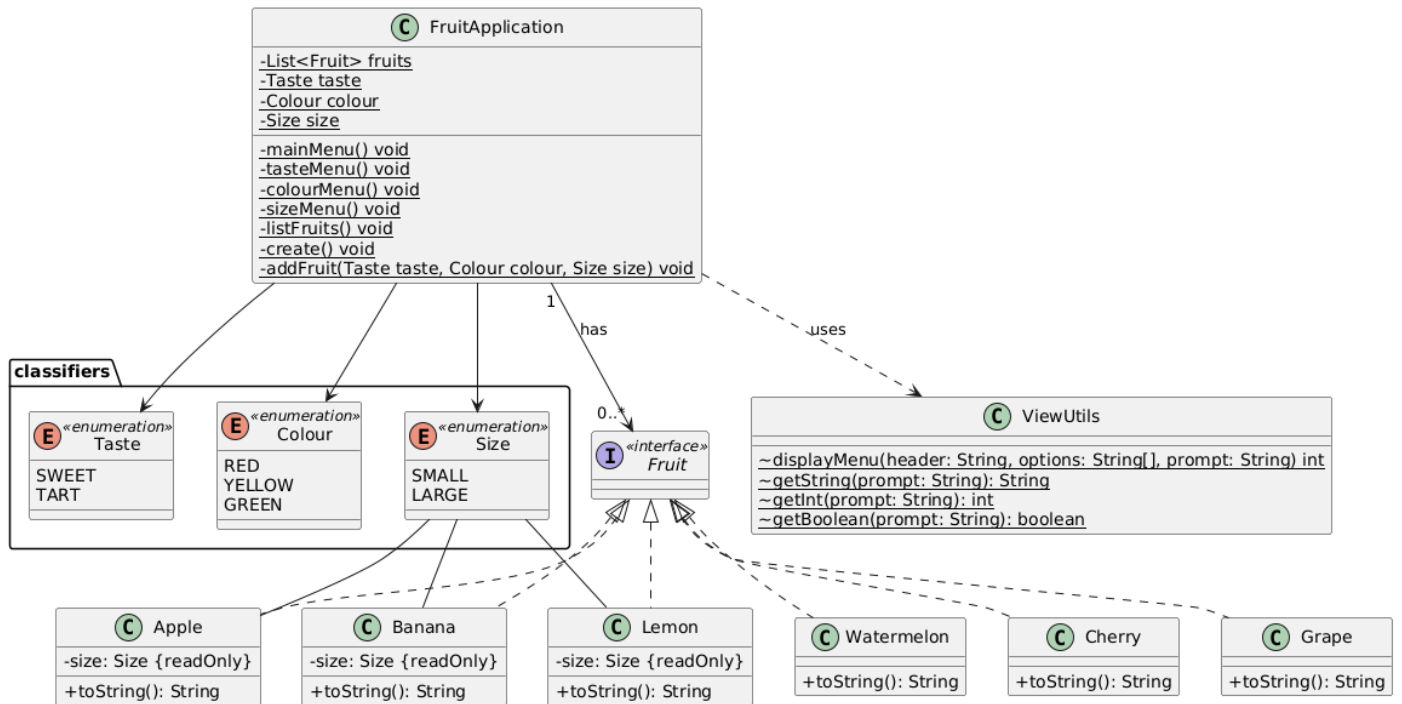
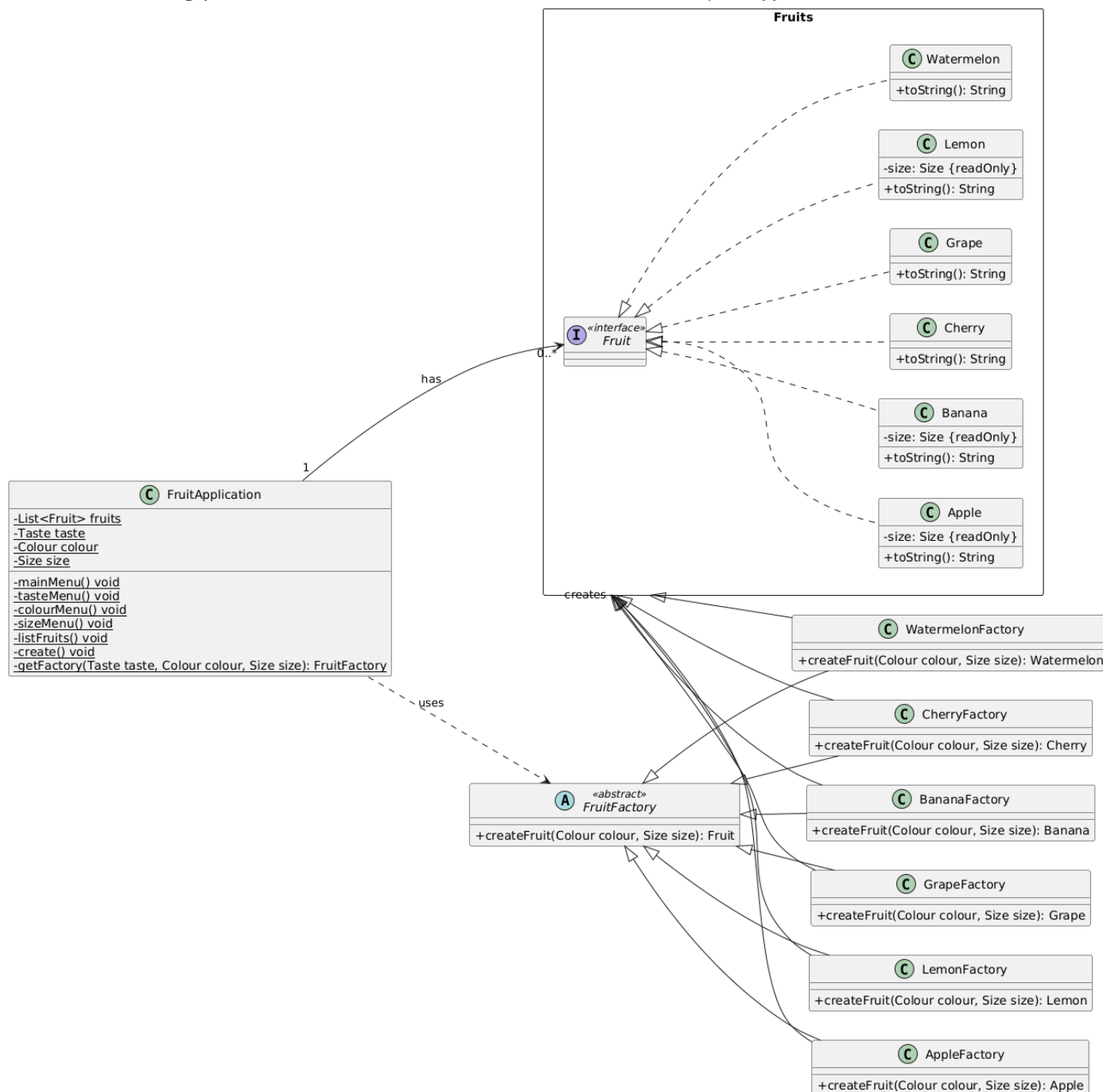


Exercise Week 4

Before Re-factoring



After Re-factoring (note: omitted Classifiers and ViewUtils for simplicity)



1) Identify the corresponding participants of the Factory Method Pattern:

The participants of the Factory method pattern include:

- **Fruit**: the “Product” which defines the interface for objects that the factory method will create
- **Different Fruits**: Apple, Banana, etc which are the specific classes that actually implement the Fruit interface. Each of these Fruits are created by each concrete factory.
- **FruitFactory**: the Creator, an abstract class declaring the factory method which will return a Fruit object, to be implemented to its subclasses.
- **Factories**: the AppleFactory, BananaFactory etc which are concrete factories implementing the Creator and implementing the abstract createFruit method.

2) Identify 1 major positive effect and negative effect

Positive: Our FruitApplication becomes more extensible, it is open for extension but closed for modification, by creating new concrete creator classes, such as a MangoFactory, and new create product classes, such as a Mango. This aligns with the Open/Closed principle. Moreover, we follow

GRASP very well – the use of specific fruit factories fulfill Information Expert and Creator. The FruitApplication remains a Controller, and we help to maintain low coupling between FruitApplication and the concrete Fruit classes.

Negative: There is now additional complexity, clearly seen through the UML diagram. There are many more classes to manage and with any additional concrete product we must introduce new factories. This may have a negative impact on readability and maintainability, with increased overhead in regards to maintenance, and potential code duplication across very similar fruits.