Design Patterns and Class Diagram Assignment AU

Solution 1:

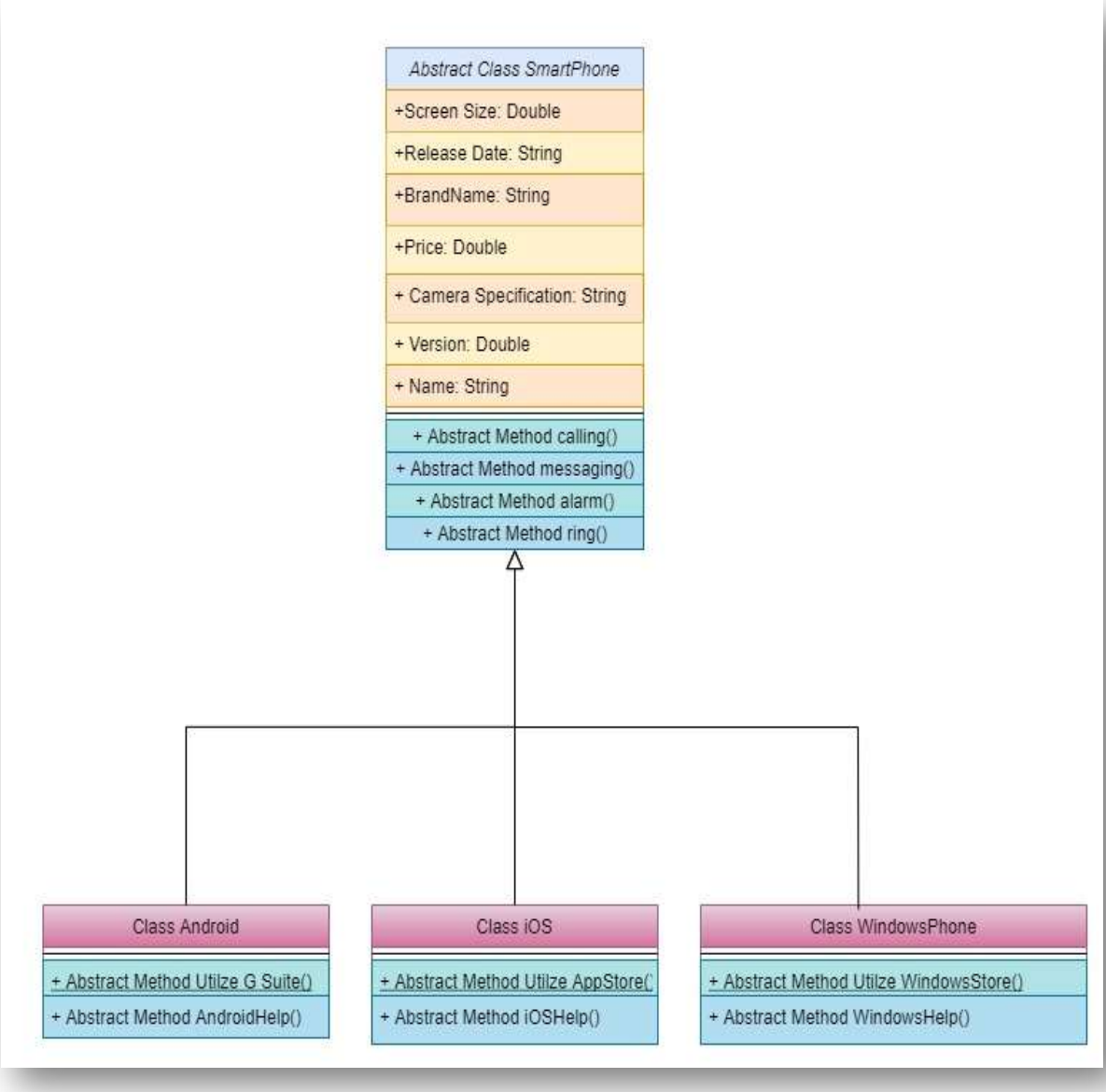

Figure 1 Showing Class Diagram of Problem 1

Solution 2:

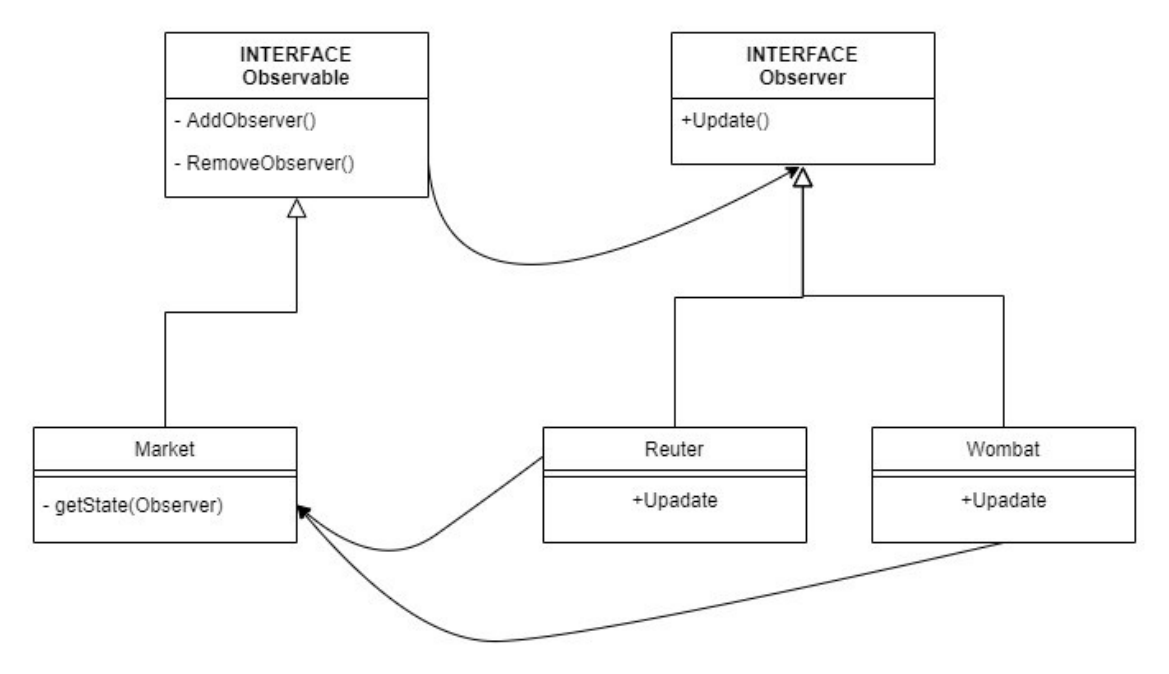

Figure 2 Showing Class Diagram of Question 2

Solution 3:

Singleton Pattern says that just define a class that has only one instance and provides a global point of access to it. In other words, a class must ensure that only single instance should be created and single object can be used by all other classes.

There are two forms of singleton design pattern:

1. **Early Instantiation:** creation of instance at load time.
2. **Lazy Instantiation:** creation of instance when required.

Advantage of Singleton design pattern

- Saves memory because object is not created at each request. Only single instance is reused again and again.

Usage of Singleton design pattern

- Singleton pattern is mostly used in multi-threaded and database applications. It is used in logging, caching, thread pools, configuration settings etc.

UML of Singleton Design Pattern

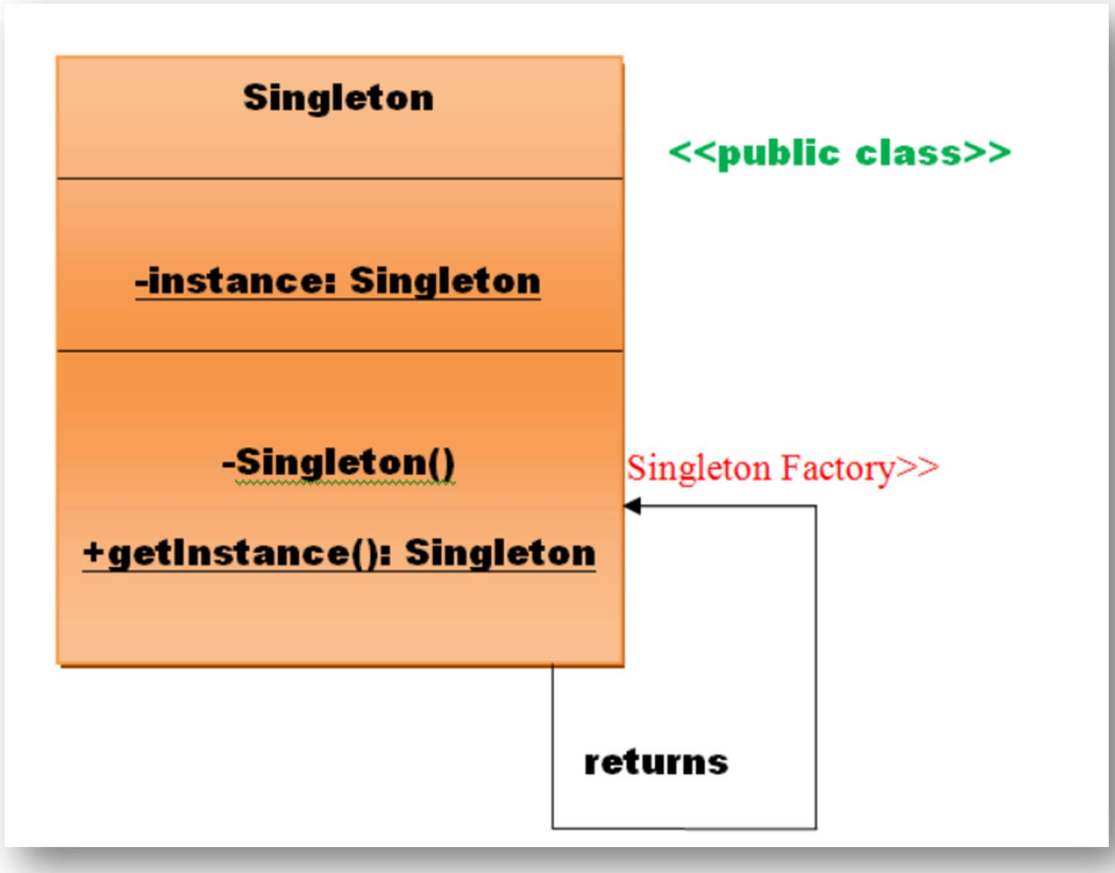

Figure 3 Showing UML of Design Pattern

Thread Safe Singleton to Handle Multiple Singleton Cases

```
public class ASingleton {
private static volatile ASingleton instance;
private static Object mutex = new Object();
private ASingleton() { }
public static ASingleton getInstance() {
ASingleton result = instance;
if (result == null)
{ synchronized (mutex) {
result = instance;
if (result == null)
instance = result = new ASingleton();
}
}
return result;
}
}
```

Solution 4:

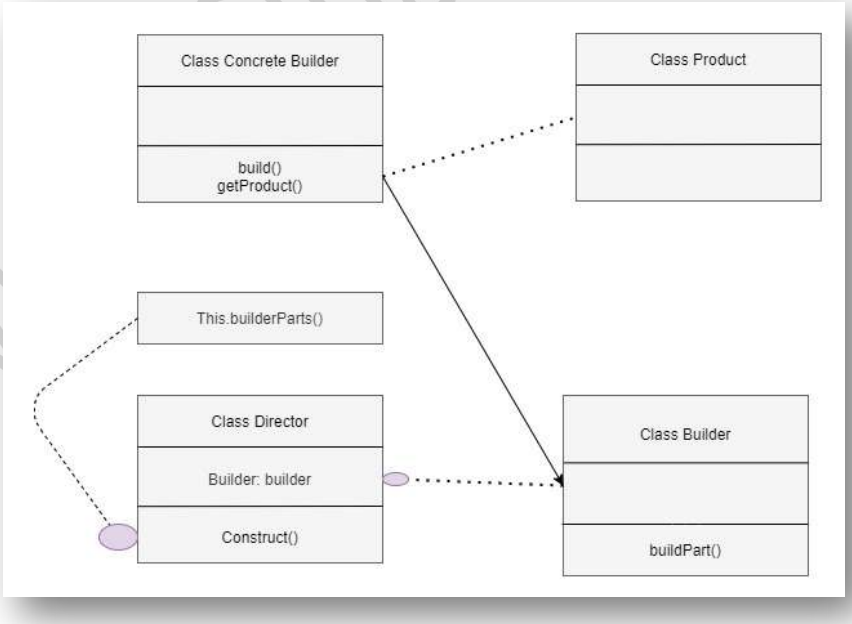

Figure 4 Showing Classes for Builder Pattern