Review: Chapter 3

**Written question:**

**3.1:** information hiding is used in object oriented used to restrict accesses to some objects. Encapsulation is bundled data and methods into one unit to be used together. Java supports these concepts by using getters and setters you can access the restricted objects, and you use classes to make those units.

**3.2:** public section of the class mean it can be accessed by anyone, while private of the class can only by accessed by members of the same class can only accesses.

**3.3:** the purpose of the constructor is the initializes for newly created objects to be used.

**3.4:** if you don’t provide a constructor then there will be a no argument constructor will be used.

**3.6:** if the void is included in the constructer it will act as a method and not the constructer.

**3.10:** difference between static and instance field, static have to use static keyword instance don’t have keywords, can invoke static field by the class name instance need an object to invoke it, if you use a static it can be used in many class or methods and only has to be created the one time, while can only use instance has to have copies if used elsewhere.

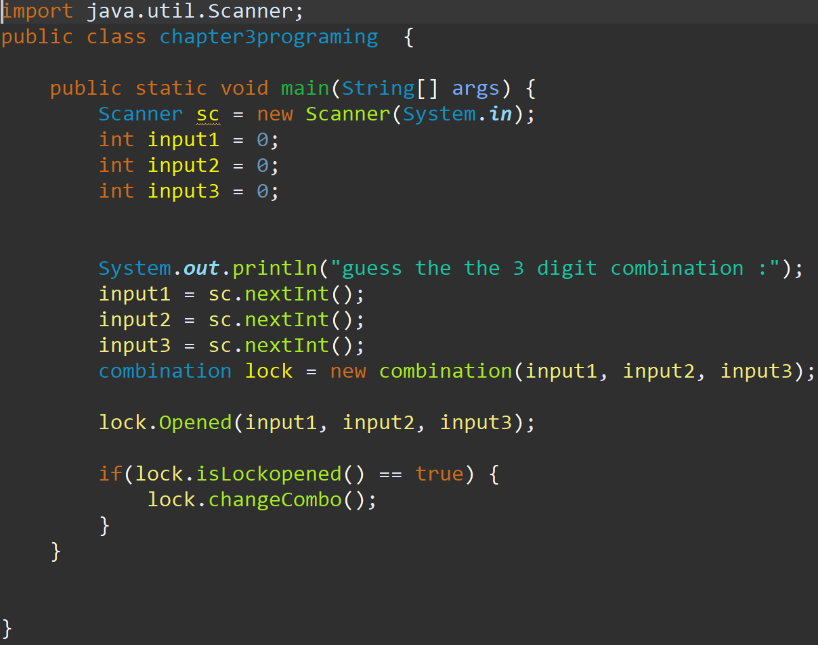
**In Theory:**

**3.14:** if you create a private constructor, then only the one object can be created and other classes do not have permission to make an object from that class because it is private.

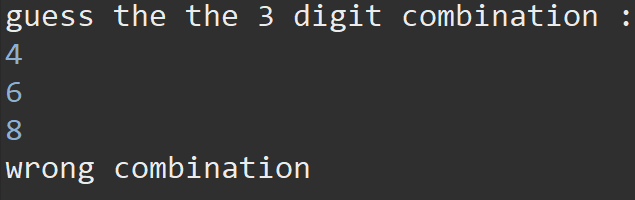
**3.15:** a. yes it would work. b. yes It would still work if it was uncommented.

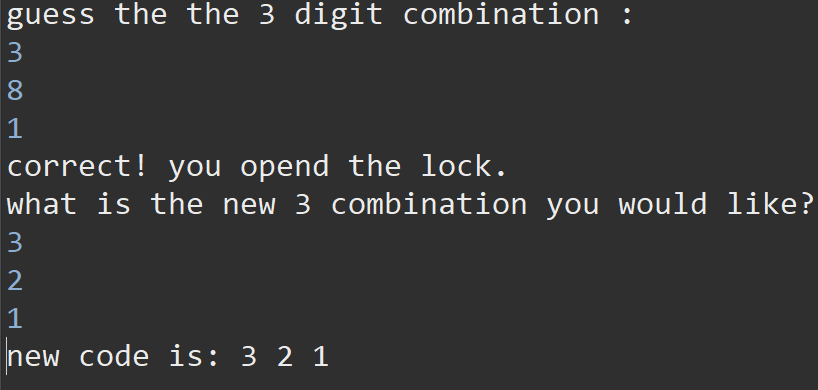
**Programming:**

**3.18:** here is my combination class that has the combination private so it can be accessed then I have a boolean isLockedopened method to check if the lock is open so people can change the combination. Then I have my method to check if the inputs are correct to have the lock opened and if is opened then I have a changeCombo method that if they got the right combo then you can put in a new combo



then I have the main which executes everything out.

what you get if you enter the wrong combination.

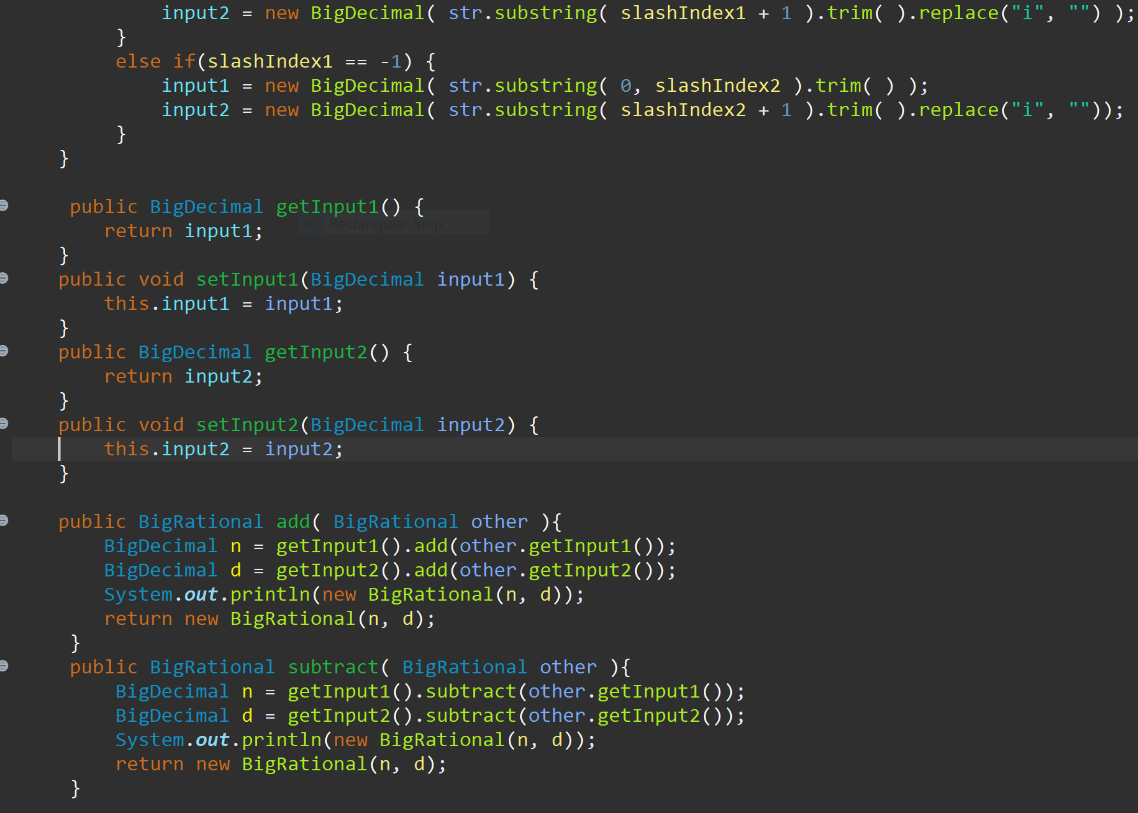


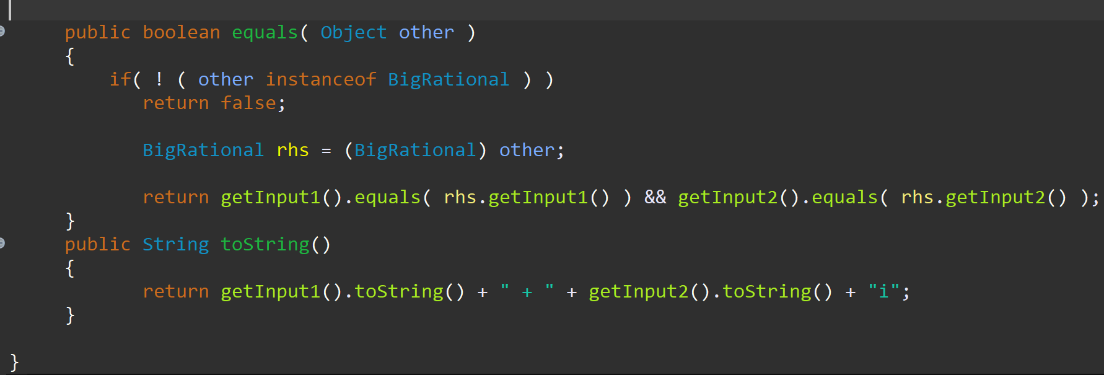
what you get if you enter the right

combination.

**3.31:**



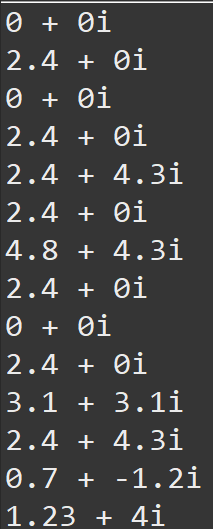






Everything I did in bigraiton class

And everything I had in my main to test out

the answer I got from testing my main all make sense and the add and subtract method are both working as well as the toStirng method and the equals method.