Day 3

Interface
- Cannot be instantiated
- Classes and interfaces can "inherit" from more than one
- All methods are implicitly public and abstract
- All fields are implicitly public static final
- In Java 8, default and static methods can have method bodies

Abstract class
- Cannot be instantiated
- Can have fields
- Can have both concrete and abstract methods
- A class can only extend one abstract class

FileInputStream vs ObjectInputStream vs FileReader vs BufferedReader vs Scanner
1. FileInputStream
    a. Reads 1 byte at a time which is not user friendly if you are trying to display the file to a user
    b. Which is why we will typically pass this into something like a ObjectInputStream's constructor.
2. ObjectInputStream
    a. Used to deserialize objects from a file.
    b. This uses an InputStream to read the bytes and then does the rest of the work creating an object for us.
3. FileReader
    a. Class in Java for reading characters of a file
    b. Reads 2 bytes at a time since Chars in Java take up 2 bytes.
    c. Not thread safe
4. BufferedReader
    a. BufferedReader wraps around the FileReader by passing a FileReader into its constructor.
    b. BufferedReader buffers by reading a whole line at a time and storing it rather than reading 1 character at a time. This gives the improved functionality to get entire lines rather than just single characters.
    c. Does still provide functionality to read single characters
    d. Thread safe.
5. Scanner
    a. Has smaller buffer memory than BufferedReader
    b. Has more functionality such as adding in delimiters
    c. Not thread safe

Conclusion:
   If all you want to do is read Bytes then use FileInputStream, if you actually want to convert these bytes to an object you will want to send the FileInputStream into an ObjectInputStream.

   If all you want to do is read Characters FileReader might be fine, if you want improved efficiency and the ability to read entire lines at a time use BufferedReader.

   If you want additional built in functionality, when reading from a file, such as being able to use delimiters then use a scanner.

Side-note:
   I went over the inputs but the corresponding FileOutputStream, ObjectOutputStream, FileWritter, and BufferedWriter would follow similar principles.