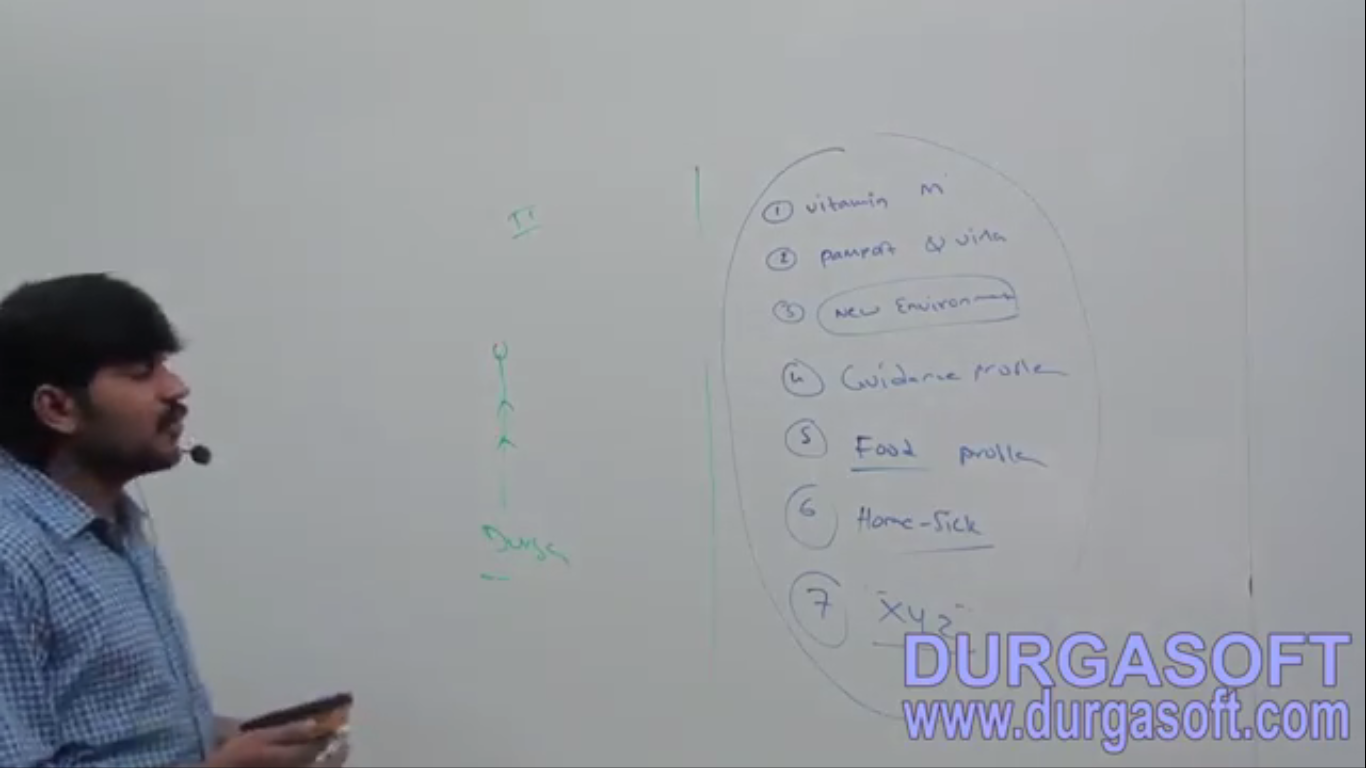
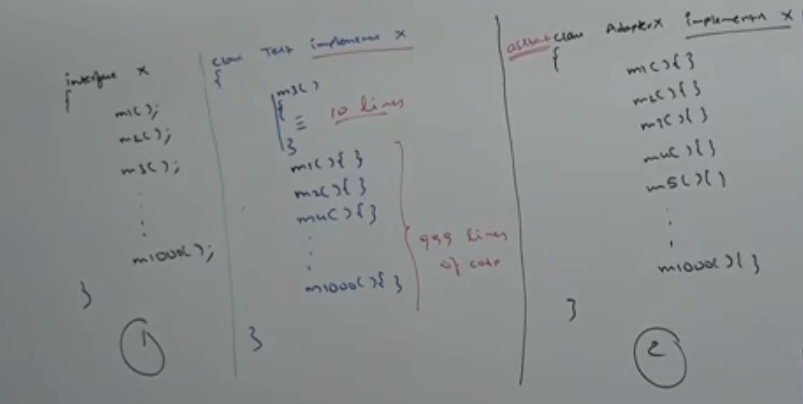
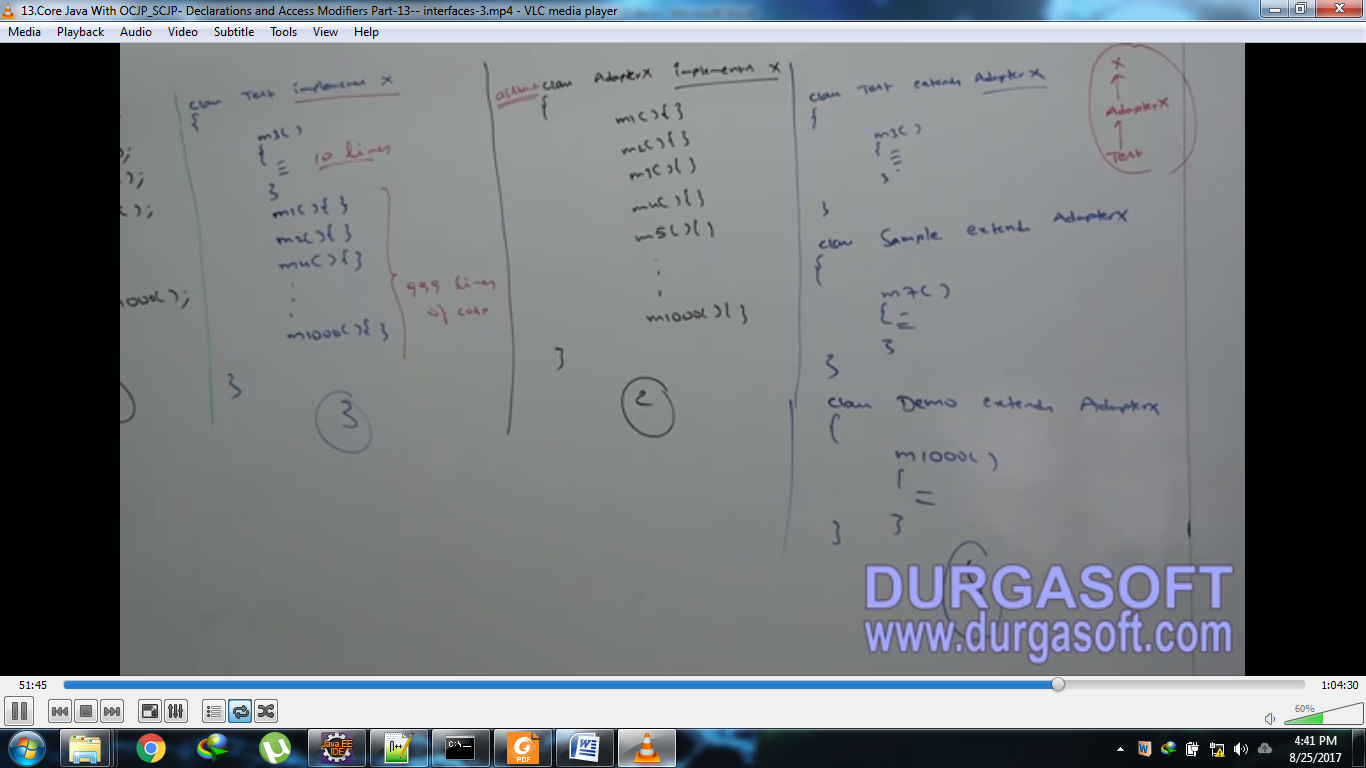
Marker interface

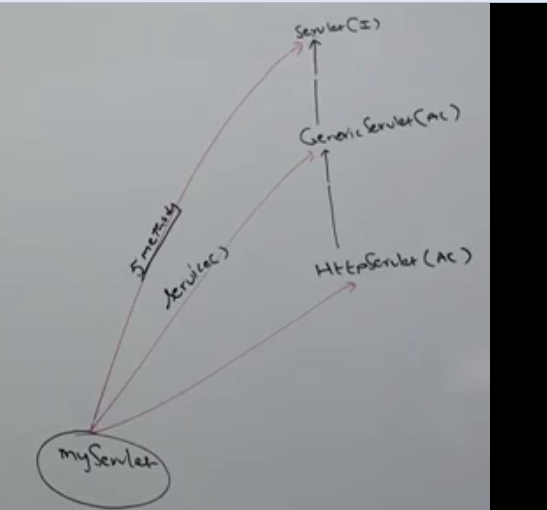
1. **Definition**: If an interface contains no method and by implementing that interface, the class object gets some ability such interface is called **“Marker Interface**”.
   1. **Serializable(I)**
   2. **Cloneable(I)**
   3. **RandomAccess(I)**
   4. **SingleThreadModel(I)  
      Etc.**

Also called 🡪 Ability Interface or Tag Interface

1. **How those implemented classes objects get the ability automatically via those marker interfaces**?
   1. Why Marker interface?  
      
   2. Internally JVM is responsible to provide the required ability.
   3. **Why Does JVM provide required ability marker interfaces?**
      1. To reduce the complexity of programming and to make java language as simple.
      2. Take example of Serializable interface.
   4. **Is it possible to write our own marker interface?**
      1. Yes but customization of JVM is required. JVM is not oracle proprietary. Any one is allowed to create his/her own JVM. Example: Weblogic has its own jvm called JRocket, Tomcat has its own jVM etc.
2. **I**

Adapter Classes

1. **Adapter Class**: A simple java class that implements an interface with only empty implementation.   
   
2. For each and every method for that interface, compulsory, we should provide implementation whether it’s required or not required.
3. **Problem if you yourself implements an interface having 1000 abstract methods out of which only 1 is required.** 
   1. Increases the length of the code.
   2. Reduces readability.
4. **Solution**: Using Adapter classes.   
   
5. **NOTE**: Declare Adapter class as abstract class as it contains only dummy method implementations.
6. **Where adapter concept we are already using?**
   1. **We can develop a servlet in the following 3 ways.**

**  
If we implement Servlet(I), for each and every method of that interface, we should provide implementation. It increases the length of the code and reduces the readability. Instead of implementing Servlet(I) directly, if we extend GenericServlet(AC), we have to provide implementation only for service(). And for remaining methods, we are not required to provide implementation. Hence, more or less, GenericServlet(AC) acts as adapter for servlet interface.**

1. **NOTE**: Marker Interfaces and adapter classes simplify complexity of programming and these are best utility to the programmers and programmers’ life becomes simple.