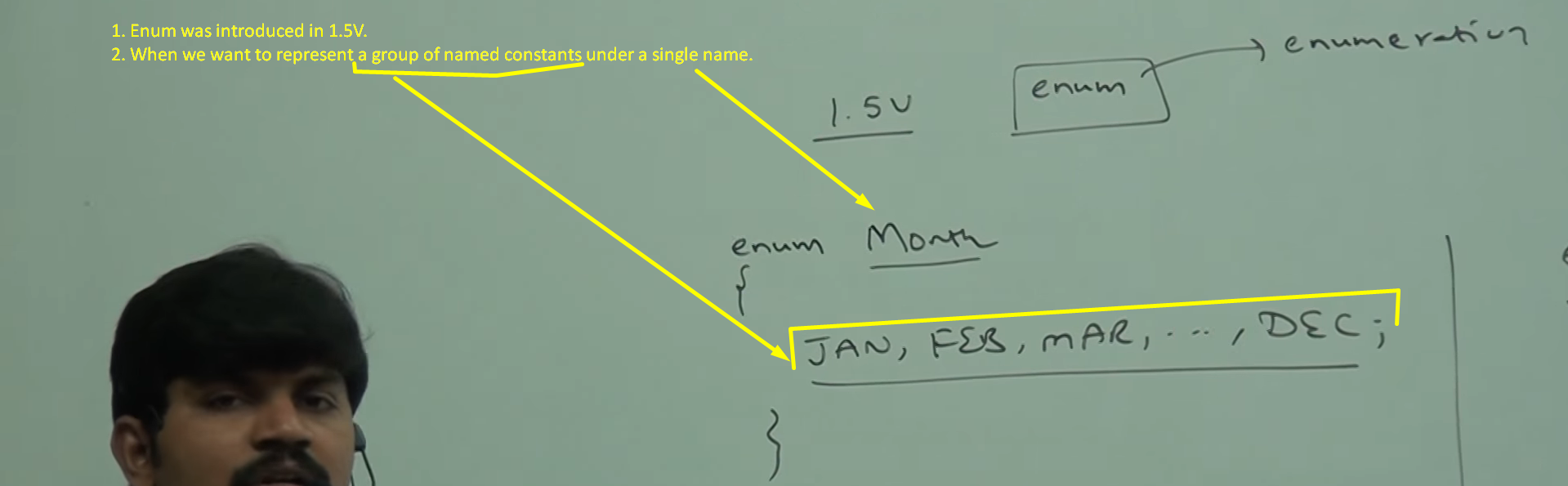
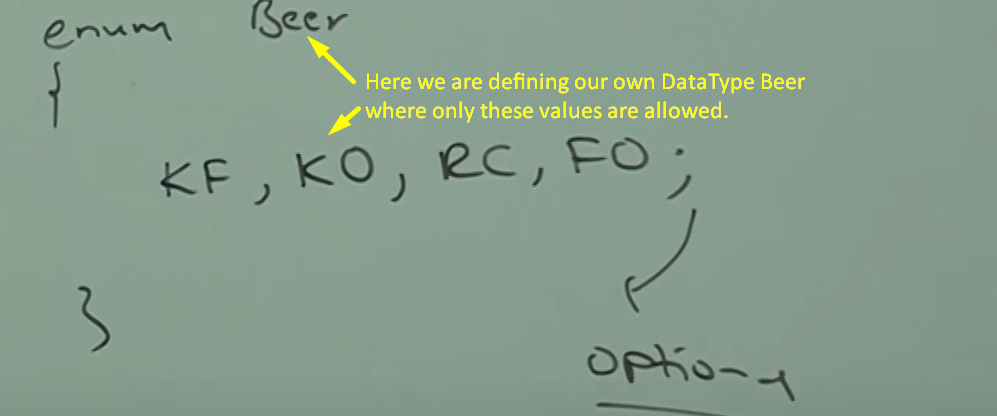
1. When we want to represent a group of **named constants** under a single name, then we should go for ENUM.
2. **Example**: We want to represent Month Names under a single name.
   1. 
3. **Main Objective/Use Case**:
   1. Like byte is in-built data type with range -128 to 127 = 256, similar, we can create our own data type using enum which is called **Enumerated Data Type**.  
        
      Semicolon is optional at the end.
4. **History Behind introducing ENUM by Java Team**:
   1. In Java 1.0V, ENUM Support was missing.
   2. World-wide Expert Programmers came with this missing part to Java Team.
      1. Even in any CS academic curriculum, in PL (Programming Language), PPL (Principal of Programming Language) exam, there is very important question like what is EDTs (Enumerated Data Types) and ADTs (Abstract Data Types).
   3. Java Team denied their requesting by a supporting statement like Java supports final static variables to define Named Constants.
   4. They went back.
   5. In Java 1.5V, Java Team introduced ENUM feature.
   6. Those people’s ego got hurt and they came back now why now you introduced this dummy feature.
   7. Java Team replied by saying that Java Enum is not only for defining named constants but you can define normal variables, constructors, methods like we do in normal class.
5. **Internal Implementation**:
   1. Using Class Concept.
   2. **For Example**:  
      enum Beer{  
       KF, RC;  
      }  
      **Equivalent Class**  
      class Beer{  
       public static final Beer KF = new Beer();  
       public static final Beer RC = new Beer();  
      }  
        
      Every Named Constant is converted into public static final and type is same as the enum name.
6. 