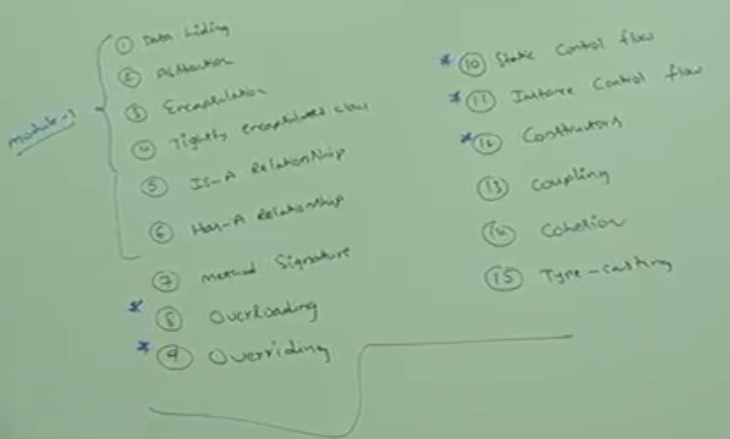
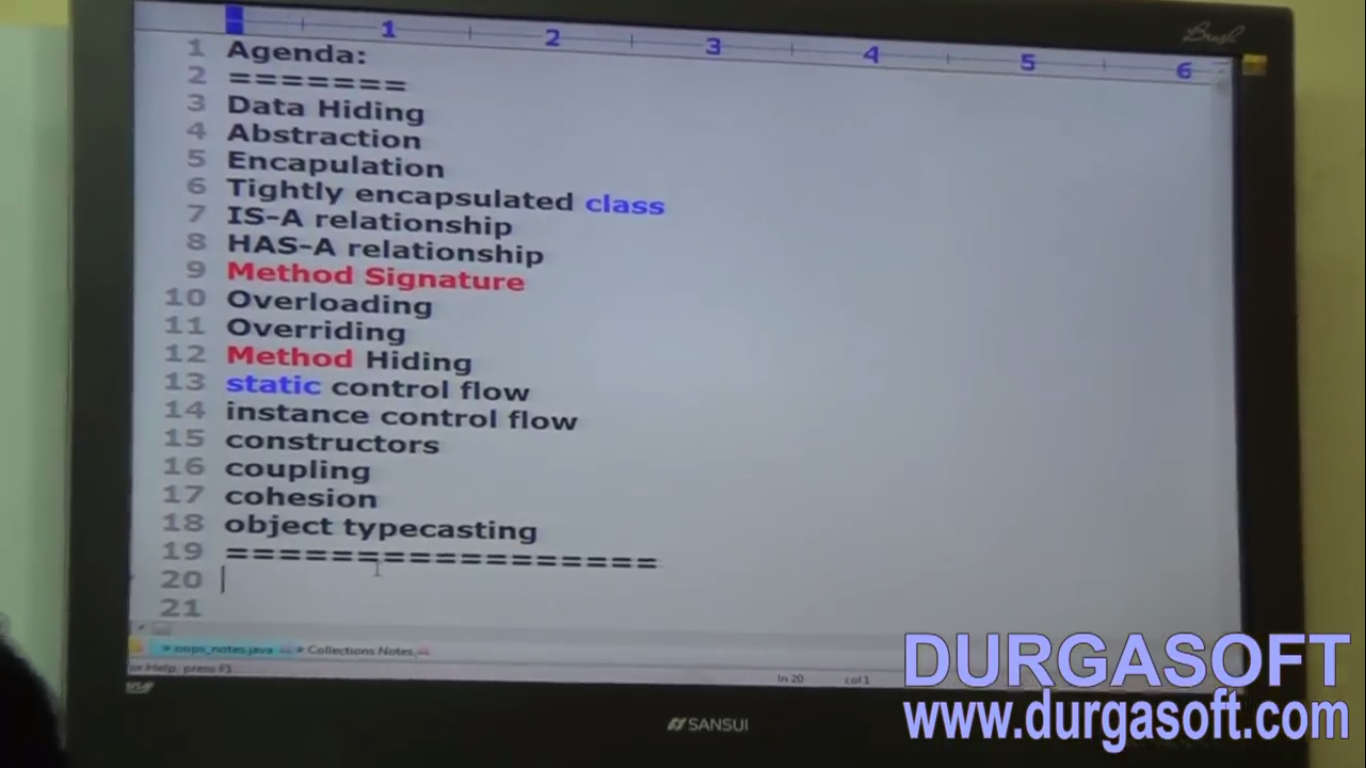
# We will discuss the following topics

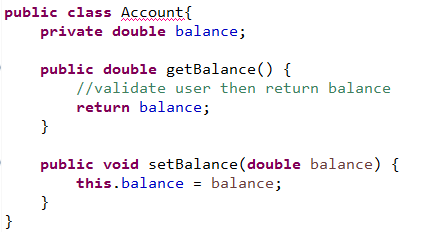
Data Hiding

1. Outside person can’t access our internal data directly or our internal data should not go out directly. This oop feature is nothing but data hiding.

After validation or authentication, outside person can access our internal data.  
**Example01**:  
 After providing proper username and password, we are able to access our gmail inbox info.

**Example02**:

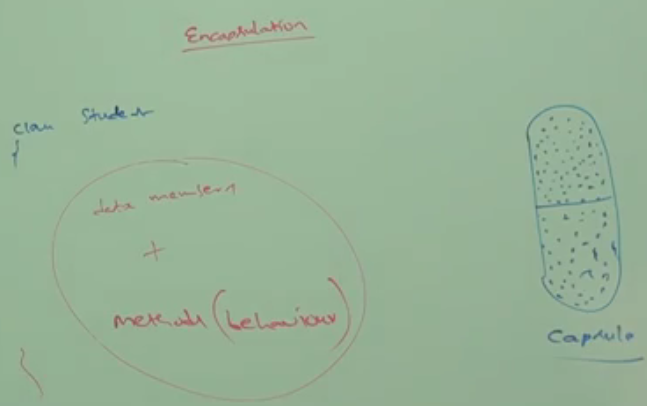
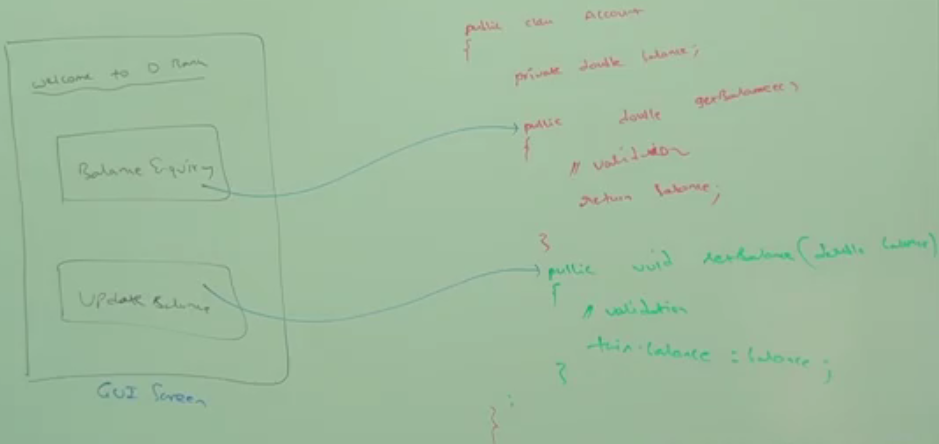
Even though we are valid custom of the bank, we are able to access our own account info but that of others.

1. **How to hide data**?
   1. By declaring data member (instance variable) as **private**, we can achieve data hiding.  
      
2. **Advantage of Data hiding**?
   1. Security
3. It’s highly **recommended** to declare instance variable (data member = variable) as private.

Abstraction

1. Outline. ATM.
2. **Definition**: Hiding internal implementation and just highlight a set of **services** what we are offering is the concept of **Abstraction.**
3. **Example**: Through Bank ATM GUI Screen, Bank people are highlighting the set of services what they are offering without highlighting internal implementation.
4. **Advantages**:
   1. **Security:** Because outside user doesn’t know internal implementation.
   2. **Modification or Enhancement**: In future, if we want to change the underlying language or change the logic of withdraw, without affecting the end user, we can do that.
   3. **Easiness Improved (Reduced Complexity)**: End user needs not to know underlying functionality.
5. z

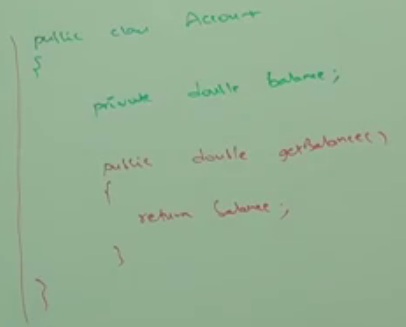
Encapsulation

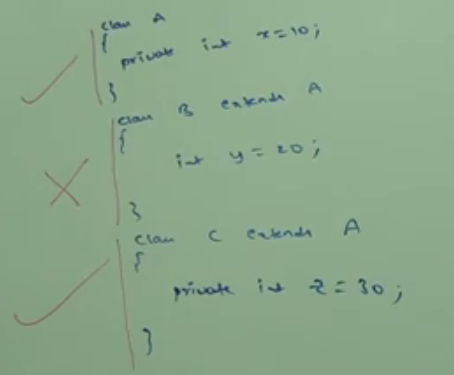
1. **Definition**: The process of **binding** **data and corresponding methods** into a **single unit** is nothing but **Encapsulation**.
2. **Example**: Class Student data members plus methods (behaviors).  
   
3. **Encapsulation** = Data Hiding + Abstraction.  
   If any component follows “data hiding” and “abstraction”, such component is **called encapsulated component**.  
     
   **Data Hiding**: Via private keyword with instance variable.

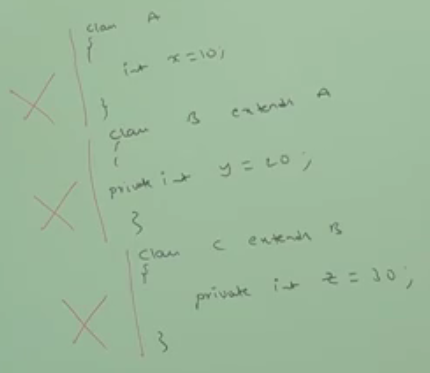
**Abstraction**: Via public methods

1. **Advantages**
   1. **Security**: We can achieve security.
   2. **Enhancement**: It becomes easy.
   3. **Maintainability**: Improves the maintainability of the app.
2. **Disadvantages**: Due to security process which takes time and many steps from end users, it slows down the execution.  
   Like, Enter password, OTP etc.

Tightly Encapsulation

1. **Definition**: A class is said to be “tightly encapsulated” if and only if each and every variable declared as private whether class contains corresponding getters/setters or not and whether these methods are declared as public or not. These things we are not required to check.  
   
2. Which of the following classes are “**tightly encapsulated”**?



1. **NOTE**: If parent class is not “tightly encapsulated” then the child is also not.  
     
   **All the following topics talk about** **Security**
   1. Data Hiding

**Security**

* 1. Abstraction
  2. Encapsulation
  3. Tightly Encapsulated Class