

1 Classes and Objects

IT-2027

```
class Laptop {  
    String brand; // private storing  
    int price;  
    void start() {  
        System.out.println("Laptop is starting...");  
    }  
}  
  
public class Main {  
    public static void main (String [] args) {  
        Laptop myLaptop = new Laptop();  
        myLaptop.brand = "HP";  
        myLaptop.price = 70000;  
        myLaptop.start();  
    }  
}
```

2 Access Modifiers

```
class Book {  
    private String title;  
    public void setTitle(String newTitle) {  
        title = newTitle;  
    }  
    public String getTitle() {  
        return title;  
    }  
}  
public class Main {  
    public static void main(String[] args) {  
        Book b = new Book();  
        b.setTitle("The Great Gatsby");  
        System.out.println(b.getTitle());  
    }  
}
```

③ Inheritance and Protected Access

```
class Vehicle {  
    protected String type = "Vehicle"; } follow radio  
    void display() { } storing  
        System.out.println("This is a vehicle.");  
    }  
  
class Car extends Vehicle {  
    void horn() { } follow radio  
        System.out.println(type + " goes Beep Beep!");  
    }  
  
public class Main { } follow car = follow  
    public static void main(String[] args) {  
        Car c = new Car(); follow  
        c.display(); follow  
        c.horn(); follow, two mistakes?  
    }  
}
```

4. Encapsulation

```
class wallet {  
    private double money; // private before  
    public void addMoney (double amount) {  
        if (amount > 0) {  
            money += amount;  
        }  
    }  
    public double getMoney () {  
        return money;  
    }  
}  
public class Main {  
    public static void main (String [] args)  
    {  
        wallet mywallet = new Wallet ();  
        myWallet.addMoney (300);  
        System.out.println (mywallet.getMoney());  
    }  
}
```