1. Explain the disadvantage of creating a class that has no set and get methods for an instance variable **(3 Marks)**

Answer:

If no setter or getter for an instance variable in a class, you can't get access or modify instance values when you call this class

2. Explain how a program could use class Scanner without importing it. (refer to textbook or lecture slides) **(3 Marks)**

Answer:

We can "new" the scanner function with its fully qualified name (java.util.Scanner), and then we can use it same as we import it

## 3. Modifying the Account Class (7 Marks)

Modify class Account (Figure 3.8 – code uploaded on canvas – source code chapter 03). To provide a method called withDraw that withdraws money from an Account. Ensure that the withdrawal amount does not exceed the Account's balance. If it does, the balance should be left unchanged, and the method should print a message indicating "Withdrawal amount exceeded account balance." Modify class AccountTest (Fig 3.9 – code uploaded on canvas – source code chapter 03) to test method withdraw.

Answer:

withDraw method

```
public void withDraw(double withDrawAmount) {

    if (withDrawAmount > balance) {
        System.out.println("Withdrawal amount exceeded account balance");

    }
    else {
        balance = balance - withDrawAmount;
    }
}
```

test code:

```
public class Test {
    public static void main(String[] args) {
        Account account1 = new Account("Zhonghua Zhang", 100.01);

        // display initial balance of account 1
        System.out.printf("%s balance: $%.2f%n",
                account1.getName(), account1.getBalance());

        // create a Scanner to obtain input from the command window
        Scanner input = new Scanner(System.in);

        System.out.print("Enter withdraw amount for account1: "); // prompt
        double withDrwaAmount = input.nextDouble(); // obtain user input

        System.out.printf("%nremoving %.2f from account1 balance%n%n",
                withDrwaAmount);


        account1.withDraw(withDrwaAmount);
        // minus withdraw amount from balance


        // display balances
        System.out.printf("%s balance: $%.2f%n",
                account1.getName(), account1.getBalance());
    }
}
```

Result:

```
Zhonghua Zhang balance: $100.01
Enter withdraw amount for account1: 200

removing 200.00 from account1 balance

Withdrawal amount exceeded account balance
Zhonghua Zhang balance: $100.01
```

```
Zhonghua Zhang balance: $100.01
Enter withdraw amount for account1: 50

removing 50.00 from account1 balance

Zhonghua Zhang balance: $50.01
```

# Question 4:

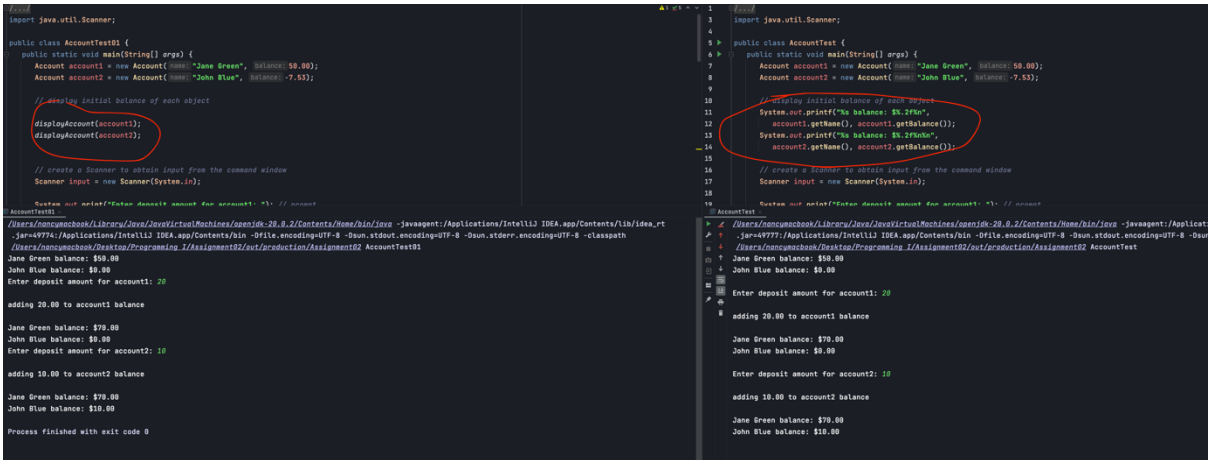
Display an account's information code:

```
public static void displayAccount(Account AccountToDisplay){
    System.out.printf("%s balance: $%.2f%n",
            AccountToDisplay.getName(),AccountToDisplay.getBalance());
}

call the method code:

displayAccount(account1);
displayAccount(account2);
```

result:



## Question 5

Car class

```
public class Car {
    String model;
    int year;
    double price;


    public void discount(double discount){

        price = price * (1 - discount/100);

    }

    public Car(String model, int year, double price) {
        this.model = model;
        this.year = year;
        if (price > 0.0) {
            this.price = price;
        } else {
            System.out.println("Car:" + this.model + "---invalid car
price");
        }

    }


    public String getModel() {
        return model;
    }

    public void setModel(String model) {
        this.model = model;
    }

    public int getYear() {
        return year;
    }

    public void setYear(int year) {
        this.year = year;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice(double price) {
        this.price = price;
    }
}
```

**test code:**

```
public class CarTest {
    public static void main(String[] args) {
        Car car1 = new Car ("ToyotaCamry", 2020, 30000);
        Car car2 = new Car("HondaCivic", 2019, 25000);
        displayCarInfo(car1);
        displayCarInfo(car2);

        System.out.println("Applying discounts...");

        car1.discount(5);
        System.out.printf("Car1 - New Price: $%.2f%n",
car1.getPrice());

        car2.discount(7);
        System.out.printf("Car2 - New Price: $%.2f%n",
car2.getPrice());
    }

    public static void displayCarInfo(Car displayCar){
        System.out.printf("Model:%s Year:%s Price:$%.2f%n",

displayCar.getModel(),displayCar.getYear(),displayCar.getPrice());
    }
}
```

**result :**

```
Model:ToyotaCamry Year:2020 Price:$30000.00
Model:HondaCivic Year:2019 Price:$25000.00
Applying discounts...
Car1 - New Price: $28500.00
Car2 - New Price: $23250.00
```