If you solved the task of previous week then you can skip those parts and start from creating Client and Bank classes and you can modify Account class when you need it.

Previous Week

1-) Create a C++ project first. We will use what we did last week.

2-) Create a header file (.h) named **account**. In this file you will have 3 attribute for the class **Account**;

- integer **id** **(private)**
- string **password** **(private)**
- float **balance**. **(private)**

Also you will have the functions;

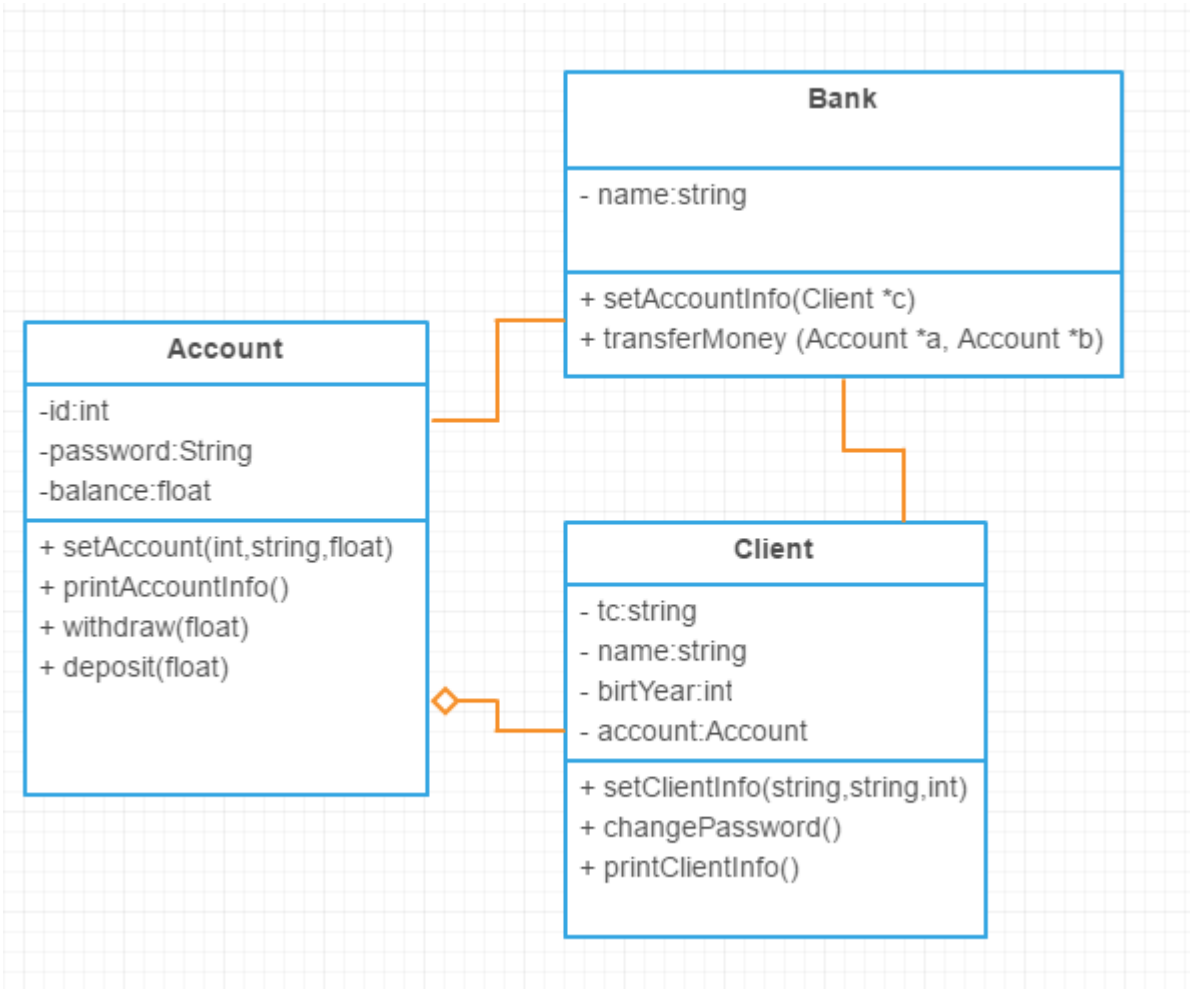
- constructor named **Account**
- **setAccount** function which will set accounts' info
- **printAccountInfo** function to show account info
- **withdraw** function to withdraw money.
- **deposit** function to deposit money.
- **You can write set and get methods if you want.**

3-) After you have the header file, you will create a C++ file (.cpp) named **account**. Then you will write a default constructor in it. You will put **0** to id as default and "**default"** value to password and **0.0** value for the balance. After that;

- You will implement the constructor.
- You will write the **setAccount** function and,
- You will write the **printAccount** function.
- You will write the **withdraw** function.
- You will write the **deposit** function.

After you did previous week task

4-) We will have another two class named **Client** and **Bank.** We will create header (.h) and (.cpp) files for them. Attributes of these classes can be seen into UML diagram below:



5-) You will create a C++ file (.cpp) named **test**. In this file you will have a main function;

- You will create two Account objects
- You will create a Bank object
- You will take tc, name and birth year from the user for each Account. (you can write a function for it in main).
- By using bank object setAccuntInfo function you will take password and balance from the user and id will be given by bank starting with 1 and incrementing 1 for each account.
- You will try to change password of one account that you created. (For this part you might need to add some attributes to Account class)
- By using bank object, you will transfer money from one account to another. (You might need to modify Account for this option also.)

Sample Run:

```
Enter your TC:123
Enter your name: Client1
Enter your birth year: 1990
Welcome to ANKARA BANK
Enter your password:user1
Enter your balance:538.65
TC: 123
Name: Client1
Birth Year: 1990

ID: 1
Password: user1
Balance: 538.65

Enter your TC:321
Enter your name: Client2
Enter your birth year: 1992
Welcome to ANKARA BANK
Enter your password:user2
Enter your balance:32.90
TC: 321
Name: Client2
Birth Year: 1992

ID: 2
Password: user2
Balance: 32.9
```

```
Enter your old password: user1
Enter your new password: newPass1
Enter your new password again: newPass2
Passwords are not matched!!
TC: 123
Name: Client1
Birth Year: 1990

ID: 1
Password: user1
Balance: 538.65

Enter the amount that you want to transfer: 235
TC: 123
Name: Client1
Birth Year: 1990

ID: 1
Password: user1
Balance: 303.65

TC: 321
Name: Client2
Birth Year: 1992

ID: 2
Password: user2
Balance: 267.9
```

Bonus: You can **check the name of the account that you will send money** for **transferMoney** operations as a bonus for this question. You can check **tc and birthyear** when bank **settingAccountInfo**.
Also, you can add the features that you want to add.