

# Encapsulation\_Lab2

May 8, 2020

```
In [ ]: #include <iostream>
#include <string>
using std::cout;
using std::string;

class BankAccount{
private:
    // TODO: declare member variables
    string accountNumber;
    string name;
    double fund;
public:
    BankAccount(string accountNumber, string name, double fund);
    // TODO: declare setters
    void setAccountNumber(string account);
    void setName(string n);
    void setFund(double f);
    // TODO: declare getters
    string getAccountNumber() const;
    string getName() const;
    double getFund() const;
};

BankAccount::BankAccount(string accountNumber, string name, double fund){
    setAccountNumber(accountNumber);
    setName(name);
    setFund(fund);
}

// TODO: implement setters
void BankAccount::setAccountNumber(string account){
    accountNumber = account;}
void BankAccount::setName(string n){name = n;}
void BankAccount::setFund(double f){fund = f;}

// TODO: implement getters
string BankAccount::getAccountNumber() const{return accountNumber;}
```

```

    string BankAccount::getName() const{return name;}
    double BankAccount::getFund() const{return fund;}

int main(){
    // TODO: instantiate and output a bank account
    BankAccount Account("3203410", "Xin Zhao", 999);
    cout<<"Account Information:\n";
    cout<<"-----\n";
    cout<<"Account ID: "<<Account.getAccountNumber()<<"\n";
    cout<<"Account Name: "<<Account.getName()<<"\n";
    cout<<"Account Amount: $"<<Account.getFund()<<"\n\n";
    Account.setFund(9999);
    cout<<"Account Information:\n";
    cout<<"-----\n";
    cout<<"Account ID: "<<Account.getAccountNumber()<<"\n";
    cout<<"Account Name: "<<Account.getName()<<"\n";
    cout<<"Account Amount: $"<<Account.getFund()<<"\n\n";
}

```

Compile & Run

Explain

Loading terminal (id\_4ra3ew2), please wait...

Hide Solution

```

In [ ]: // Example solution for creating a BankAccount class
#include <iostream>
#include <string>

class BankAccount
{
private:
    // Class attributes:

    long int number;
    std::string owner;
    float amount;

public:
    // Set methods:
    void setNumber(long int number);
    void setOwner(std::string owner);
    void setAmount(float amount);
    // Get methods:
    long int getNumber() const;
    std::string getOwner() const;
    float getAmount() const;
};

```

```

// Implementation of Set methods:
void BankAccount::setNumber(long int number) {
    // Changing attribute to incoming value
    BankAccount::number = number;
}

void BankAccount::setOwner(std::string owner) {
    BankAccount::owner = owner;
}

void BankAccount::setAmount(float amount) {
    BankAccount::amount = amount;
}

// Implementation of Get methods:
long int BankAccount::getNumber() const {
    // Getting specified attribute
    return BankAccount::number;
}

std::string BankAccount::getOwner() const {
    return BankAccount::owner;
}

float BankAccount::getAmount() const {
    return BankAccount::amount;
}

// main function
int main(){
    BankAccount ba;
    ba.setAmount(100);

    std::cout << ba.getAmount() << std::endl;
}

```