Algorithms Level 3

PROGRAMMING ADVICES

LEARN THE RIGHT WAY

26+ Years of Experience

Mohammed Abu-Hadhoud

MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD

ProgrammingAdvices.com

```cpp
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include <iomanip>

using namespace std;
const string ClientsFileName = "Clients.txt";

struct sClient
{
    string AccountNumber;
    string PinCode;
    string Name;
    string Phone;
    double AccountBalance;

};

vector<string> SplitString(string S1, string Delim)
{

    vector<string> vString;

    short pos = 0;
    string sWord; // define a string variable

    // use find() function to get the position of the delimiters
    while ((pos = S1.find(Delim)) != std::string::npos)
    {
        sWord = S1.substr(0, pos); // store the word
        if (sWord != "")
        {
            vString.push_back(sWord);
        }

        S1.erase(0, pos + Delim.length());
    }

    if (S1 != "")
    {
        vString.push_back(S1); // it adds last word of the string.
    }

    return vString;

}
```

```cpp
sClient ConvertLinetoRecord(string Line, string Seperator =
"#//#")
{

    sClient Client;
    vector<string> vClientData;

    vClientData = SplitString(Line, Seperator);

    Client.AccountNumber = vClientData[0];
    Client.PinCode = vClientData[1];
    Client.Name = vClientData[2];
    Client.Phone = vClientData[3];
    Client.AccountBalance = stod(vClientData[4]);//cast string to
double

    return Client;
}

vector <sClient> LoadCleintsDataFromFile(string FileName)
{
    vector <sClient> vClients;
    fstream MyFile;
    MyFile.open(FileName, ios::in);//read Mode

    if (MyFile.is_open())
    {

        string Line;
        sClient Client;

        while (getline(MyFile, Line))
        {

            Client = ConvertLinetoRecord(Line);

            vClients.push_back(Client);
        }

        MyFile.close();

    }

    return vClients;

}
```

```cpp
void PrintClientRecord(sClient Client)
{

    cout << "| " << setw(15) << left << Client.AccountNumber;
    cout << "| " << setw(10) << left << Client.PinCode;
    cout << "| " << setw(40) << left << Client.Name;
    cout << "| " << setw(12) << left << Client.Phone;
    cout << "| " << setw(12) << left << Client.AccountBalance;

}


void PrintAllClientsData(vector <sClient> vClients)
{

    cout << "\n\t\t\t\t\tClient List (" << vClients.size() << ")
Client(s).";
    cout <<
"\n_____";
    cout << "_____\n" << endl;

    cout << "| " << left << setw(15) << "Accout Number";
    cout << "| " << left << setw(10) << "Pin Code";
    cout << "| " << left << setw(40) << "Client Name";
    cout << "| " << left << setw(12) << "Phone";
    cout << "| " << left << setw(12) << "Balance";
    cout <<
"\n_____";
    cout << "_____\n" << endl;

    for (sClient Client : vClients)
    {

        PrintClientRecord(Client);
        cout << endl;
    }

    cout <<
"\n_____";
    cout << "_____\n" << endl;

}
```

```cpp
int main()

{
    vector <sClient> vClients =
LoadCleintsDataFromFile(ClientsFileName);

    PrintAllClientsData(vClients);

    system("pause>0");
    return 0;
}
```