



Code de création de compte bancaire

SOMMAIRE

- Explication du code
- Client.cpp
- Client.h
- Compte.cpp (1)
- Compte.cpp (2)
- Compte.cpp (3)
- Compte.h
- Main.cpp
- Remerciment



Explication du code

Dans ce diaporama, je vais vous expliquer comment j'ai créer ce code.

Création de compte en banque

Ce code permet de :

- créditer
- débiter
- transférer
- afficher les infos



Client.cpp

Déclaration du constructeur de la classe qui prend quatre paramètres : identifiant, nomFamille, prenomClient et telephone. Ce constructeur initialise les attributs de l'objet.

initialisation des attributs

~client : Déclaration du destructeur de la classe, qui libérera les ressources lorsque l'objet sera détruit.

Déclarations des fonctions.

```
client.cpp > ...
1  #include <iostream>
2  #include <cstring>
3  #include "client.h"
4
5  using namespace std;
6
7  Client::Client(string identifiant, string nomFamille, string prenomClient, string telephone) {
8      CIN = identifiant;
9      nom = nomFamille;
10     prenom = prenomClient;
11     numero = telephone;
12 }
13
14 Client::~~Client() {
15 }
16
17
18 string Client::getCIN() {
19     return CIN;
20 }
21
22 string Client::getNom() {
23     return nom;
24 }
25
26 string Client::getPrenom() {
27     return prenom;
28 }
29
30 string Client::getTelephone() {
31     return numero;
32 }
33
```


Client.h

Ce code définit la classe avec des attributs privés pour stocker l'identifiant (CIN), le nom, le prénom et le numéro de téléphone. Il comprend un constructeur pour initialiser ces attributs, un destructeur, et plusieurs méthodes publiques pour récupérer les valeurs des attributs. Il contient également une méthode pour modifier le numéro de téléphone, bien que sa définition ne soit pas encore fournie. Le fichier d'en-tête utilise des directives de préprocesseur pour éviter les inclusions multiples.

```
C client.h > Client > getTelephone()
1  #ifndef CLIENT_H
2  #define CLIENT_H
3
4  #include <iostream>
5  #include <string>
6
7  using namespace std;
8
9  class Client {
10 private:
11     string CIN;
12     string nom;
13     string prenom;
14     string numero;
15
16 public:
17     Client(string, string, string, string);
18     ~Client();
19
20     string getCIN();
21     string getNom();
22     string getPrenom();
23     string getTelephone();
24 };
25
26 #endif
```


Compte.cpp (1)

Ce code définit la classe Compte, qui représente un compte bancaire. Le constructeur initialise l'identifiant du client, le solde et le propriétaire du compte. La classe inclut des méthodes pour gérer les opérations bancaires : Crediter() pour déposer de l'argent, Debiter() pour retirer des fonds, et Transfert() pour transférer de l'argent. Il y a également une méthode AfficherInfos() pour afficher les détails du compte et une méthode menu() qui présente un menu interactif à l'utilisateur pour sélectionner l'opération souhaitée. Des vérifications sont effectuées pour s'assurer que les montants sont valides et que le solde est suffisant pour les opérations.

```
compte.cpp > Crediter()
1  #include <iostream>
2  #include "compte.h"
3
4  using namespace std;
5
6  Compte::Compte(string idClient, double montantInitial, Client clientProprio)
7  |   : ID(idClient), solde(montantInitial), proprietaire(clientProprio) {
8  |   }
9
10  Compte::~Compte() {
11  |   }
12
13  double Compte::getSolde() {
14  |   return solde;
15  |   }
16
17  string Compte::getIban() {
18  |   return iban;
19  |   }
20
21  void Compte::Crediter() {
22  |   system("CLS");
23  |   cout << "\n\tDEPOT D'ARGENT\n\n";
24  |   cout << "Entrez le montant a deposer : ";
25  |   double montant;
26  |   cin >> montant;
27  |   if (montant > 0) {
28  |       solde += montant;
29  |       cout << "\nARGENT AJOUTE ! Solde actuel : " << solde << " euros\n";
30  |   } else {
31  |       cout << "\nERREUR : Montant invalide\n";
32  |   }
33  |   system("pause");
34  |   }
35
36  void Compte::Debiter() {
37  |   system("CLS");
```


Compte.cpp (2)

```
compte.cpp > Crediter()

36 void Compte::Debiter() {
37     system("CLS");
38     cout << "\n\tRETRAIT D'ARGENT\n\n";
39     cout << "Entrez le montant a retirer : ";
40     double montant;
41     cin >> montant;
42     if (montant > 0 && montant <= solde) {
43         solde -= montant;
44         cout << "\nARGENT RETIRE ! Solde actuel : " << solde << " euros\n";
45     } else {
46         cout << "\nERREUR : Montant invalide ou insuffisant\n";
47     }
48     system("pause");
49 }

50
51 void Compte::Transfert() {
52     system("CLS");
53     cout << "\n\tTRANSFERT D'ARGENT\n";
54     cout << "Montant du transfert : ";
55     double montant;
56     cin >> montant;
57     if (montant > 0 && montant <= solde) {
58         solde -= montant;
59         cout << "\nTRANSFERT REUSSI ! Solde actuel : " << solde << " euros\n";
60     } else {
61         cout << "\nERREUR : Montant incorrect\n";
62     }
63     system("pause");
64 }

65
66 void Compte::AfficherInfos() {
67     system("CLS");
68     cout << "\n\tDETAILS DU COMPTE\n\n";
69     cout << "Nom du titulaire : " << proprietaire.getNom() << endl;
70     cout << "Prenom du titulaire : " << proprietaire.getPrenom() << endl;
71     cout << "Telephone : " << proprietaire.getTelephone() << endl;
72     cout << "Solde disponible : " << solde << " €" << endl;
```


Compte.cpp (3)

```
71     cout << "Telephone : " << proprietaire.getTelephone() << endl;
72     cout << "Solde disponible : " << solde << " €" << endl;
73     system("pause");
74 }
75
76 void Compte::menu() {
77     int choix = 0;
78     while (choix != 5) {
79         system("CLS");
80         cout << "\n--- MENU DU COMPTE ---\n";
81         cout << "1. Crediter\n2. Debiter\n3. Transfert\n4. Afficher Infos\n5. Quitter\n\nVotre choix : ";
82         cin >> choix;
83         switch (choix) {
84             case 1: Crediter(); break;
85             case 2: Debiter(); break;
86             case 3: Transfert(); break;
87             case 4: AfficherInfos(); break;
88             case 5: break;
89             default: cout << "Option incorrecte\n"; break;
90         }
91     }
92 }
93
```


Compte.h

Ce code définit la classe, qui représente un compte bancaire. Elle contient des attributs privés pour l'identifiant du compte (), le solde, l'IBAN, et le propriétaire (un objet de la classe). La classe comprend un constructeur pour initialiser ces attributs, un destructeur, ainsi que des méthodes publiques pour récupérer l'IBAN et le solde, ainsi que pour gérer les opérations bancaires telles que le dépôt, le retrait, le transfert d'argent et l'affichage des informations du compte. Un menu interactif est également prévu pour faciliter l'interaction de l'utilisateur avec le compte.

```
C compte.h > ...
1  #ifndef COMPTE_H
2  #define COMPTE_H
3
4  #include <iostream>
5  #include <string>
6  #include "client.h"
7
8  using namespace std;
9
10 class Compte {
11 private:
12     string ID;
13     double solde;
14     string iban;
15     Client proprietaire;
16
17 public:
18     Compte(string, double, Client);
19     ~Compte();
20
21     string getIban();
22     double getSolde();
23
24     void menu();
25     void Crediter();
26     void Debiter();
27     void Transfert();
28     void AfficherInfos();
29 };
30
```


Main.cpp

Ce code constitue le point d'entrée d'un programme de gestion de comptes bancaires. Il affiche un menu permettant à l'utilisateur de choisir entre deux clients, Amir SBIAI et Ryiad Mahrez. Selon le choix, un objet et un objet sont créés avec des informations spécifiques. L'utilisateur est ensuite invité à entrer un code PIN pour accéder aux fonctionnalités du compte. Si le code est correct, le menu des opérations du compte s'affiche. Sinon, un message d'erreur est affiché. Si le choix de client est invalide, un message d'erreur est également donné.

```
main.cpp > main()
1  #include <iostream>
2  #include "client.h"
3  #include "compte.h"
4
5  using namespace std;
6
7  int main() {
8      system("CLS");
9      cout << "1. Amir SBIAI\n2. Ryiad Mahrez\nVotre choix : ";
10     int choix;
11     cin >> choix;
12
13     if (choix == 1) {
14         Client client1("FR-KHZN45379844", "Amir", "SBIAI", "06153485795");
15         Compte compte1("00001", 500.0, client1);
16         cout << "Entrez le code PIN : ";
17         int code;
18         cin >> code;
19         if (code == 1111) {
20             compte1.menu();
21         } else {
22             cout << "Code incorrect\n";
23         }
24     } else if (choix == 2) {
25         Client client2("FR-LDEN41256687", "Ryiad", "Mahrez", "0645278845");
26         Compte compte2("00002", 400.0, client2);
27         cout << "Entrez le code PIN : ";
28         int code;
29         cin >> code;
30         if (code == 2222) {
31             compte2.menu();
32         } else {
33             cout << "Code incorrect\n";
34         }
35     } else {
36         cout << "Choix incorrect\n";
37     }
```


Merci !

