

Notes on Classes in C++

General Form of a Class Declaration

```
class ClassName  
{  
    private:  
  
    public:  
  
};
```

Access
Specifiers



```
class BankAccount  
{  
    private:  
        double balance;  
  
    public:  
        BankAccount()  
        { balance = 0; }  
  
        BankAccount(double startingBal)  
        { balance = startingBal; }  
  
        void deposit (double amt)  
        { balance += amt; }  
  
        void withdraw(double amt)  
        { balance -= amt; }  
  
        double getBalance()  
        { return balance; }  
};
```

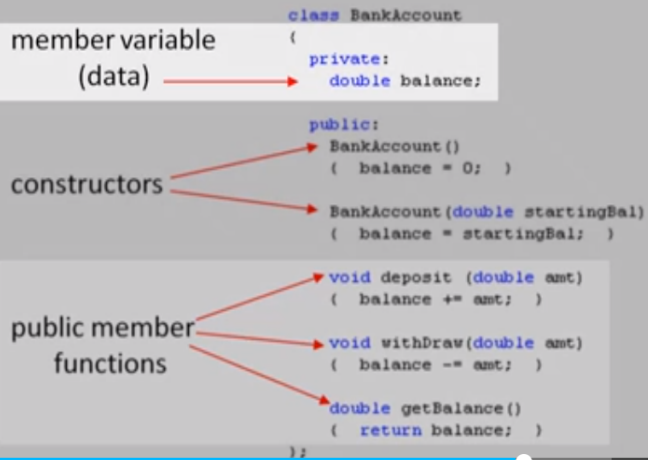
BankAccount acct1; [0]

Constructors

BankAccount acct2(500);

acct2 [500]





```
09:24 / 13:19
class BankAccount
{
private:
    double balance;

public:
    BankAccount()
    { balance = 0; }

    BankAccount(double startingBal)
    { balance = startingBal; }

    void deposit (double amt)
    { balance += amt; }

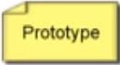
    void withdraw(double amt)
    { balance -= amt; }

    double getBalance()
    { return balance; }
};

11:10 / 13:19
6 {
7 private:
8     double balance;
9
10 public:
11     BankAccount()
12     { balance = 0; }
13
14     BankAccount(double startingBal)
15     { balance = startingBal; }
16
17     void deposit (double amt)
18     { balance += amt; }
19
20     void withdraw(double amt)
21     { balance -= amt; }
22
23     double getBalance()
24     { return balance; }
25 };
26
27 // Client program that uses the BankAccount class
```

Inline functions

```
BankAcct2.cpp
(Global Scope)
4 // BankAccount class declaration
5 class BankAccount
6 {
7     private:
8         double balance;
9
10    public:
11        BankAccount();
12        BankAccount(double);
13        void deposit(double);
14        void withdraw(double);
15        double getBalance();
16    };
17
18 // BankAccount class function definitions
19 BankAccount::BankAccount()
20 {
21     balance = 0;
22 }
23
24 BankAccount::BankAccount(double startingBal)
25 {
26     balance = startingBal;
27 }
28
29 void BankAccount::deposit(double amt)
```



The double colons are known as:

```
BankAcct2.cpp
(Global Scope)
4 // BankAccount class declaration
5 class BankAccount
6 {
7     private:
8         double balance;
9
10    public:
11        BankAccount();
12        BankAccount(double);
13        void deposit(double);
14        void withdraw(double);
15        double getBalance();
16    };
17
18 // BankAccount class function definitions
19 BankAccount::BankAccount()
20 {
21     balance = 0;
22 }
23
24 BankAccount::BankAccount(double startingBal)
25 {
26     balance = startingBal;
27 }
28
29 void BankAccount::deposit(double amt)
```

