**Ограничение доступа к членам класса**

В объявлении класса используются и другие ключевые слова. Двумя самыми важными из них являются public (открытый) и private (закрытый), определяющие доступ к членам класса.

Все члены класса — данные и методы — являются закрытыми по умолчанию. К закрытым членам можно получить доступ только с помощью методов самого класса. Открытые члены доступны для всех других функций программы. Определение доступа

к членам класса имеет очень важное значение, и именно при решении этой задачи начинающие программисты часто сталкиваются с трудностями. Чтобы прояснить ситуацию, рассмотрим пример, который уже приводился выше в этой главе:

class Cat {

unsigned int itsAge;

unsigned int itsWeight;

void Meow();

};

В этом объявлении переменные itsAge и itsWeight, а также метод Meow() являются закрытыми, поскольку все члены класса закрытые по умолчанию. Если требуется изменить доступ к членам класса, то это следует сделать явно.

Если в программе будет описан класс Cat, как показано выше, то обращение к переменной-члену itsAge из функции main() вызовет ошибку компиляции:

Cat Boots;

Boots.itsAge = 5; // Ошибка! Нельзя обращаться к закрытым данным

И в самом деле, сначала компилятору указывается, что члены itsAge, itsWeight и Meow() можно использовать только внутри класса Cat, а затем делается попытка использовать во внешней функции переменную-член itsAge, безраздельно принадлежащую объекту Boots класса Cat. Хотя объект Boots реально существует в программе, это не означает, что можно получать доступ к членам данного объекта, закрытым для постороннего глаза.

Именно эти моменты с определением доступа к членам класса служат источником бесконечных недоразумений у начинающих программистов. Я прямо-таки слышу ваш удивленный вопрос: "Если в программе объявлен реальный объект Boots класса Cat, почему же нельзя присвоить значение переменной-члену этого объекта, даже обратившись к ней с помощью оператора прямого доступа?"

Дело в Том, что в объявлении класса Cat ничего не говорится о ваших правах обращаться к членам этого класса, а это значит, что вы таких прав не имеете. Только собственные методы объекта Boots всегда имеют доступ ко всем данным класса, как открытым, так и закрытым. Даже несмотря на то, что вы сами создали класс Cat, это не дает вам права возвращать или изменять в программе его данные, которые являются закрытыми.

Однако из любого положения есть выход. Чтобы получить доступ к переменным- членам класса Cat, откройте их следующим способом:

class Cat {

public:

unsigned int itsAge;

unsigned int itsWeight;

void Meow();

};

Согласно общей стратегии использования классов переменные-члены класса следует оставлять закрытыми. Благодаря этому достигается инкапсуляция данных внутри класса. Доступ следует открывать только к функциям-членам класса, обеспечивающим доступ к его закрытым данным (эти функции еще называют методами доступа). Эти методы можно вызывать из любого места в программе для возвращения или установки значений закрытых переменных-членов.