

Статические свойства и методы и сокрытие данных

Информатика
10-11 классы

3 апреля 2012 г.

Скрытие данных

- Одним из важнейших свойств инкапсуляции является *сокрытие методов*.
- Пример (rubydev): для пояснения термина нам помогут самые обычные хоббиты.
- Они дружелюбны и открыты для других, но и у них есть также личная жизнь.
- Многое из их личной жизни доступно для других хоббитов так как в своем хоббитовом царстве они могут бродить по жилищам друг друга.
- Конечно, каждый хоббит имеет что-то очень личное, такое, что знает один лишь он и никому и никогда не раскроет своей тайны.
- Такие особенности устройства поведения хоббитов очень похожи на методы в Ruby.
- Давайте представим каждого хоббита как объект.

Класс Хоббит

Listing 1: Класс Хоббит

```
class Hobbit
  def initialize(name, rooms, has_ring)
    @name, @rooms, @has_ring = name, rooms, has_ring
  end

  def name
    @name
  end

  def name_of(hobbit)
    hobbit.name
  end

  def rooms_of(hobbit)
    hobbit.rooms
  end
end
```

Класс Хоббит

Listing 2: Класс Хоббит

```
def hobbit_has_ring?(hobbit)
  hobbit.has_ring?
end

protected

def rooms
  @rooms
end

private

def has_ring?
  @has_ring
end
end
```

Что мы получили?

- Итого, в нашем классе Хоббит есть 5 методов.
- Три из них расположены в публичной (public) зоне, которая всегда идёт по умолчанию: `name_of`, `rooms_of`, `hobbit_has_ring?`
- Один метод является защищённым (protected): `rooms`.
- Ещё один метод — приватным (private): `has_ring?`

Используем хоббитов

- Создадим пару хоббитов и выведем их имена.

Listing 3: Использование Хоббита

```
frodo = Hobbit.new('Frodo', 3, true)  
samwise = Hobbit.new('Samwise', 2, false)  
  
puts frodo.name  
puts samwise.name
```

Защищённые методы

- А если мы хотим узнать количество комнат у хоббита?
- Простой вызов **frodo.rooms** вызовет ошибку, так как метод `rooms` является защищённым.
- Защищённые методы можно вызывать только изнутри объекта. При вызове снаружи возникает ошибка.
- На нашем примере: хоббиты скрывают количество комнат в своих домах от других живых существ, но между собой у них в этом секрета нет.
- Любой хоббит может узнать количество комнат себя или другого хоббита.

Количество комнат

Listing 4: Количество комнат

```
frodo    = Hobbit.new('Frodo', 3, true)
samwise  = Hobbit.new('Samwise', 2, false)

puts frodo.rooms_of(frodo)
puts frodo.rooms_of(samwise)
```


А что насчёт кольца?

- Кольцо расположено в приватной области.
- При попытке вызвать метод `has_ring` или `hobbit_has_ring` возникает ошибка:

Listing 5: Ошибка при вызове

```
...  
frodo.has_ring?  
frodo.hobbit_has_ring?(frodo)  
frodo.hobbit_has_ring?(samwise)
```

Приватные методы

- Доступ к приватным методам возможен только из самого объекта.
- Извне ни к своим, ни к чужим методам доступ получить нельзя.
- Условно говоря, приватные свойства и методы — это та информация, которую знает только один конкретный экземпляр класса.

Статические свойства и методы

- В обычном случае мы вызываем методы у объектов.
- То есть, процедура такая: написать класс, создать объект, вызвать метод.
- Однако методы могут быть не только у объекта, но и у класса в целом.
- По своей логике они напоминают обычные функции, в названии которых дополнительно встречается приставка с именем класса.

Пример



Listing 6: Пример статического метода

```
class Hobbit

  def self.location
    return "Shire"
  end

end

puts Hobbit.location
```

Статические переменные

- Статические переменные — это переменные, которые существуют у класса в единственном экземпляре.
- То есть, разные экземпляры класса будут иметь одну и ту же переменную.

Пример программы

Listing 7: Статическое свойство

```
class Hobbit
  attr_accessor :number
  @@number = 0
  def initialize
    @@number+=1
  end
  def show
    puts "#{@@number}"
  end
end
frodo = Hobbit.new
frodo.show
samwise = Hobbit.new
samwise.show
frodo.show
```

Задание

- Написать эссе на тему: Что такое Синглтон (Singleton).
Реализовать на синглтон на ruby и продемонстрировать работу.

References

- При подготовке данного материала использовались сайты:
<http://ru.wikibooks.org/wiki/Ruby>, <http://rubydev.ru>,
<http://en.wikipedia.org>, <http://ruby-lang.org>.
- Все презентации доступны на <http://school.smirik.ru>!
- Вопросы, предложения, д/з: smirik@gmail.com