**Что есть объект**

Нас окружает множество объектов. У многих из нас сформировано бытовое понятие объекта, которое достаточно близко к философскому (объект есть нечто, на что направлена деятельность), поэтому не будем заострять на нём внимание.

В программировании **объект** — это некоторая сущность в цифровом пространстве, обладающая определёнными признаками. В целом, объект в программировании является описанием-отражением реального или виртуального объекта.

Описание объекта представляет собой набор признаков:

* + свойств;
  + методов;
  + состояний;
  + поведения.

В первую очередь стоит различать свойства (*данные*) и методы (*действия*), так как их реализация существенно отличается. Состояние и поведение являются производными от этих признаков.

Для рассмотрения возьмем объект «автомобиль». Обозначим его следующим образом:

myCar {

}

При описании объекта в любом языке программирования следует соблюдать некоторые правила:

* Абстрагируйтесь от реального объекта. Некоторые признаки существенны с точки зрения поставленной задачи, а некоторые — нет. Описывайте только существенные признаки;
* Помещайте все признаки объекта в сам объект. Если какое-то действие над объектом находится вне его — оно не относится к объекту.

**Свойства**

Все данные объекта хранятся в переменных. Переменные при этом связаны с объектом. Такие переменные называются **свойствами** (*property*) объекта.

У автомобиля такими свойствами могут являться цвет, марка, стоимость, модель.

myCar {

brand: 'Toyota',

model: 'Corolla Fielder',

color: 'red',

manufactured: **2004**,

price: **320000.00**

}

## ****Методы****

**Метод** (method) — это функция (подпрограмма), принадлежащая объекту.

Методы определяют возможные действия в отношении объекта. Например, для автомобиля можно определить метод «покрасить», который изменит свойство «цвет»:

myCar {

brand: 'Toyota',

model: 'Corolla Fielder',

color: 'red',

manufactured: **2004**,

price: **320000.00**

**function** paint($color) {

**return** **this**.color = $color;

}

}

В зависимости от того, какой цвет будет передан в метод, при этом будет изменено свойство color у нашего объекта. В этом случае цвет можно считать состоянием.

* управляющие конструкции:
  + - присваивание мы обозначили через знак =;
    - переменные выделили знаком $;
    - подпрограмму обозначили через *function*, а возврат результата — через *return*;
    - ввели псевдообъект this, который ссылается на сам объект.

## ****Состояние****

**Состояние** (state) объекта — это набор изменяющихся во времени свойств. Состояние неизменно до тех пор, пока над объектом не будет произведено действие; если над объектом будет произведено некоторое действие, его состояние может измениться. Обычно состояние влияет и на поведение объекта.

## ****Поведение****

Для автомобиля, например, можем выделить состояние broken (сломан). Если состояние broken истинно, то автомобиль иначе отреагирует на метод drive (ехать). Зачастую поведение зависит от свойств.

**Поведение** (behavior) — это то, как объект действует и реагирует.

## ****Класс как множество объектов****

При написании программы часто требуется использовать множество похожих объектов. Для того, чтобы не производить описание всех признаков таких объектов многократно, используются классы.

Класс представляет собой шаблон для создания объектов, обеспечивающий начальные значения свойств и состояний, а также описание методов и поведений.

Например, все автомобили имеют общие признаки, поэтому разумнее создать класс для того, чтобы создавать несколько автомобилей в системе на основе этого класса.

### Задание 2.5.1

1/1 point (graded)

Какие свойства типичны для объекта «карандаш»?

Фамилия

Цвет

Длина

Перемещение

Стоимость

Вид двигателя

верно

### Задание 2.5.2

1/1 point (graded)

Какие свойства типичны для объекта «файл»?

Наименование

Расширение

Объем

Редактирование

Удаление

Автор

Цвет

верно