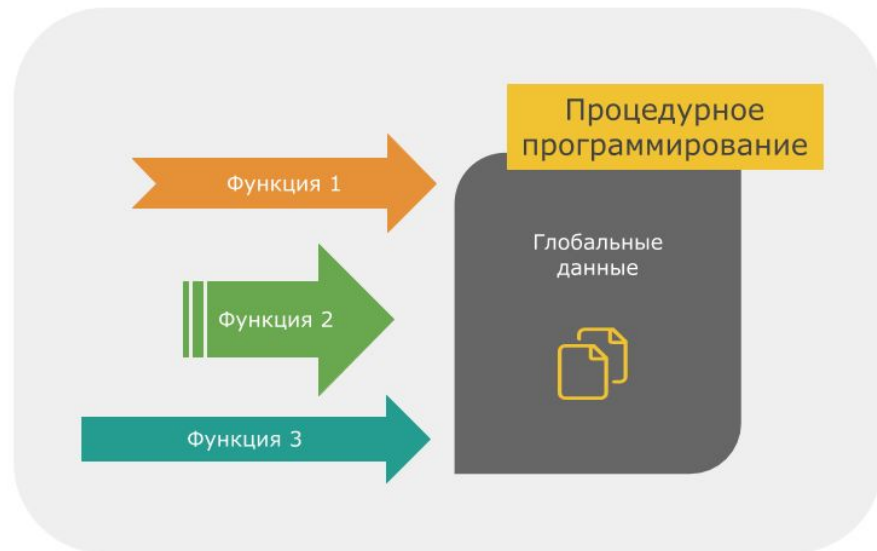

Объектно-ориентированное программирование

— Принципы ООП. Классы, объекты,
поля и методы. Уровни доступа. —

Что такое ООП?

Объектно-ориентированное программирование (ООП) — парадигма программирования, в которой основными концепциями являются понятия объектов и классов.



В чем суть процедурного подхода?

Процедурное программирование – это написание функций и их последовательный вызов в некоторой главной(main) функции.

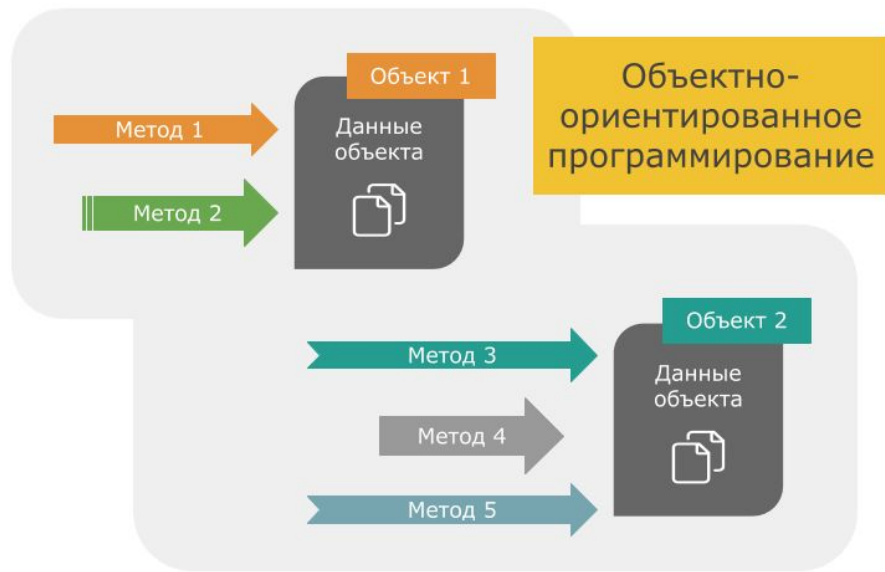
Идеальные условия для применения данного подхода - простые программы, где весь функционал можно реализовать несколькими десятками процедур/функций. Функции аккуратно вложены друг в друга и легко взаимодействуют посредством передачи данных из одной функции в другую.

Недостатки данного подхода

- В больших проектах приходится создавать огромное количество процедур и функций. В свою очередь, это неизбежно ведет к возникновению множества ошибок и снижает читаемость кода программы.
- Все данные внутри процедуры видны только локально, а значит их нельзя использовать в другом месте.
- Как следствие, код наполняется дубликатами.
- Высокий порог вхождения - по статистике начинающим данный подход дается сложнее, чем ООП.
- Долгое время процедурный подход был довольно популярным и считался самым прогрессивным.

Как определяет данный подход Википедия

Объектно-ориентированное программирование (ООП) — методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.



Преимущества ООП

1. Программа разбивается на объекты.
1. Уменьшается дубликация кода.
1. Упрощается и ускоряется процесс написания программ.

Подведем итог

1. В процедурном подходе основой программы является функция. Функции вызывают друг друга и при необходимости передают данные.
2. Основной недостаток процедурного подхода - сложность создания и поддержки больших программ.
3. В основе объектно-ориентированного программирования лежит понятие объекта. Объект совмещает в себе и функции и данные.
4. Основное преимущество ООП перед процедурным программированием - изоляция кода на уровне классов, что позволяет писать более простой и лаконичный код.

Объекты и классы в ООП

Термины

Класс — в объектно-ориентированном программировании, представляет собой шаблон для создания объектов, обеспечивающий начальные значения состояний: инициализация полей-переменных и реализация поведения функций или методов.

Объект — некоторая сущность в цифровом пространстве, обладающая определённым состоянием и поведением, имеющая определенные свойства (атрибуты) и операции над ними (методы).

Термины «экземпляр класса» и «объект» взаимозаменяемы.

1. Класс описывает множество объектов, имеющих общую структуру и обладающих одинаковым поведением. **Класс** - это шаблон кода, по которому создаются объекты. Т. е. сам по себе класс ничего не делает, но с его помощью можно создать объект и уже его использовать в работе.
2. Данные внутри класса делятся на свойства и методы. Свойства класса (они же поля или атрибуты) - это характеристики объекта класса.
3. **Методы класса** - это функции, с помощью которых можно оперировать данными класса.
4. **Объект** - это конкретный представитель класса.
5. Объект класса и экземпляр класса - это одно и то же.



Грубо говоря

Класс = Свойства + Методы

Пример

Есть класс “Автомобиль”, он представляет собой набор чертежей и инструкций, с помощью которых можно собрать машину.

При этом каждая машина, которую мы будем собирать, должна обладать рядом характеристик, которые соответствуют нашему классу. Как мы уже выяснили, данные характеристики - называются свойствами класса и в нашем примере могут быть следующими:

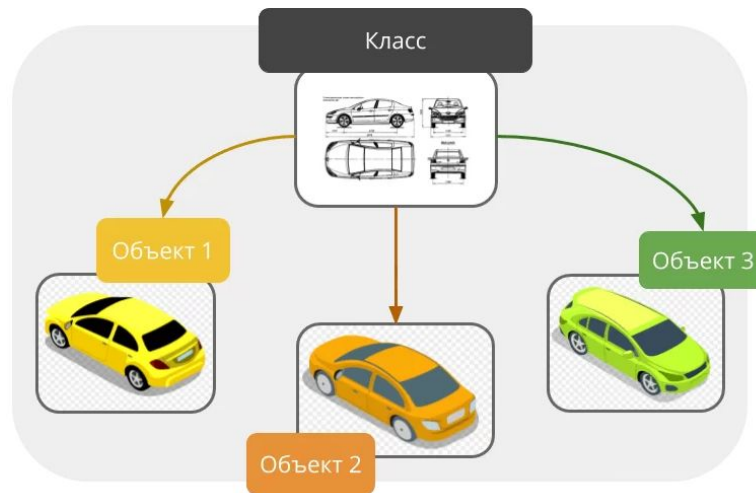
- Цвет
- Объем двигателя
- Мощность
- Тип коробки передач

Также наш автомобиль может выполнять какие-то действия, характерные для всего класса. Эти действия, как мы теперь знаем, есть методы класса.

Что общего в объектах из примера?

Все объекты создаются по одному шаблону, то есть на выходе обязательно будут машины, никаких велосипедов и мотоциклов.

Они будут покрашены в какой-то цвет, ехать они будут за счет наличия в них двигателя, скорость будет регулироваться с помощью коробки передач. Также объекты данного класса будут обладать одинаковыми **методами**.



Пример созданного объекта "Автомобиль Volkswagen Tiguan":

1. Свойства: Цвет="Белый", Объем двигателя="1984 см³", Мощность="180 л.с.", Тип коробки передач="Робот"
2. Методы: Ехать, Остановиться, Заправиться, Поставить на сигнализацию, Включить дворники

Домашнее задание

1. Что такое процедурное программирование?
2. Какие проблемы в программировании привели к появлению ООП?
3. Объясните, почему один и тот же объект может иметь много разных моделей.
4. Какие преимущества даёт объектный подход в программировании?
5. Зачем использовать ООП?

Спасибо за внимание!