

Основные принципы объектно-ориентированного программирования (ООП)

Разница процедурного программирования и ООП



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



ООП — это парадигма разработки программных систем, в которой приложения состоят из объектов.

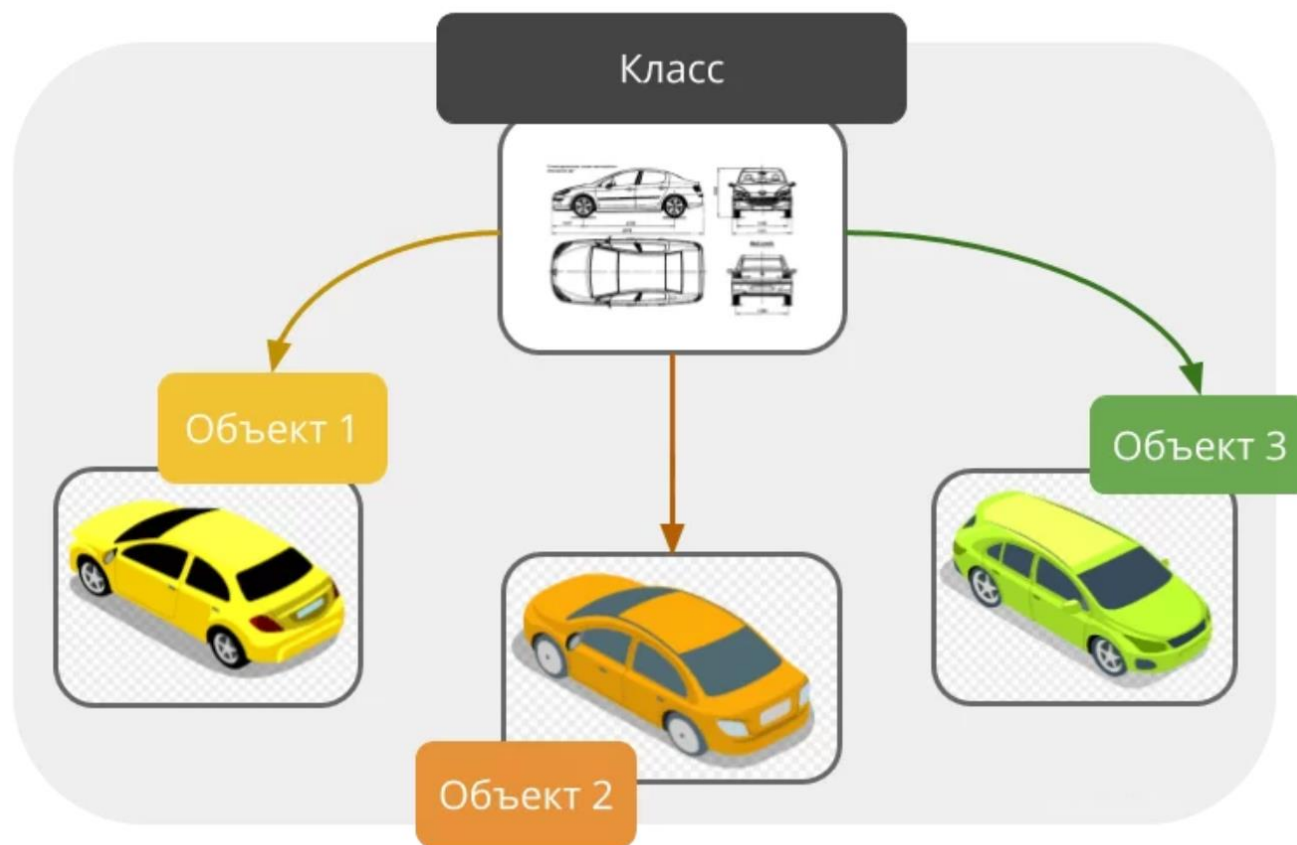
Объектно-ориентированное программирование (ООП) — методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования.

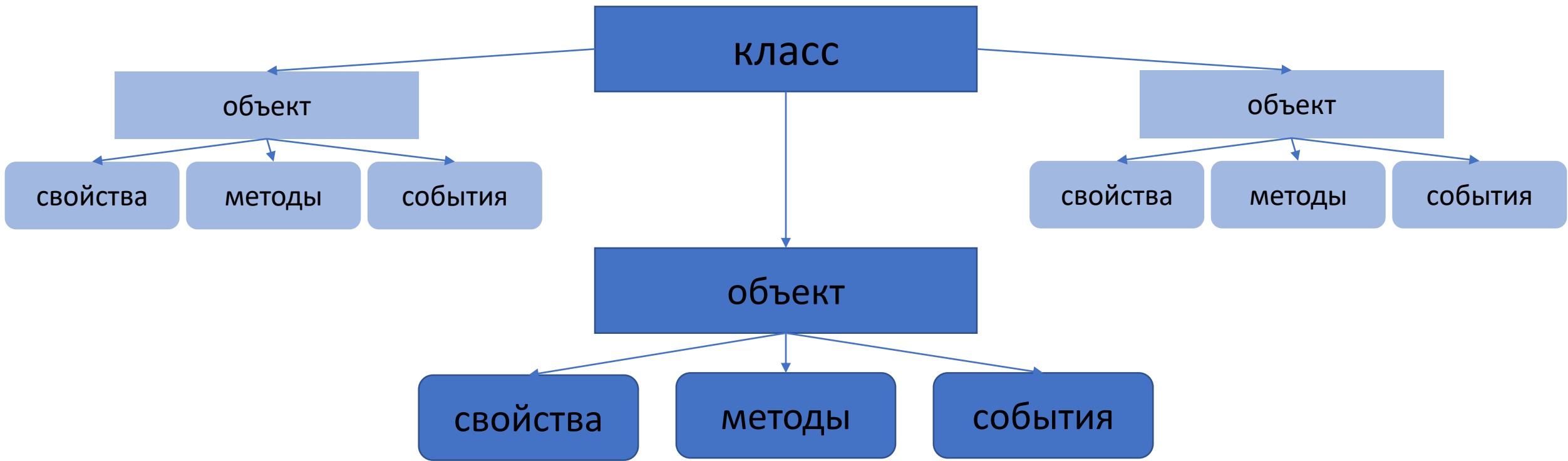
главное в программе - это данные

Класс – это способ описания сущности, определяющий состояние и поведение, зависящее от этого состояния, а также правила для взаимодействия с данной сущностью (контракт).

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

Объект (экземпляр) – это отдельный представитель класса, имеющий конкретное состояние и поведение, полностью определяемое классом.





- Абстракция
- Инкапсуляция
- Наследование
- Полиморфизм

Абстракция

Абстрагирование – это способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые.

Абстракция – это набор всех таких характеристик.



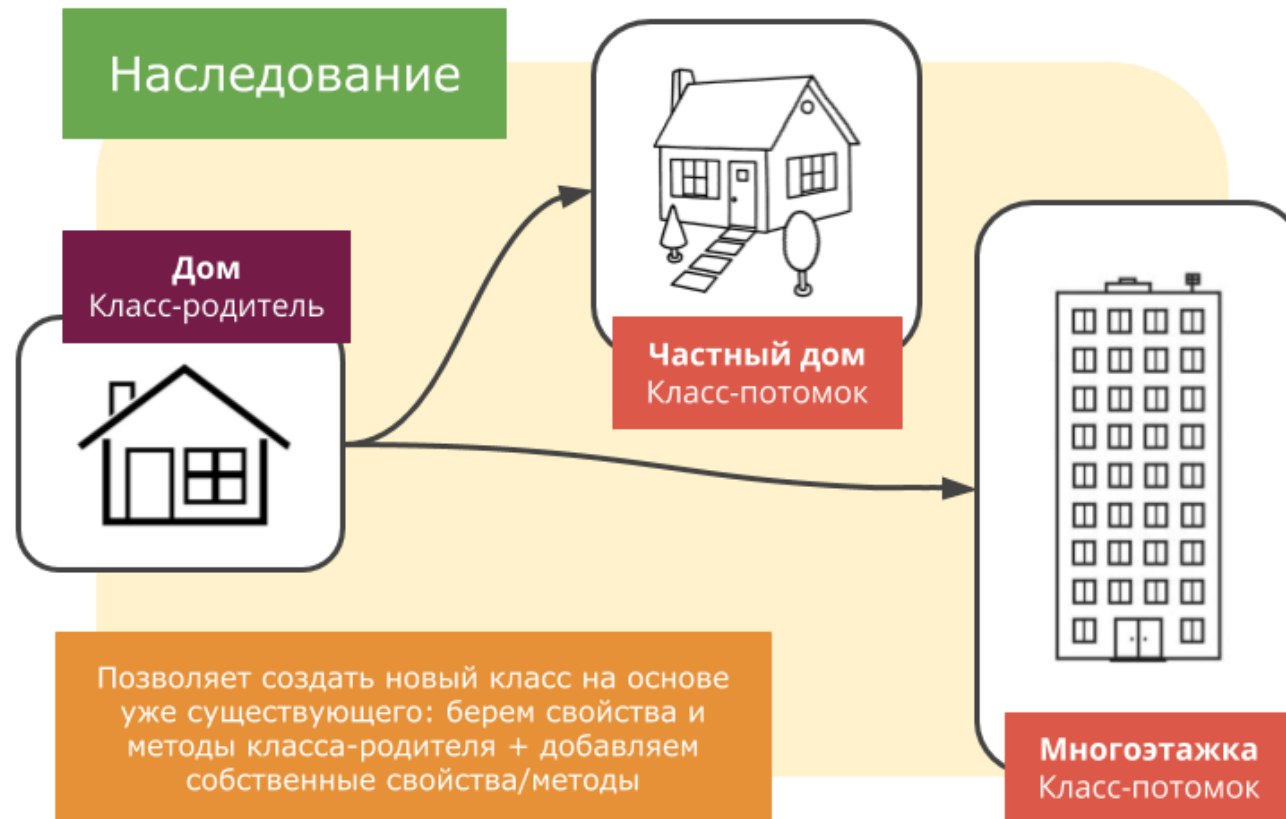
Инкапсуляция

Инкапсуляция – это свойство системы, позволяющее объединить данные и методы, работающие с ними, в классе и скрыть детали реализации от пользователя.



наследование

Наследование – это свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью. Класс, от которого производится наследование, называется базовым или родительским. Новый класс – потомком, наследником или производным классом.



Класс Дом

СВОЙСТВА

- 1) Тип фундамента
- 2) Материал крыши
- 3) Количество окон
- 4) Количество дверей

МЕТОДЫ

- 1) Построить
- 2) Отремонтировать
- 3) Заселить
- 4) Снести

•
•

Класс Частный дом

СВОЙСТВА

- 1) Тип фундамента (УНАСЛЕДОВАНО)
- 2) Материал крыши (УНАСЛЕДОВАНО)
- 3) Количество окон (УНАСЛЕДОВАНО)
- 4) Количество дверей (УНАСЛЕДОВАНО)
- 5) Количество комнат
- 6) Тип отопления
- 7) Наличие огорода

МЕТОДЫ

- 1) Построить (УНАСЛЕДОВАНО)
- 2) Отремонтировать (УНАСЛЕДОВАНО)
- 3) Заселить (УНАСЛЕДОВАНО)
- 4) Снести (УНАСЛЕДОВАНО)
- 5) Изменить фасад
- 6) Утеплить
- 7) Сделать пристройку

Класс Дом

СВОЙСТВА

- 1) Тип фундамента
- 2) Материал крыши
- 3) Количество окон
- 4) Количество дверей

МЕТОДЫ

- 1) Построить
- 2) Отремонтировать
- 3) Заселить
- 4) Снести

•
•

Класс Многоэтажный дом

СВОЙСТВА

- 1) Тип фундамента (УНАСЛЕДОВАНО)
- 2) Материал крыши (УНАСЛЕДОВАНО)
- 3) Количество окон (УНАСЛЕДОВАНО)
- 4) Количество дверей (УНАСЛЕДОВАНО)
- 5) Количество квартир
- 6) Количество подъездов
- 7) Наличие коммерческой недвижимости

МЕТОДЫ

- 1) Построить (УНАСЛЕДОВАНО)
- 2) Отремонтировать (УНАСЛЕДОВАНО)
- 3) Заселить (УНАСЛЕДОВАНО)
- 4) Снести (УНАСЛЕДОВАНО)
- 5) Выбрать управляющую компанию
- 6) Организовать собрание жильцов
- 7) Нанять дворника

Полиморфизм

Полиморфизм - это поддержка нескольких реализаций на основе общего интерфейса.

Абстрактный метод (он же виртуальный метод) - это метод класса, реализация для которого отсутствует

