

# **Достоинства и недостатки методов программирования**

Выполнил: Паршуков Никита

# Метод программирования

**Метод** в объектно-ориентированном программировании — это функция или процедура, принадлежащая какому-то классу или объекту . Различают простые методы и статические методы (методы класса):

- простые методы имеют доступ к данным объекта (конкретного экземпляра данного класса),
- статические методы не имеют доступа к данным объекта, и для их использования не нужно создавать экземпляры (данного класса).
- Методы предоставляют интерфейс, при помощи которого осуществляется доступ к данным объекта некоторого класса, тем самым, обеспечивая инкапсуляцию данных.
- В зависимости от того, какой уровень доступа предоставляет тот или иной метод, выделяют:

# Метод доступа

- В зависимости от того, какой уровень доступа предоставляет тот или иной метод, выделяют:
- Открытый (public) интерфейс – общий интерфейс для всех пользователей данного класса
- Защищенный (protected) интерфейс – внутренний интерфейс для всех пользователей данного класса;
- Закрытый (private) интерфейс, доступный только изнутри данного класса

# Достоинства

- Классы позволяют проводить конструирование из полезных компонент, обладающих простыми инструментами, что дает возможность абстрагироваться от деталей реализации.
- Данные и операции вместе образуют определенную сущность и они не «размазываются» по всей программе, как это нередко бывает в случае процедурного программирования.
- Локализация кода и данных улучшает наглядность и удобство сопровождения программного обеспечения.
- Инкапсуляция информации защищает наиболее критичные данные от несанкционированного доступа.

# Недостатки

- ООП порождает огромные иерархии классов, что приводит к тому, что функциональность расплзается или, как говорят, размывается по базовым и производным членам класса, и отследить логику работы того или иного метода становится сложно.
- В некоторых языках все данные являются объектами, в том числе и элементарные типы, а это не может не приводить к дополнительным расходам памяти и процессорного времени.
- Также, на скорости выполнения программ может неблагоприятно сказаться реализация полиморфизма, которая основана на механизмах позднего связывания вызова метода с конкретной его реализацией в одном из производных классов.
- Психологический аспект. Многие считают, что ООП это круто и начинают использовать его подходы всегда и везде и без разбору. Все это приводит к снижению производительности программ в частности и дискредитации ООП в целом. Ну, про плохих танцоров я уже сказал...