

Принцип открытости/закрытости

№ урока: 2 **Курс:** SOLID принципы в Java

Средства обучения: IntelliJ IDEA, Eclipse

Обзор, цель и назначение урока

Цель урока – Подробное рассмотрение принципа открытости-закрытости.

Изучив материал данного занятия, учащийся сможет:

- Использовать один из принципов SOLID.
- Понимать архитектуру приложения на языке программирования Java.
- Проектировать приложение согласно принципам SOLID.

Содержание урока

1. Как звучит принцип
2. Что такое точка расширения поведения
3. Что значит “закрыт для модификации”
4. Архитектура проекта
5. Принцип открытости-закрытости.
6. Рассмотрение примеров

Резюме

На данном уроке студенты познакомились с понятием SOLID.

Учащиеся узнали понятие принципа открытости-закрытости и его использования. Также познакомились с написанием кода согласно этому принципу.

Если применять его правильно, он делает код более расширяемым, логичным и читабельным.

Принцип открытости-закрытости — “A software artifact should be open for extension but closed for modification.”

“Программные сущности (классы, модули, функции и т. п.) должны быть открыты для расширения, но закрыты для изменения”

Закрепление материала

1. Как звучит принцип?
2. Что такое точка расширения поведения?
3. Что значит “закрыт для модификации”?
4. Наведите пример применения такого принципа

Самостоятельная деятельность учащегося

Задание 1

Выучить основные понятия, рассмотренные на уроке.

Задание 2

Создайте класс Human с полями firstName, lastName. В этом классе создайте метод getReadyForWork() в котором с помощью вывода на экран опишите последовательность того, как человек собирается на работу. Вы сможете перечислить много пунктов, но каждый раз что-то

будете вспоминать и дописывать в данном методе. Напишите решение такого рода нарушения принципа.

Рекомендуемые ресурсы

["Clean Architecture" Robert C. Martin](#)

<https://vimeo.com/43592685>