

# Принцип разделения интерфейса

**№ урока:** 4    **Курс:** SOLID принципы в Java

**Средства обучения:** IntelliJ IDEA, Eclipse

## Обзор, цель и назначение урока

Цель урока – Подробное рассмотрение принципа разделения интерфейса.

## 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. Предистория
2. Как звучит принцип
3. Пример нарушения
4. Последствия
5. Примеры

## Резюме

На данном уроке студенты познакомились с понятием SOLID.

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

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

**Принцип разделения интерфейса** — Клиенты не должны вынужденно зависеть от методов, которыми не пользуются.

При нарушении этого принципа клиент, использующий некоторый интерфейс со всеми его методами, зависит от методов, которыми не пользуется, и поэтому оказывается восприимчив к изменениям в этих методах. В итоге мы приходим к жесткой зависимости между различными частями интерфейса, которые могут быть не связаны при его реализации.

## Закрепление материала

1. Как звучит принцип ?
2. Как звучит каноническая формулировка принципа ?
3. Наведите пример применения такого принципа

## Самостоятельная деятельность учащегося

### Задание 1

Выучить основные понятия, рассмотренные на уроке.

### Задание 2

Создайте интерфейс Shape. В нем объявите 4 метода: drawCircle(), drawSquare(), drawTriangle(), drawRectangle(). Разделите данный интерфейс и реализуйте классы-наследники.

## Рекомендуемые ресурсы

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

<https://vimeo.com/43592685>