

A grayscale photograph of a person with dark hair wearing large headphones, looking intently at a laptop screen. Their hands are clasped near their chin in a thoughtful pose. The background is blurred, showing what appears to be a desk or office environment. A white rectangular border is superimposed over the center of the image, framing the text.

# INTERFACE SEGREGATION PRINCIPLE

Java Clean code

# Outline

## **Lesson 6.**

The Single Responsibility  
Principle

## **Lesson 7.**

The Open Closed Principle

## **Lesson 8.**

The Liskov Substitution Principle

## **Lesson 9.**

The Interface Segregation  
Principle

## **Lesson 10.**

The Dependency Inversion  
Principle

*Any fool can write code that  
a computer can understand.  
Good programmers write code that  
humans can understand.*

**Martin Fowler**



# Interface segregation principle

- *“Clients should not be forced to depend upon interfaces that they do not use.”*
- Code is easier to maintain and debug
- Interfaces that are responsible for a client specific needs







# Fat classes

- A lot of methods
- Tight Coupling
- Dependencies

# ISP Java Examples

- FocusListener
- KeyListener
- MouseMotionListener
- TestListener
- WindowFocustListener



\*KeyListenerImp.java

```
1 package com.kirilanastsov.icp.gui;
2
3 import java.awt.event.KeyEvent;
4 import java.awt.event.KeyListener;
5
6 public class KeyListenerImp implements KeyListener {
7     @Override
8     public void keyPressed(KeyEvent keyEvent) {
9         if (keyEvent.getKeyCode() == KeyEvent.VK_ENTER) {
10             hanleKeyCodePressed();
11         }
12     }
13
14     @Override
15     public void keyReleased(KeyEvent keyEvent) {
16         hanleKeyCodePressed();
17     }
18
19     @Override
20     public void keyTyped(KeyEvent keyEvent) {
21         hanleKeyTypedPressed();
22     }
23
24
25
26     public static void hanleKeyCodePressed() {
27     }
28
29
30     public static void hanleKeyReleasedPressed() {
31     }
32
33
34     public static void hanleKeyTypedPressed() {
35     }
36 }
37
38
```



A background image showing three people in a meeting. Two men and one woman are gathered around a table. One man is sitting on a stool, leaning over the table, while the other man and the woman stand behind him, looking at a laptop. The scene is dimly lit with a warm, sepia tone.

# Interface Segregation Principle Summary

- Clients should not be forced to implement unnecessary code
- Code is easier to maintain and debug
- Interfaces that are responsible for a client specific needs



# Course Progress

## Lesson 6

The Single  
Responsibility  
Principle

## Lesson 7

The Open  
Closed Principle

## Lesson 8

The Liskov  
Substitution  
Principle

## Lesson 9

The Interface  
Segregation  
Principle

## Lesson 10

The Dependency  
Inversion  
Principle

A sepia-toned photograph of a person clapping their hands. In the foreground, a wooden desk holds an open notebook with a smartphone resting on it. A laptop is partially visible in the background. A dark grey rectangular box with a thin white border is overlaid on the left side of the image, containing the text 'THANK YOU!' in a white serif font.

THANK YOU!