# Canvas Tech Task

Gorbunov Anton @tonyorb

# Contents

- Set Up manual

- Task Analysis

- Code Overview

# Set up manual

# Technical requirements

- Fresh Java Developer Kit. Download here

- Fresh maven. Download here

- Please set environment variables:
  - to JDK_HOME_DIR/bin directory
  - to MAVEN_HOME_DIR/bin directory

- To check if everything is ready run «mvn - version» command in your terminal. Result is following:

```
antongorbunov@MacBook-Pro IdeaProjects % mvn -version
Apache Maven 3.8.6 (84538c9988a25aec085021c365c560670ad80f63)
Maven home: /usr/local/Cellar/maven/3.8.6/libexec
Java version: 19.0.1, vendor: Homebrew, runtime: /usr/local/Cellar/openjdk/19.0.1/libexec/openjdk.jdk/
Contents/Home
Default locale: ru_RU, platform encoding: UTF-8
OS name: "mac os x", version: "12.2.1", arch: "x86_64", family: "mac"
```

# Run Tests

- Clone the repo using this <u>link</u>

- Open the terminal and go to project directory «canvasTest»

- Run the command «mvn test»

- If everything is compiled you can see test results as follow

```
[INFO] ------------------------------------------------------------
[INFO]  T E S T S
[INFO] ------------------------------------------------------------
[INFO] Running TitleManagerTests
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.028 s – in TitleManagerTests
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------
[INFO] Total time:  1.690 s
[INFO] Finished at: 2023-01-23T11:01:34+08:00
[INFO] ------------------------------------------------------------
antongorbunov@MacBook-Pro canvasTest %
```

# Task Analysis

# Expected result

- I use Java as a programming language with low level functions

- Tests are written using JUNIT 5 framework

- I have 1 function and 7 tests as a result

- I created a function, it receives article and return a title
  String createATitle (String article){
  // function body
  }

# Test design

- There are 7 tests that I believe cover all major scenarios:

  - Happy path scenario

  - Empty article

  - Article Without Spaces

  - Article With Number To Cut

  - Article With Special Chars To Cut

  - Article Nothing To Cut // less than 25 symbols

  - Article Without Words //only digits

# Code overview

# The function

```java
static String createATitle(String article){
    char[] array = article.toCharArray();
    if (array.length == 0) return "Article is empty";
    if (array.length < 25) return String.valueOf(array);
    boolean trim = false;
    for (int i = 22; i > 0; i--){
        if (!trim){
            if (array[i] == ' '){
                trim = true;
            }
            continue;
        }
        if ((array[i] >= 97 && array[i] <= 122) ||
(array[i] >= 65 && array[i] <= 90)){
            array[i + 1] = '.';
            array[i + 2] = '.';
            array[i + 3] = '.';
            return String.valueOf(array, 0 , i + 4);
        }
    }
    if (!trim) return "No spaces in article";
    return "Unable to create a title";
}
```

# Preconditions

- In Java String is array of characters
  char[] array = article.toCharArray();

- First I should check if article is empty string or it has less than 25 chars in it

```java
if (array.length == 0) return "Article is empty";
if (array.length < 25) return String.valueOf(array);
```

# Algorytm

- I use marker «trim» to check if char is « » - it indicates that there is a space between words, by default it false
  boolean trim = false;

- Then I have a loop starts with 22 and decrease by 1 till 0
  The max length is 25 and we should add 3 dots … There is why 22
  for (int i = 22; i > 0; i--){}

- I check each char till I find « ». Then I set trim = true
  if (!trim){

```
    if (array[i] == ' '){
        trim = true;
    }
    continue;
}
```

# Algorytm

- When « » is found I continue to check previous chars.
  If it is letter I assume it is a word and cut, if not I continue till I find a letter.
  Numbers here are related to unicode table

```
if ((array[i] >= 97 && array[i] <= 122) || (array[i] >= 65 && array[i] <= 90)){
    array[i + 1] = '.';
    array[i + 2] = '.';
    array[i + 3] = '.';
    return String.valueOf(array, 0 , i + 4);
}
```

# Algorytm

- If I don't find any letter I assume that String contains only digits or spec chars
  return "Unable to create a title»;

- If there is no « » in article
  if (!trim) return "No spaces in article";

# Test description

- Each of 7 test has article and expected title

- I compare result of function call and expected result using JUnit framework

```java
  @Test
void userCaseArticle(){
    String article = "Volvo released a new car with the
following spec: V6 236HP. It will cost $22647 and going to
be sold in New York only";
    String expectedTitle = "Volvo released a new...";

    assertEquals(expectedTitle,
TitleManager.createATitle(article));
}
```

# Thank You