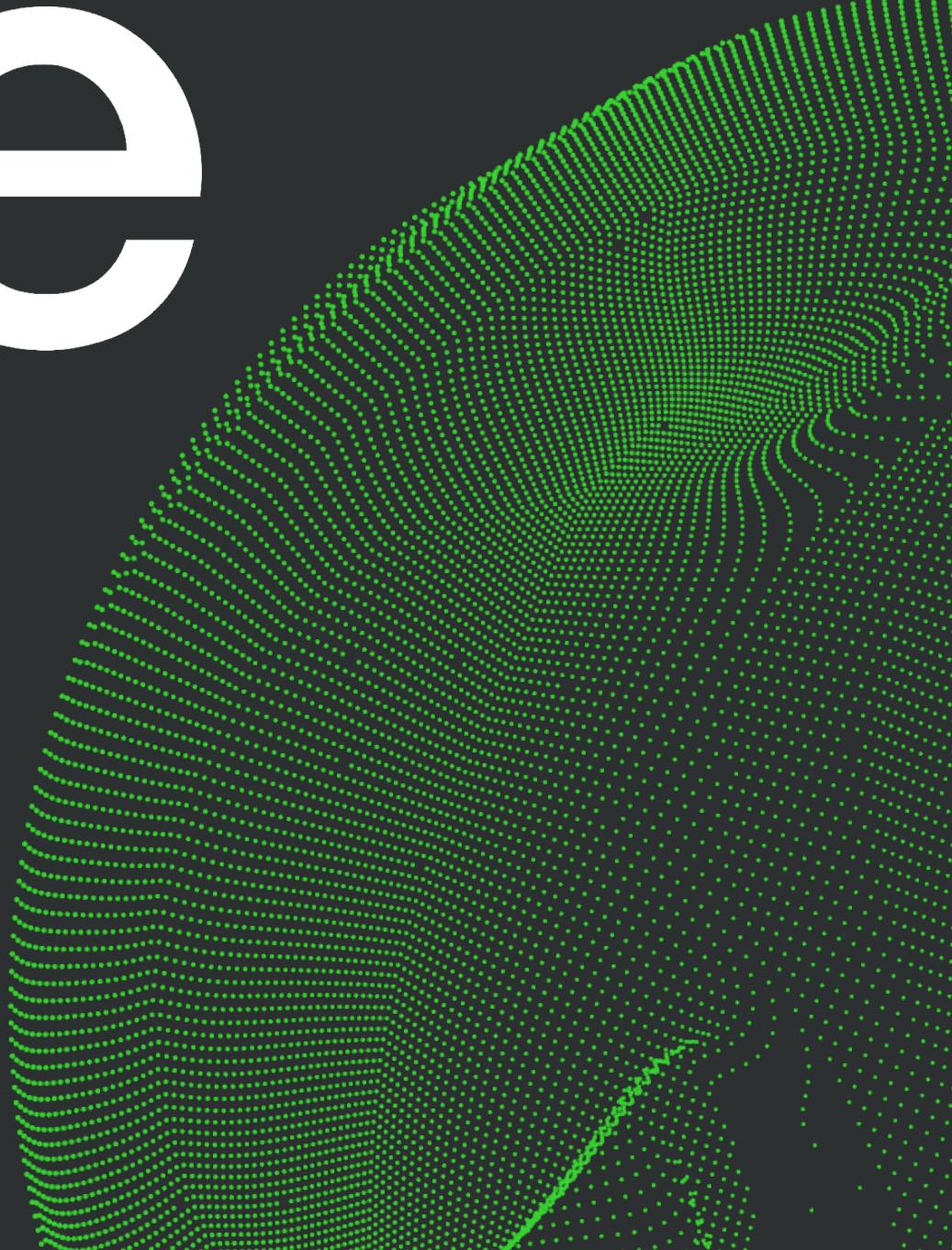


# merge

Точка слияния профессиональных  
IT-конференций

## АВТОМАТИЗАЦИЯ ОБУЧЕНИЯ С ИСПОЛЬЗОВАНИЕМ GITHUB



# РОМАНОВ АЛЕКСЕЙ

Software Архитектор, ИННОТЕХ

- 11+ лет коммерческой разработки (Java/Kotlin)
- 6+ лет в роли TeamLead / Архитектора
- 7 лет преподаю МГТУ им. Н.Э.Баумана курс «Распределенные системы обработки информации»



# ЗАЧЕМ НУЖНА АВТОМАТИЗАЦИЯ ПРИЕМА ЗАДАНИЙ?



# ПРИЕМ ЛАБОРАТОРНЫХ

- Проверка корректности выполнения задания
- Обсуждение реализации и feedback

...

© 2022 Иннополис. Конфиденциальная информация. Все права защищены.  
Все изображения являются собственностью Иннополиса.



# **Название**

## **Формулировка**

Краткое описание проблематики и что нужно сделать.

## **Требования**

Подробно по пунктам, что нужно сделать, какие технологии использовать и какие есть ограничения.

## **Пояснения**

В этом пункте стараемся подробно описать нюансы реализации, как и для чего использовать конкретные технологии (если применимо).

Это самый важный пункт, т.к. в нем мы стараемся предусмотреть все возникающие в процессе реализации у студента вопросы и дать на них ответы или ссылки на материалы.

## **Прием задания**

Описание шагов, как запустить и проверить корректность выполнения задания. Например, если у нас Java приложение, то `./gradlew clean test`.

## **Литература**

Ссылки на статьи, которые помогут выполнить лабораторную работу.

# АВТОМАТИЗАЦИЯ ЧЕРЕЗ GITHUB ACTIONS

```
# ./github/workflows/build.yml
name: Build
on: [push]
jobs:
  build:
    name: Build
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: actions/setup-java@v1
        with:
          java-version: 11
      - uses: eskatos/gradle-command-action@v1
        with:
          arguments: clean build
```

# FORK + PULL REQUEST

## Плюсы:

- Простое и очевидное решение
- В Pull Request можно писать feedback

## Минусы:

- Pull Request виден всем студентам, а значит сложно сдержаться от просмотра чужих решений
- в Pull Request не доступны ваши Secrets, а значит все токены и т.п. нужно прописывать в явном виде

# GITHUB CLASSROOM

The screenshot shows the GitHub Classroom interface for a classroom named "demo-classroom". The top navigation bar includes the GitHub Classroom logo, "GitHub Education", and a user profile icon. The main header displays the classroom name "demo-classroom" and the owner "Romanow-Education". Below the header, there are tabs for "Assignments" (2), "Students" (1), "TAs and Admins" (1), and "Settings". The "Assignments" tab is selected, showing two assignments: "homework1" and "homework2", both listed as "Individual assignment". Each assignment has an "Invite link" dropdown, a pencil icon for editing, and a trash can icon for deletion. A green "New assignment" button is located in the top right corner of the assignments section. A blue notification bar at the bottom encourages users to complete a survey.

Classrooms / demo-classroom

## demo-classroom

Romanow-Education

Assignments 2 Students 1 TAs and Admins 1 Settings

### Assignments

New assignment

We value your feedback! Please take a few minutes to complete our [survey](#). X

---

 **homework1**  
Individual assignment

 **homework2**  
Individual assignment

---

# GITHUB CLASSROOM

- Dashboard с информацией о сданных работах
- Можно использовать приватные репозитории, студенты не будут видеть реализации друг друга
- Есть интеграции с различными LMS (Moodle, Google Classroom и т.п.) \*
- Автоматический прогон тестов

# СОЗДАЕМ ОРГАНИЗАЦИЮ НА GITHUB

The screenshot shows the GitHub organization page for 'Romanow-Education'. The page includes a search bar, navigation links for Pull requests, Issues, Marketplace, and Explore, and a user profile icon. The organization's logo is a graduation cap icon. The 'Overview' tab is selected, showing 'Popular repositories' and 'Repositories' sections. Popular repositories include 'homework1-template' (Public template, Kotlin), 'homework1-aromanow' (Public, created by GitHub Classroom, Kotlin), 'homework2-template' (Public template, Kotlin), and 'homework2-aromanow' (Public, created by GitHub Classroom, Kotlin). The 'Repositories' section features a search bar and filters for Type, Language, Sort, and a 'New' button. A sidebar on the right shows 'View as: Public', 'People' (with an 'Invite someone' button), and 'Top languages' (Kotlin). The footer contains a copyright notice for Innotech.

Search or jump to... / Pull requests Issues Marketplace Explore

Romanow-Education

Follow

Overview Repositories 4 Projects Packages Teams People 1 Settings

Popular repositories

homework1-template Public template

Homework template

Kotlin

homework1-aromanow Public

homework1-aromanow created by GitHub Classroom

Kotlin

homework2-template Public template

Kotlin

homework2-aromanow Public

homework2-aromanow created by GitHub Classroom

Kotlin

Repositories

Find a repository... Type Language Sort New

homework2-template Public template

Kotlin ⭐ 0 0 0 0 Updated 23 hours ago

homework2-aromanow Public

View as: Public

You are viewing this page as a public user.

You can create a README file or pin repositories visible to anyone.

People

Invite someone

Top languages

Kotlin

© 2022 Innotech. Конфиденциальная информация. Все права защищены.  
Все товарные знаки являются собственностью их владельцев.

# СОЗДАЕМ GITHUB CLASSROOM

The screenshot shows the GitHub Classroom interface for a classroom named "demo-classroom". The top navigation bar includes the GitHub Classroom logo, the text "GitHub Education", and a user profile icon. Below the header, the URL "Classrooms / demo-classroom" is visible. The main title "demo-classroom" is displayed above the owner "Romanow-Education". The navigation bar at the top of the page has four items: "Assignments" (0), "Students" (0), "TAs and Admins" (1), and "Settings". The "Assignments" item is highlighted with a red underline. The main content area is titled "Assignments". It features two main sections: one for creating assignments and one for teaching Git & GitHub fundamentals.

**Assignments**



**Create an assignment to get started.**

Create an individual assignment to generate an assignment repository for each student to work from. Or, create a group assignment and have students work collaboratively in groups from team repositories.

[Create an assignment](#)

Learn more about [individual](#) and [group assignments](#).



**Need to teach Git & GitHub fundamentals?**

The Classroom team has created an assignment for you to use to teach your students the fundamentals of Git & GitHub.

[Use starter assignment](#)

[Learn more](#)

# СОЗДАЕМ TEMPLATE REPOSITORY

The screenshot shows a GitHub repository page for 'Romanow-Education / homework1-template'. The repository is public and has one commit from 'Romanov Alex' with the message 'init commit'. There are no contributors. The README.md file contains the following content:

```
Лабораторная работа #1

Калькулятор

Формулировка

Требуется проверить корректность арифметического выражения и вычислить его результат. На вход через аргументы командной строки поступает выражение, содержащие:
• скобки (, );
• сложение +, вычитание -, умножение *, деление /;
```

# СОЗДАЕМ ASSIGNMENT



# СОЗДАЕМ ASSIGNMENT

The screenshot shows the 'New assignment' creation interface in GitHub Classroom. At the top, there's a navigation bar with the GitHub Classroom logo, 'GitHub Education', and a user profile icon. Below the navigation, the path 'Classrooms / demo-classroom / New assignment' is visible. On the left, a sidebar lists three steps: 'Assignment basics' (marked with a checkmark), 'Starter code and environment' (which is selected and highlighted in grey), and 'Grading and feedback'. The main content area has a heading 'Add your starter code and choose an optional online IDE.' It contains two sections: 'Add a template repository to give students starter code' (with a note about empty repositories if no template is added) and 'Add a supported editor' (with a note about automatically linking editors). Both sections have dropdown menus: 'Romanow-Education/homework1-template' for the template and 'Select an editor' for the editor. At the bottom, there are 'Back', 'Cancel', and 'Continue' buttons.

Classroom

GitHub Education

Classrooms / demo-classroom / New assignment

✓ Assignment basics

● Starter code and environment

Grading and feedback

Add your starter code and choose an optional online IDE.

Add a template repository to give students starter code

Your assignment will be created with empty student repositories if you don't add starter code. Changes to starter code after students have accepted the assignment will not retroactively change existing student repositories.

ⓘ Note: All starter code must use a [template repository](#). Your starter code repository must be either in the same organization as this classroom or a public repository if elsewhere. Learn about [transferring your repositories](#).

Romanow-Education/homework1-template

Add a supported editor

Automatically include a link to an editor in students' repositories to give them a one-click experience for getting started coding, running, and collaborating on their code.

Select an editor

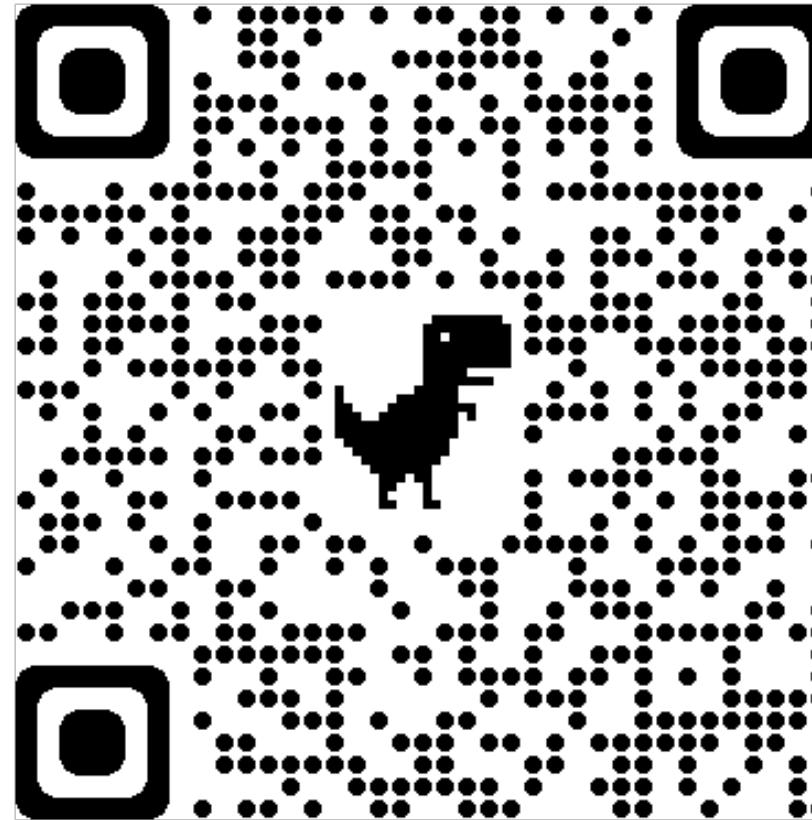
← Back

Cancel

Continue

<https://docs.github.com/en/rest/reference/repos#transfer-a-repository>

# ВЫДАЕМ ЗАДАНИЕ СТУДЕНТАМ



<https://classroom.github.com/a/D9u-zBOw>

# CLASSROOM DASHBOARD

The screenshot shows the GitHub Classroom interface for an assignment named 'homework1'. The top navigation bar includes the Classroom logo, 'GitHub Education', and a user profile icon. The URL 'Classrooms / demo-classroom / homework1' is visible in the address bar.

The main content area displays the assignment details:

- Assignment type: Individual assignment (Active)
- Links: A code link (<https://classroom.github.com>), an 'Edit' button, and a 'Download' button.

On the left, a sidebar provides student statistics:

Rostered students	1
Added students	0
Accepted students	1
Assignment submissions	0
Passing students	0/1

On the right, the 'Classroom roster' table lists one student:

Classroom roster		Unlinked accounts	Accepted	Submitted	Passing	Sort
 Романов Алексей @aromanow	Latest commit failed	10/10	-○-	1 commit		

# НАСТРАИВАЕМ AUTO GRADING

```
// .github/classroom/autograding.json
{
  "tests": [
    {
      "name": "Run tests",
      "setup": "",
      "run": "./gradlew clean test",
      "input": "",
      "output": "",
      "comparison": "included",
      "timeout": 10,
      "points": 10
    }
  ]
}
```

# НАСТРАИВАЕМ AUTO GRADING

```
# ./github/workflows/classroom.yml
name: GitHub Classroom Workflow
on: [push]
jobs:
  build:
    name: Autograding
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: actions/setup-java@v1
        with:
          java-version: 11
      - uses: eskatos/gradle-command-action@v1
        with:
          arguments: clean build
      - uses: education/autograding@v1
```

# УСПЕШНОЕ ВЫПОЛНЕНИЕ ТЕСТОВ

The screenshot shows the Google Classroom interface for a course named 'demo-classroom' with an assignment titled 'homework1'. The assignment is set as an individual assignment and is active. The student 'Романов Алексей' (@aromanow) has submitted a commit, which is highlighted with a green checkmark and the message 'Latest commit passed'. The assignment has a total score of 10/10 and 1 commit. The student's GitHub profile picture is shown next to their name.

Classroom

Classrooms / demo-classroom / homework1

## homework1

Individual assignment • Active

Rostered students: 1  
Added students: 0  
Accepted students: 1  
Assignment submissions: 0  
Passing students: 1/1

Search by GitHub username or student identifier

Classroom roster

Unlinked accounts ▾ Accepted ▾ Submitted ▾ Passing ▾ Sort ▾

Романов Алексей  
@aromanow

Latest commit passed

10/10 1 commit

https://classroom.github.io Edit Download

# Виды тестов

- Если язык и framework заранее известны, то unit-тесты + интеграционные
- Если этих тестов недостаточно или может быть любой язык разработки, то можно использовать Postman / Newman

# ПРОБЛЕМЫ ПРИ ИСПОЛЬЗОВАНИИ GITHUB CLASSROOM

2000 мин/месяц на **приватные**  
репозитории, а дальше...

# Choose the plan that's right for you.

How often do you want to pay?

Monthly

Yearly  Get 1 month free

## Free

The basics for individuals and organizations

\$ **0** per user/month

[Create a free organization](#)

## MOST POPULAR

### Team

Advanced collaboration for individuals and organizations

\$ **4** per user/month

[Continue with Team ▾](#)

## Enterprise

Security, compliance, and flexible deployment

\$ **21** per user/month

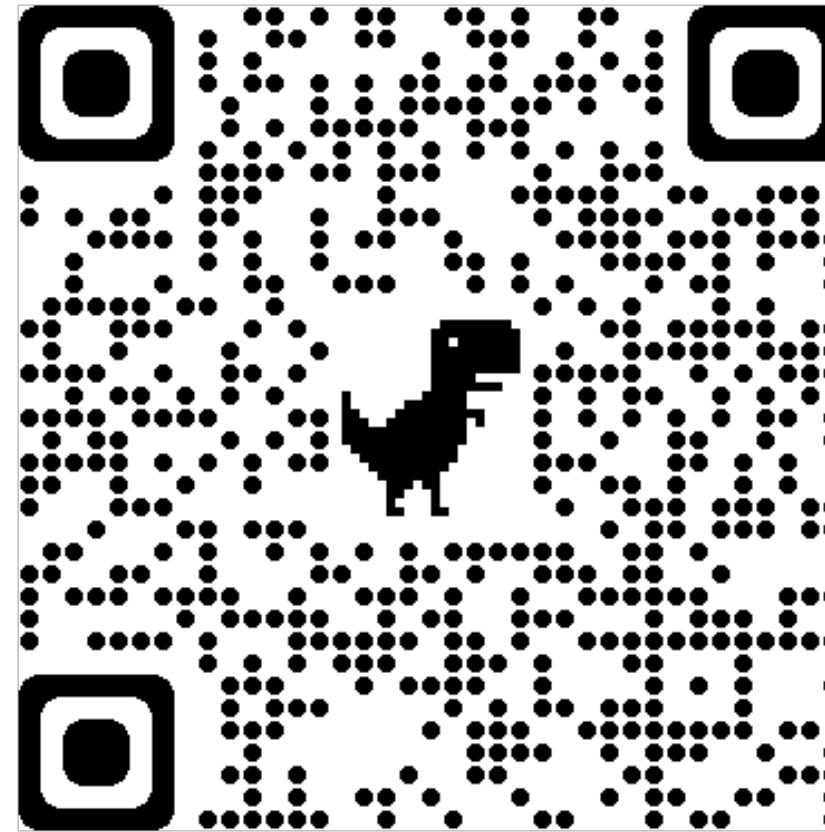
[Start a free trial](#)

[Contact Sales](#)

# ВАРИАНТЫ РЕШЕНИЯ

- Использовать публичные репозитории
- Установить прогон тестов только на ветку master
- Self-Hosted Runner

# ОБРАЗЫ ДЛЯ SELF-HOSTED RUNNER



<https://github.com/actions/virtual-environments/tree/main/images/linux>

# УЧЕТ ЛАБОРАТОРНЫХ

[2022] Успеваемость ☆ ! o

Файл Правка Вид Вставка Формат Данные Инструменты Расширения Справка Последнее изменение: толь...

Настройки Доступа

Настройки Доступа

31

31

Сумма сдавших

3 3 2

Лист1

#	Фамилия Имя	GitHub	1 л/р	2 л/р	3 л/р
1	Шипилев Алексей	shipilev	+	+	+
2	Садогурский Барух	jbaruch	+	+	+
3	Борисов Евгений	Jeka1978	+	+	
4	Романов Алексей	aromanow			
Сумма сдавших			3	3	2

# УЧЕТ ЛАБОРАТОРНЫХ

```
# ./github/workflows/classroom.yml
name: GitHub Classroom Workflow
on: [push]
jobs:
  build:
    name: Autograding
    runs-on: ubuntu-latest
    steps:
      . . .
      - uses: education/autograding@v1
      - name: Github auto grader mark
        uses: Romanow/google-sheet-autograder-marker@v1.0
        with:
          google_token: ${{secrets.GOOGLE_API_KEY}}
          sheet_id:"1XbYbAtG...QI4Y"
          homework_number: 1
```

# УЧЕТ ЛАБОРАТОРНЫХ

← Сего́дня, 01:39

Всего: 1 изменение

1	#	Фамилия Имя	GitHub	1 л/р	2 л/р	3 л/р	G	H	I
2	1	Шипилев Алексей	shipilev	+	+	+			
3	2	Садогурский Барух	jbaruch	+	+	+			
4	3	Борисов Евгений	Jeka1978	+	+				
5	4	Романов Алексей	aromanow	+	+				
6		Сумма сдавших		4	4	2			
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									
26									



Лист1 ▾

История версий

Все версии

СЕГОДНЯ

5 мая, 01:39

Текущая версия

github-autograder-account@github-  
classroom-autograder.iam.gserviceaccount.com  
Алексей Романов

ВЧЕРА

▶ 4 мая, 20:38

github-autograder-account@github-  
classroom-autograder.iam.gserviceaccount.com  
Алексей Романов

4 мая, 19:02

Алексей Романов

4 мая, 17:24

Алексей Романов

4 мая, 01:57

Алексей Романов

ПОНЕДЕЛЬНИК

▶ 2 мая, 23:55

Показать изменения

# ПРОБЛЕМЫ СПИСЫВАНИЯ

brain thinking of



ways to  
cheat on homework

brain actually



doing the homework

# КАК ОПРЕДЕЛИТЬ, ЧТО РАБОТА СПИСАНА?

В своей практике я чаще всего определял что работа списана, если:

- во время обсуждения студент терялся в навигации по коду;
- не мог объяснить что делает конкретный метод;
- я замечал какие-то странные и бросающиеся в глаза конструкции, которые я уже видел раньше.

# КАК ОПРЕДЕЛИТЬ, ЧТО РАБОТА СПИСАНА?

```
@RestController
@RequestMapping("/api/v1/eval")
class CalculationController(
    private val calculationService: CalculationService
) {

    @PostMapping(
        consumes = [MediaType.TEXT_PLAIN_VALUE],
        produces = [MediaType.TEXT_PLAIN_VALUE]
    )
    fun eval(@RequestBody expression: String): String {
        println("Vychislyuem vyrazhenie: $expression") // <-- например, такое
        return calculationService.eval(expression).toString()
    }
}
```

# КАК ОПРЕДЕЛИТЬ, ЧТО РАБОТА СПИСАНА?

- Для каждого языка и фреймворка описать типовые файлы конфигурации, которые стоит исключить из рассмотрения
- В коде выделить методы, и выполнять сравнение реализаций конкретных методов нечетким сравнением, т.е. без учета названия *переменных и пробельных символов*
- Выводить предупреждение, если степень схожести больше 85%.  
*Потому что схожесть реализации 2-3 методов не означает, что студенты списали друг у друга*

# ВЫВОДЫ

- Можно автоматизировать проверку с помощью GitHub Classroom
- Есть большой выбор GitHub Actions, который позволяют написать тесты практически для любых сценариев
- Если требуется какая-то особая интеграция, это можно сделать самостоятельно
- Если кончились бесплатные минуты для приватных репозиториев, то можно подключить Self-Hosted Runner