

Тестирование (Q&A)

Практическая работа

Задача 1

Что нужно сделать

Протестируйте вручную свой код:

1. Возьмите выполненную вами задачу на языке Java по разработке системы расчёта штрафов в Германии (задача 1 из модуля «Разработка на Java»). Вам нужен код, сохранённый в **OnlineGDB** (во-первых, это удобно, а, во-вторых, в следующей задаче он понадобится вам именно в таком виде).
2. Составьте проверочную таблицу из двух столбцов: **carSpeed** и **result**. В столбец **carSpeed** впишите различные значения скорости, а в столбец **result** — ожидаемые результаты, которые должна выдать программа при подстановке в неё соответствующих значений.

Таблица должна выглядеть примерно так же, как показано в блоке «Как проверить результат» модуля «Разработка на Java» (таблица 1), но содержать значения скоростей и штрафов в Германии.

Вручную проверьте работу присланных вам кодов подстановкой в код различных значений переменной **carSpeed**.

3. Если вы нашли ошибку в коде, исправьте её.

Задача 2

Что нужно сделать

Напишите на Java код. Он должен тестировать написанную вами на Java систему расчёта штрафов в Германии, для которой вы составили таблицу в предыдущей задаче:

1. Взгляните [на код, который был написан в видео](#). Вам необходимо написать аналогичный код.

2. Откройте написанный вами код на Java в редакторе **OnlineGDB** (мы выбрали его вместо IdeOne, поскольку IdeOne перестал работать) и нажмите кнопку **Fork this**.
3. В открывшемся редакторе создайте новый метод **calculateFine** и скопируйте в него код, который до этого был в методе **main**:

```
public static int calculateFine(int carSpeed)

{

    //Код копировать сюда

}
```

Замените **System.out.println** на **return**, как показано в примере выше и в видеоматериале.

4. Создайте метод **check**, [как в примере](#), для своего кода.
5. В методе **main** вызовите метод **check** со всеми значениями переменной, которые вы записали в таблице.
6. Запустите код и посмотрите на результат его работы.

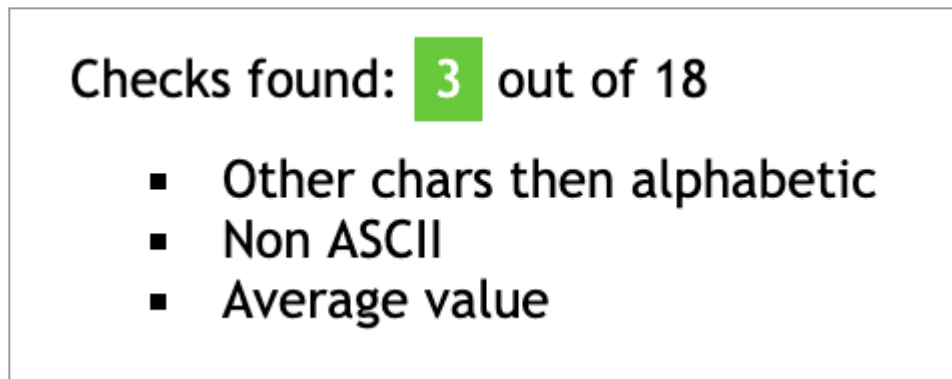
Задача 3

Что нужно сделать

Попробуйте себя в роли тестировщика веб-формы:

1. Откройте [страницу с набором задач для тестирования](#). По умолчанию слева будет выбрана первая задача (**challenge #1**).
2. В рамках задачи **challenge #1** существует форма из четырёх полей, три из которых уже заполнены и неизменяемы.
3. В четвёртое поле **First Name** потенциальные пользователи могут вводить разные необычные значения, которые необходимо отдельно отлавливать на бэкенде и выдавать сообщение об ошибке.

4. Существует 18 вариантов значений, которые можно ввести в эту форму. Обычным вариантом будет верное значение — какое-либо имя на латинице, например Alex. Необычные варианты — это, например, цифры или имя на кириллице. Если после введения каждого из типов вариантов нажимать кнопку **Submit**, то в конце вы получите такое сообщение:



5. Ваша задача — вводить в форму разные необычные значения, чтобы количество проверенных вариантов увеличивалось вплоть до 18. Продолжайте это делать до тех пор, пока не станет совсем сложно.

Как проверить результат

Вы можете сделать это самостоятельно. Главное, чтобы вы видели, что написанный вами код работает верно, и что при проверке формы вам удалось найти хотя бы несколько новых вариантов значения поля (три варианта показаны выше).

Предлагаем для проверки задач ориентироваться на самих себя: довольны ли вы полученным результатом, получилось ли у вас проверить все варианты значений, которые могут быть в коде или в поле формы?