

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

ОТЧЁТ ПО ЛАБОРАТОРНЫМ РАБОТАМ №3
курса «Моделирование распределительных систем»

Выполнил студент:

Амельченко Николай Андреевич
группа: ИА-831

Проверил:

преподаватель
Ахпашев Руслан Владимирович

Новосибирск, 2020 г.

Содержание

1. Создание бота для Honor Cup на Python	2
1.1. Задание	2
1.2. Способ решения	2
1.3. Функции со скриншотами	2
1.3.1. Листинг программы	7
1.3.2. Результаты	10
1.4. Вывод	11

Лабораторная работа №2 1

Создание бота для Honor Cup на Python

1.1. Задание

Создать бота, способного самостоятельно выигрывать викторину на сайте Honor Cup методом проб и ошибок и обучаться при этом максимально эффективно. Использовать язык Python и библиотеку selenium, которая дает пользователю горизонт для взаимодействия кода с функционалом браузера.

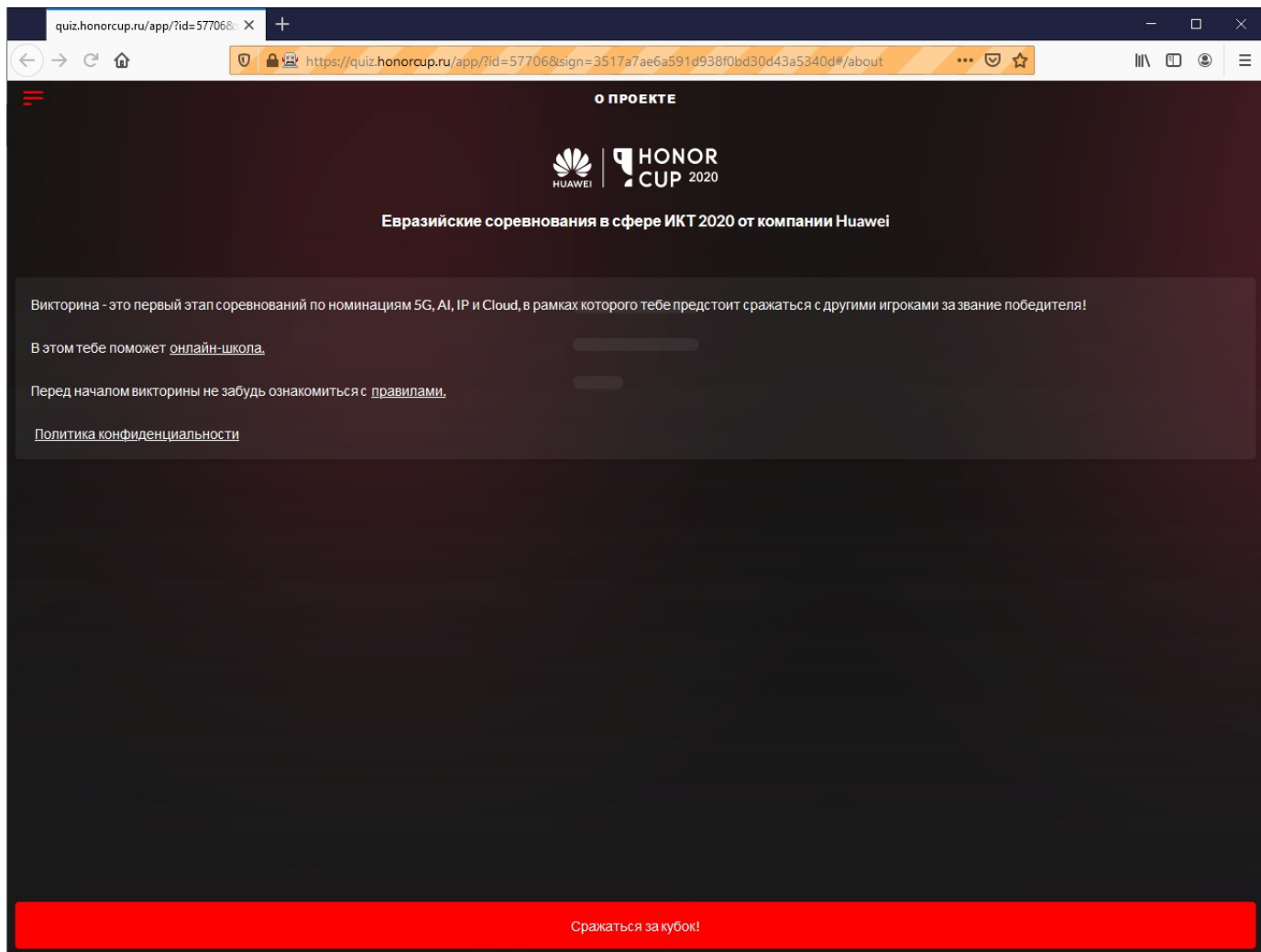
1.2. Способ решения

Мой бот, авторизуясь на сайте начинает викторину, считывает вопрос и ищет на него ответ в базе данных сформированной методом проб и ошибок. Найдя ответ он кликает на него, в противном случае он пробует нажать на ответ, который еще не пробовал нажимать для текущего вопроса, при верном ответе база данных дополняется, при неверном сужается круг потенциально правильных ответов для данного вопроса

1.3. Функции со скриншотами

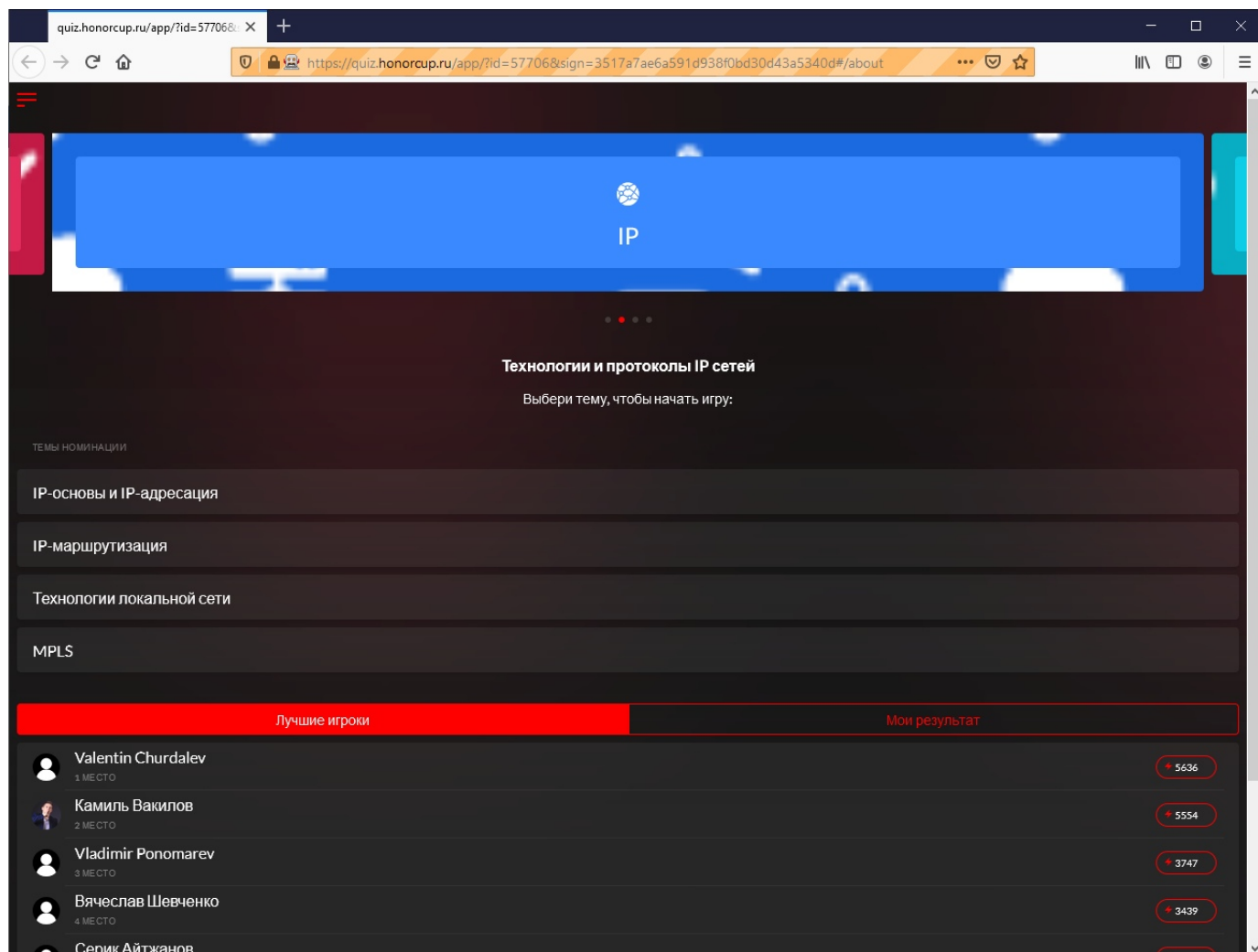
Открываем браузер и авторизуемся

```
1 TOKEN = 'https://quiz.honorcup.ru/app/?id=57706&sign=3517  
    a7ae6a591d938f0bd30d43a5340d'  
2 #open a browser (Firefox)  
3 browser = webdriver.Firefox(executable_path='geckodriver.exe')  
4 browser.get(TOKEN) авторизуемся#
```



Кликаем по кнопке "Сразаться за кубок!":

```
1 time.sleep(2) ручная# задержка
2 # #click battle_button
3 battle_button = browser.find_element_by_class_name('about__buttons')
   создаем# переменную содержащую адрес кнопки 'сразаться за кубок' для
   дальнейшего взаимодействия с ней
4 battle_button.click() кликаем# на полученную кнопку
```

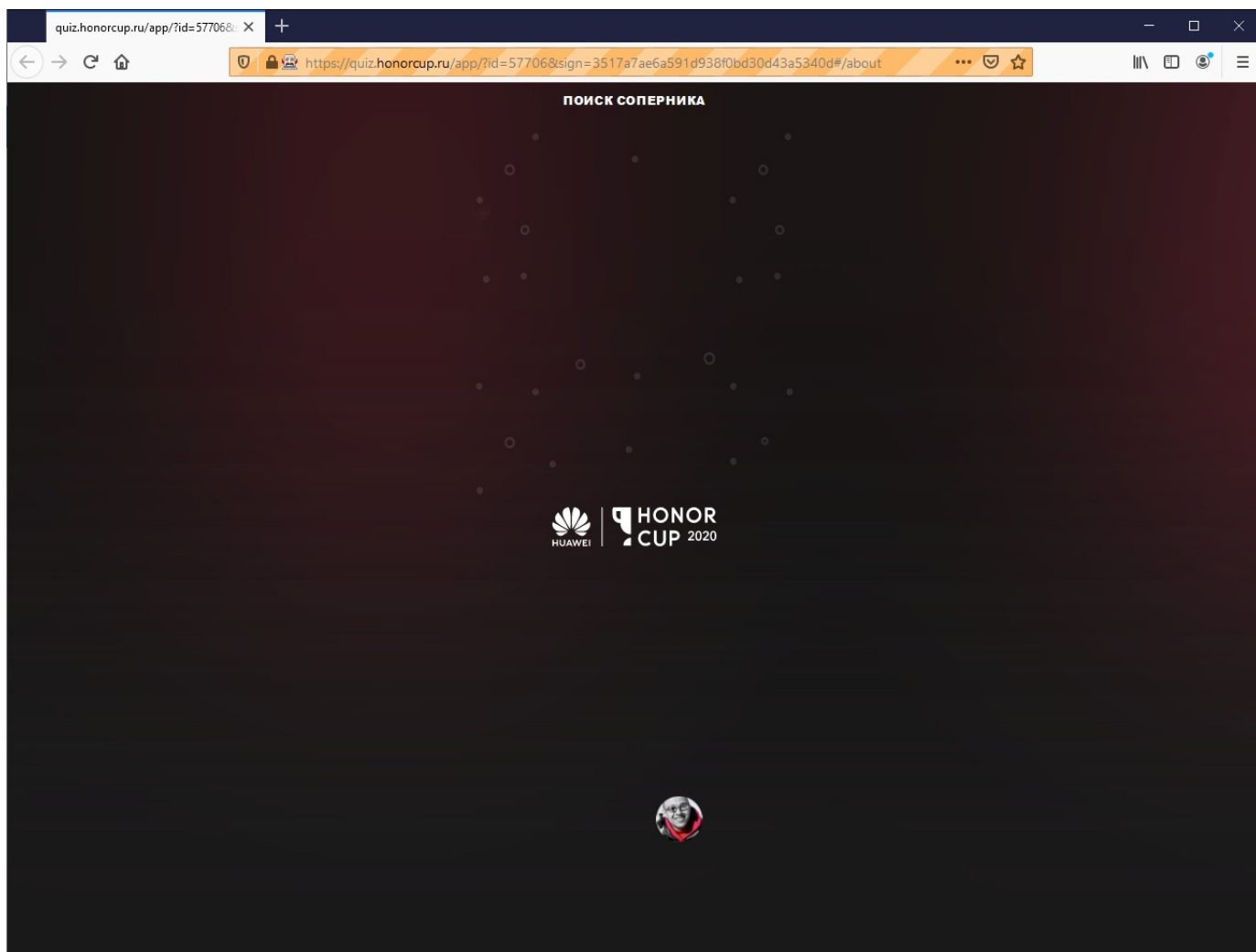


Попадаем на страницу выбора раздела, выбираем тему 5G и начинаем викторину:

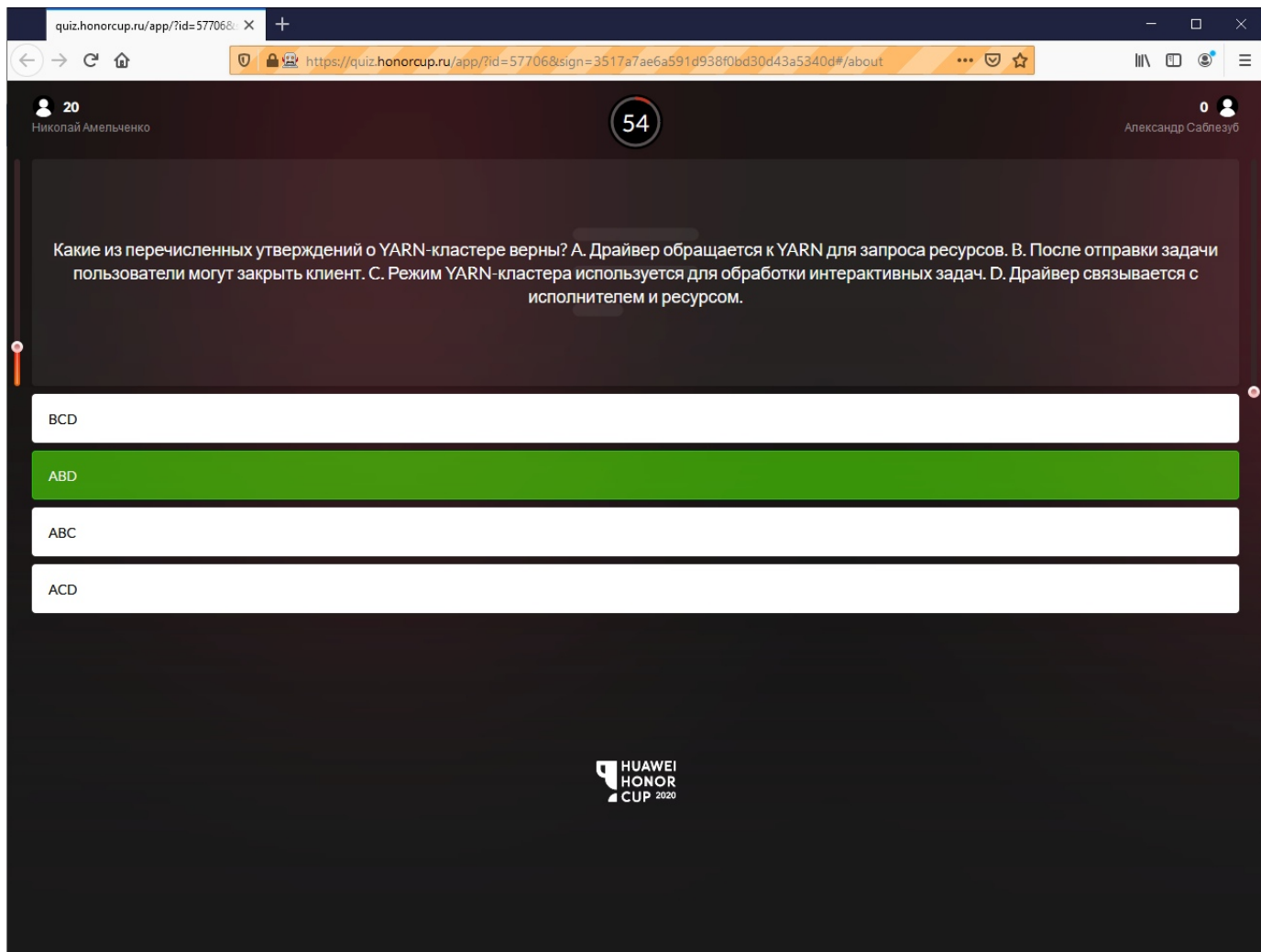
```

1 # #choose a category and theme
2 category = browser.find_elements_by_class_name('slider__item') адрес#
   раздела на сайте
3 category[3].click()
4
5 theme = browser.find_elements_by_class_name('profile__theme') адрес#
   темы викторины на сайте
6 theme[1].click()
7
8 categories_play_button = browser.find_element_by_xpath('/html/body/app
   /div[1]/nomination/div/div/div[2]/div[3]/div[2]/div/div/div[2]/div '
   )#('button-group-2x')
9 categories_play_button.click()

```

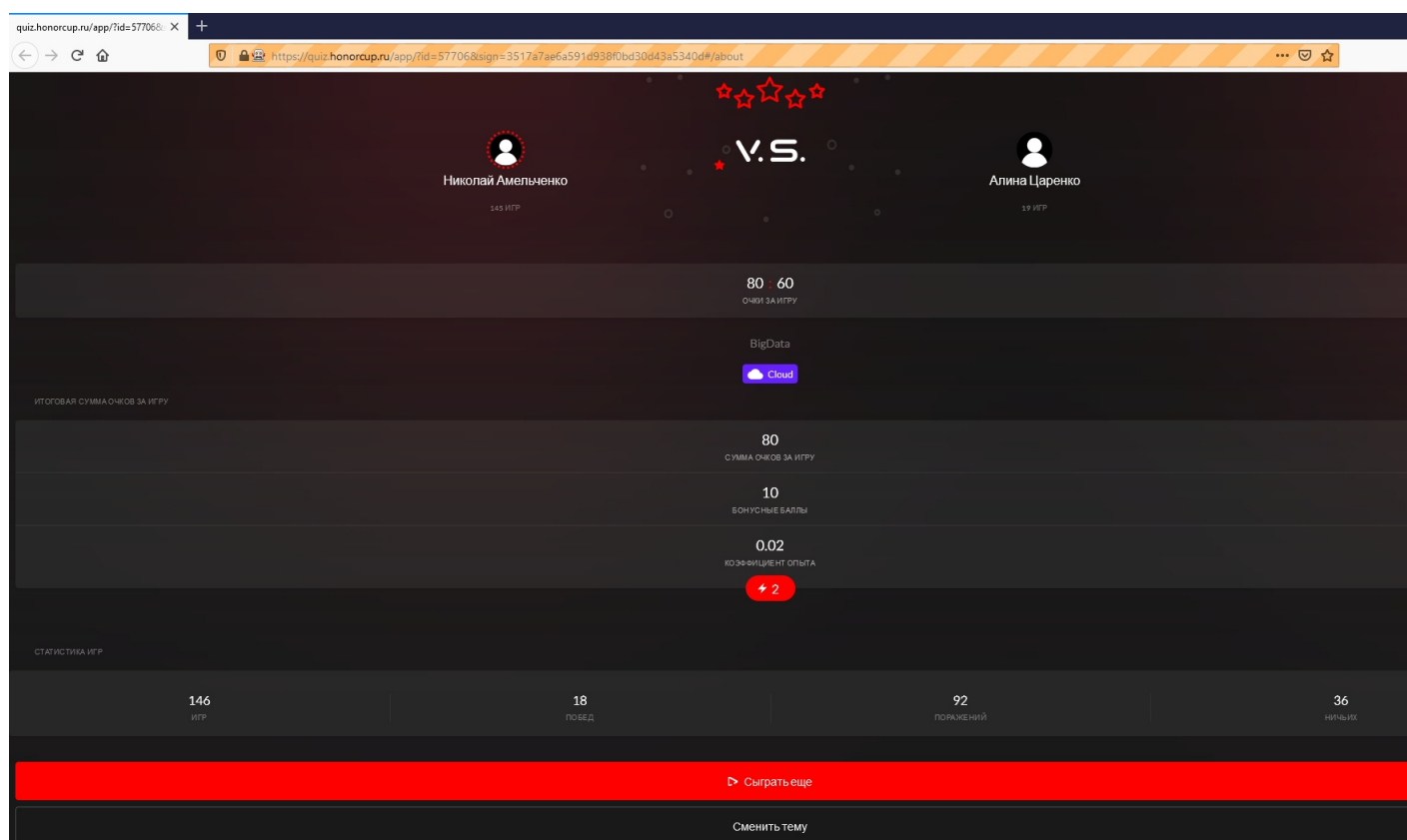


Найдя противника, начинаем сражение (функция play - логика функции описана в листинге) и кликаем по правильным ответам:



Одержав победу над противником стартуем следующую игру:

```
1 for j in range(10000): # проходим викторину условно к10 раз
2     time.sleep(25)
3     play() # играем
4     one_more = browser.find_elements_by_xpath('/html/body/app/div[1]/
5     result/div/div/div[9]/div[1]')
6     one_more[0].click() жмем# сыграть еще раз
```



1.3.1. Листинг программы

Листинг 1.1: Код работы бота

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on 10/22/2020 12:09 PM
4
5 @author: N. Amelchenko
6 """
7
8
9 #pip install selenium
10 # download geckodriver (https://github.com/mozilla/geckodriver/
    releases)
11
12
13 from selenium import webdriver подключаем# селениум
14 import time
15
16 #first of all, get a token
17
18 TOKEN = 'https://quiz.honorcup.ru/app/?id=57706&sign=3517
    a7ae6a591d938f0bd30d43a5340d' код# авторизации
19
20 #open a browser (FireFox)
  
```



```

21 browser = webdriver.Firefox(executable_path='geckodriver.exe')
    запускаем# браузер
22 browser.get(TOKEN) авторизуемся#
23 time.sleep(2) ручная# задержка
24 # #click battle_button
25 battle_button = browser.find_element_by_class_name('about__buttons')
    создаем# переменную содержащую адрес кнопки 'сражаться за кубок' для
    дальнейшего взаимодействия с ней
26 battle_button.click() кликаем# на полученную кнопку
27 time.sleep(2)
28
29 # #choose a category and theme
30 category = browser.find_elements_by_class_name('slider__item') адрес#
    раздела на сайте
31 category[3].click()
32
33 theme = browser.find_elements_by_class_name('profile__theme') адрес#
    темы викторины на сайте
34 theme[1].click()
35
36 categories_play_button = browser.find_element_by_xpath('/html/body/app
    /div[1]/nomination/div/div[2]/div[3]/div[2]/div/div/div[2]/div '
    )#('button-group-2x')
37 categories_play_button.click()
38
39 def play(s1=None): функция# содержащая в себе обработку сражения вплоть до
    следующей игры
40
41     for i in range(6): #6 раундов
42
43         k=0
44         round_question = browser.find_element_by_class_name('
game__question-text') вопрос#
45         round_answers = browser.find_elements_by_class_name('
game__answer') массив# ответов
46         s = round_question.text запоманием# вопрос
47         s0=s
48         f = open('base.txt', 'r') считываем# базу данных с целью поиска
    ответа
49         for line in f: проходим# файл построчно
50             if line.startswith(s): каждый# ответ построчно расположен в
    порядке: вопрос — ответ. Вопрос сравниваем с началом строки, а ответ с концом
51                 print('****')
52                 k = 1 запоманием# что ответ найден
53                 for ij in range(4): сравниваем# подлинный ответ с
    предложенными мя4 т кликаем на верный
54
55                 s1=round_answers[ij].text+'\n'
56                 if line.endswith(s1):
57                     round_answers[ij].click()

```

```

58         time.sleep(20)
59     f.close()
60     if k==0: Проверяем# отсутствие ответа в базе данных
61         points = browser.find_elements_by_class_name('game__user
-value')
62         print(points[0].text, 'до<< ответа')
63         a = int(points[0].text) запоминаем# наши баллы до клика,
        чтобы сравнить их для того, чтобы узнать правильно кликнули или нет
64         p=0
65         def skip(p): функция#, которая выбирает новый ответ,
        основываясь на предыдущих ошибках, с целью не повторить ошибку и сузить круг
        поиска
66             s1 = round_answers[p].text + '\n'
67             f = open('base_of_wrong.txt', 'r')
68             for line in f:
69                 if line.startswith(s):
70
71                     if line.endswith(s):
72
73                         print('ПРОПУСКАЕМ')
74                         p=p+1
75             f.close()
76         skip(p)
77         skip(p)
78         skip(p)
79
80         round_answers[p].click() выбираем# ответ
81         time.sleep(1)
82         points1 = browser.find_elements_by_class_name('
game__user-value') баллы# после
83         print(round_question.text)
84         print(round_answers[0].text)
85         b = int((points1[0].text))
86         print('A=', a, 'B=', b)
87         if b > a: если# ответ верный...
88             f = open('base.txt', 'a') дописываем# базу с
        правильными ответами
89             otvet = round_question.text + ' Ответ: ' +
round_answers[0].text + '\n'
90             f.write(otvet)
91             f.close()
92         else:
93             f = open('base_of_wrong.txt', 'a') иначе#
        дописываем базу наших ошибок, что бы не повторить их
94             otvet = round_question.text + ' Ответ: ' +
round_answers[0].text + '\n'
95             f.write(otvet)
96             f.close()
97
98         time.sleep(5)

```

```

99         if s0==round_question.text: сменился# ли вопрос? Для того
      что бы уменьшить задержку между ответами
100             time.sleep(10)
101         else:
102             continue
103
104
105 for j in range(10000): # проходим викторину условно к10 раз
106     time.sleep(25)
107     play() # играем
108     one_more = browser.find_elements_by_xpath('/html/body/app/div[1]/
      result/div/div/div[9]/div[1]')
109     one_more[0].click() жмем# сыграть еще раз

```

1.3.2. Результаты

Полученная база данных:

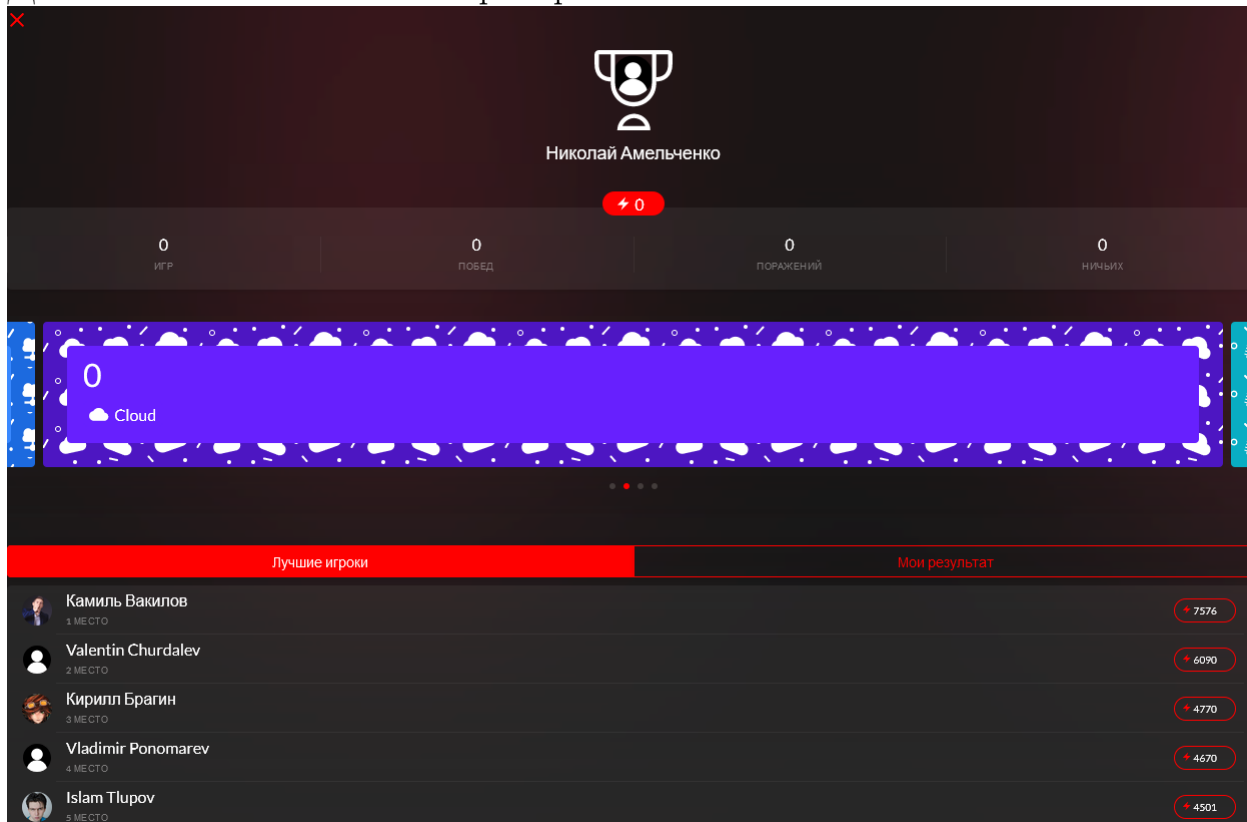
Flink — это система обработки потоков в реальном времени. Ответ: Верно

Для каких целей не применяется Flink? Ответ: Обработка графов

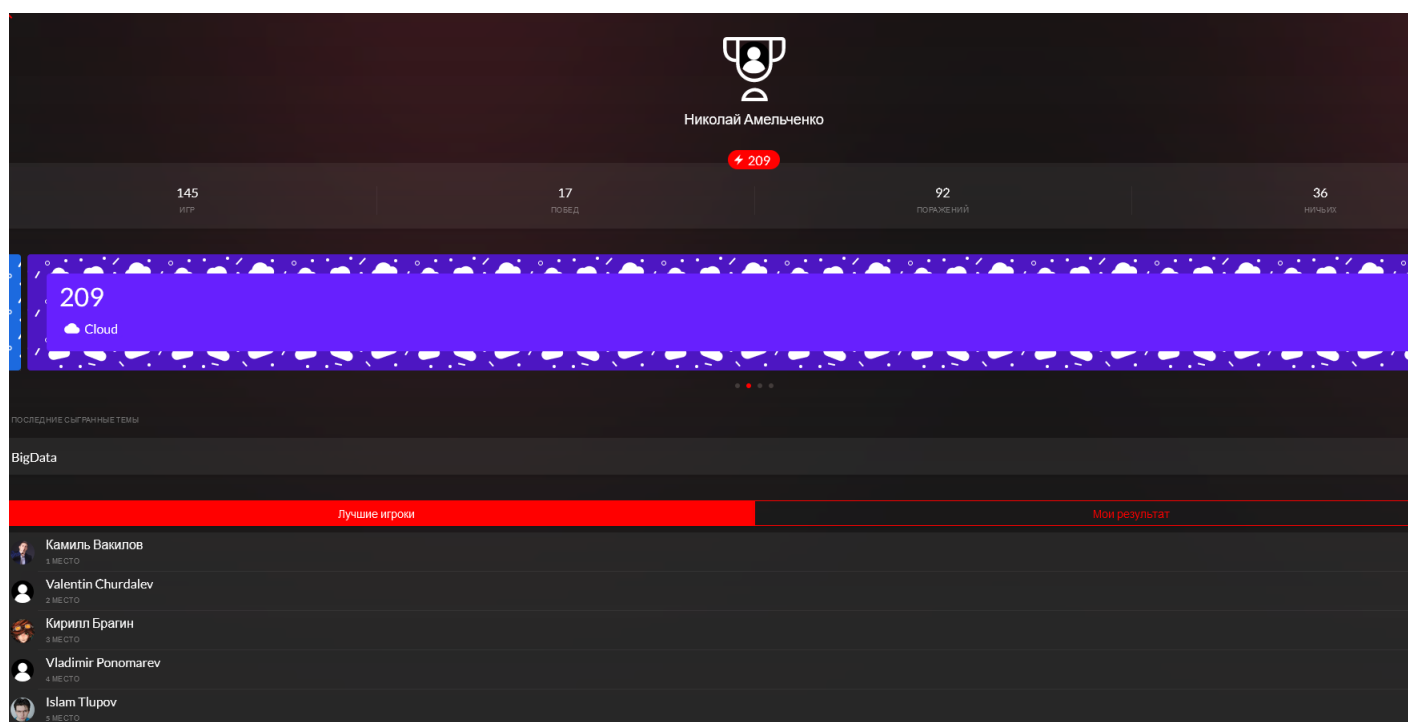
Какой источник (Source) необходимо использовать Flume для сбора последних 100 строк журналов? Ответ: Источник Eхес

...

До начала выполнения лабораторной:



После выполнения лабораторной:



1.4. Вывод

В ходе данной лабораторной я подтянул свои знания Python, научился через него взаимодействовать с браузером и создал самообучающегося бота способного после несколько часов тренировок выигрывать все викторины на Honor Cup