
ДОМАШНЕЕ ЗАДАНИЕ №10

«На прошлом уроке мы изучали изучали общие принципы современного автоматизированного тестирования чуть ближе к практике. В качестве ДЗ необходимо поинтересоваться принципами и азами современного автоматизированного тестирования и предложить несколько ссылок на современные технологии автоматизированного тестирования.

Итак, в качестве домашнего задания принимается несколько ссылок на статьи или видео в интернете, рассказывающие о современных тенденциях в автоматизированном тестировании, а также ссылки на технологии, автоматизированного тестирования, которые хотелось бы изучить.

Жду ваших ссылок, удачи!!!

1. Думаю полезно будет познакомиться с Pytest и работе с ним в Pycharm. Хотелось бы написать и запустить хотя бы один тест, а в идеале запустить экспортированный файл из Selenium IDE.

<https://www.youtube.com/watch?v=1HtEPEn4-LY>

2. [Статья о современных](#) инструментах тестирования, можно выбрать что-то наиболее актуальное для нас.

<https://www.g2.com/categories/automation-testing>

3. Хотелось бы разобраться с автотестами на Postman. Складывается ощущение что это базовый навык.

<https://www.youtube.com/watch?v=B2TxMRFBm8I&list=PLKbJd47KcbjvLgn-ukTvfpaGoXAqybza0>

<https://www.youtube.com/watch?v=30SOED9zV4c>

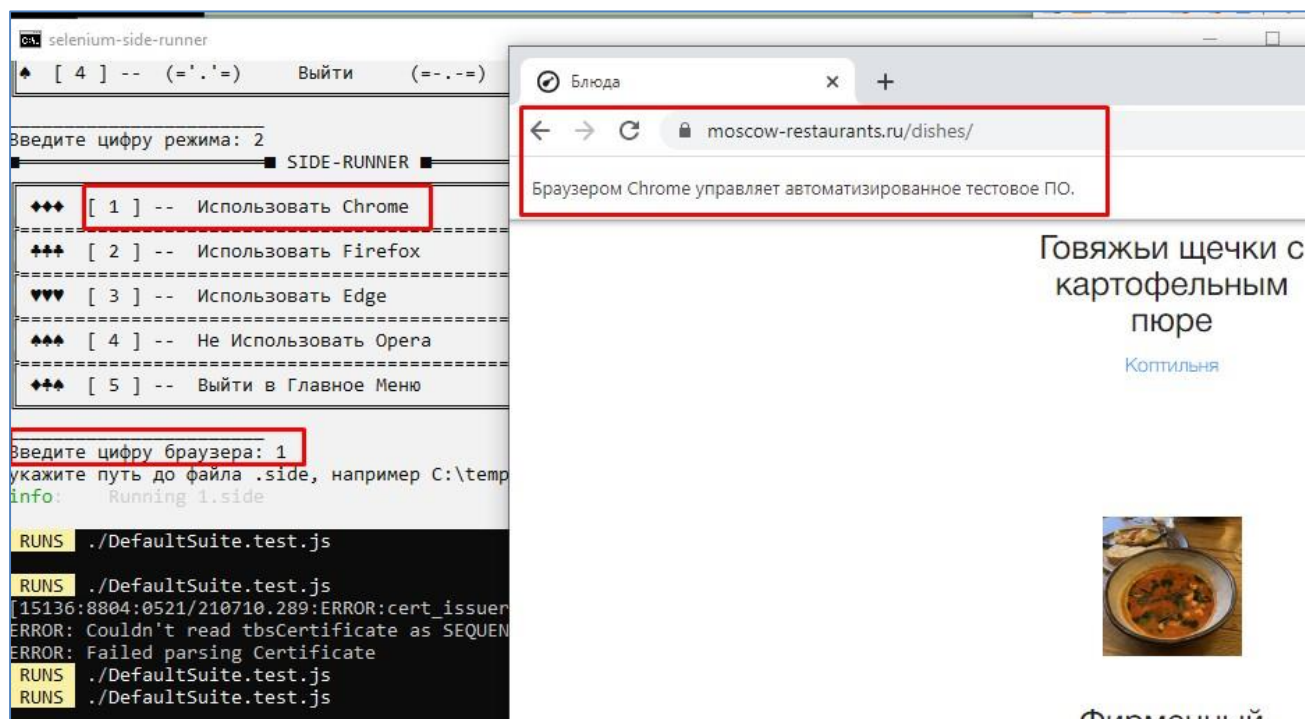
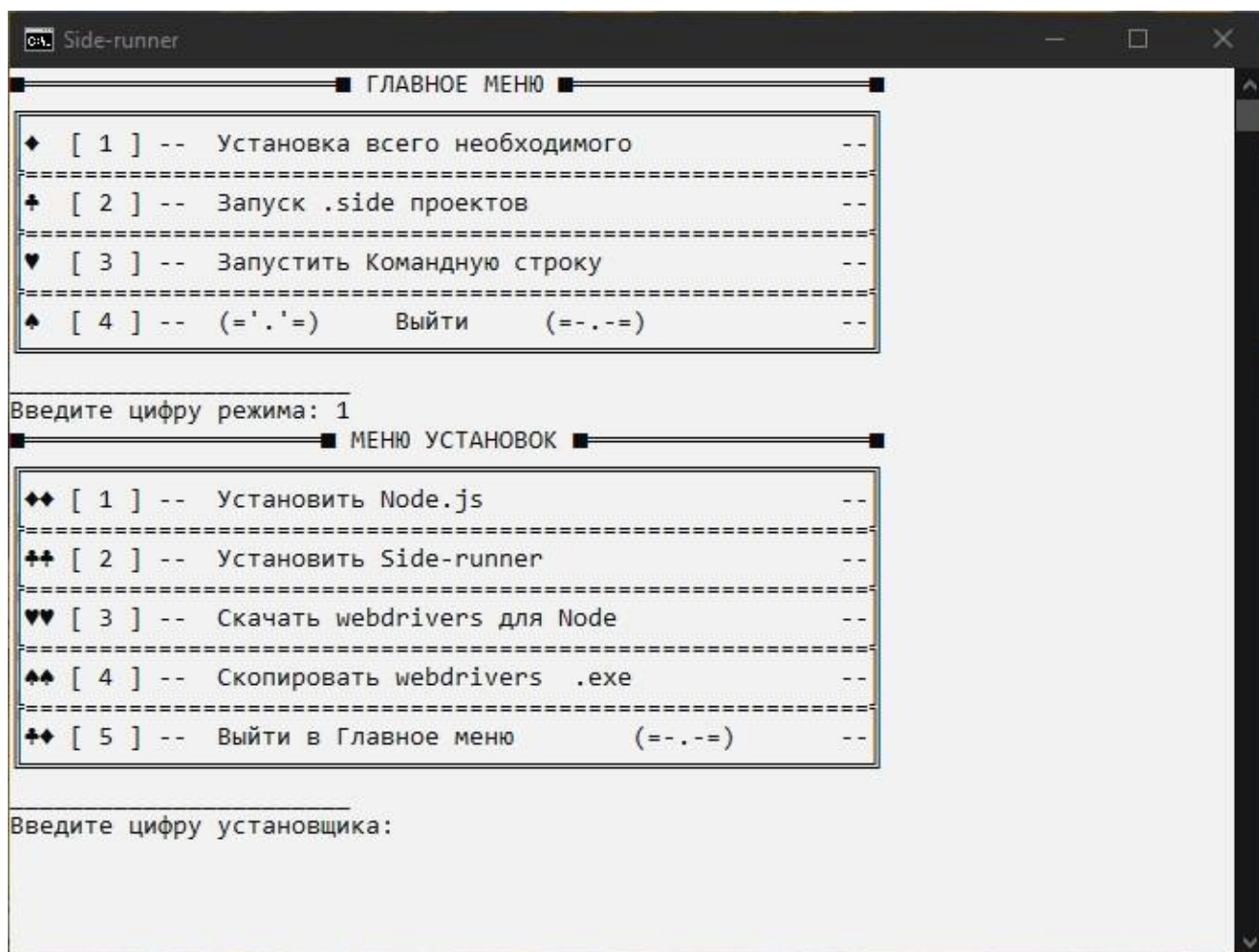
<https://www.youtube.com/watch?v=cp852oJWhPo>

4. Часто встречается упоминание о **Selenide**.

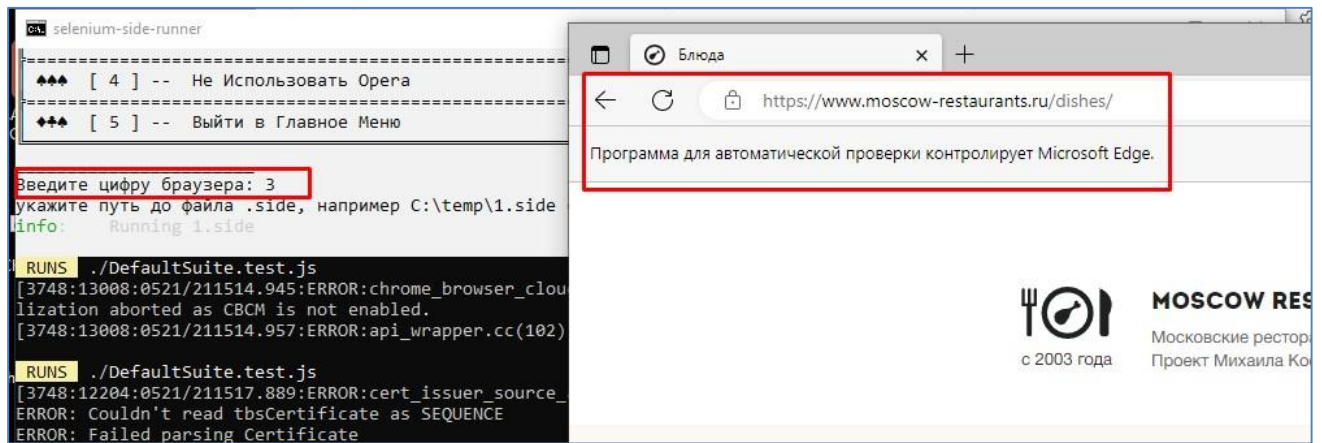
<https://www.youtube.com/watch?v=gVrvosAU0yg>

Думаю, хорошим источником для пополнения этого списка являются требования в вакансиях на hh.ru, что успел заметить, то и добавил сюда.

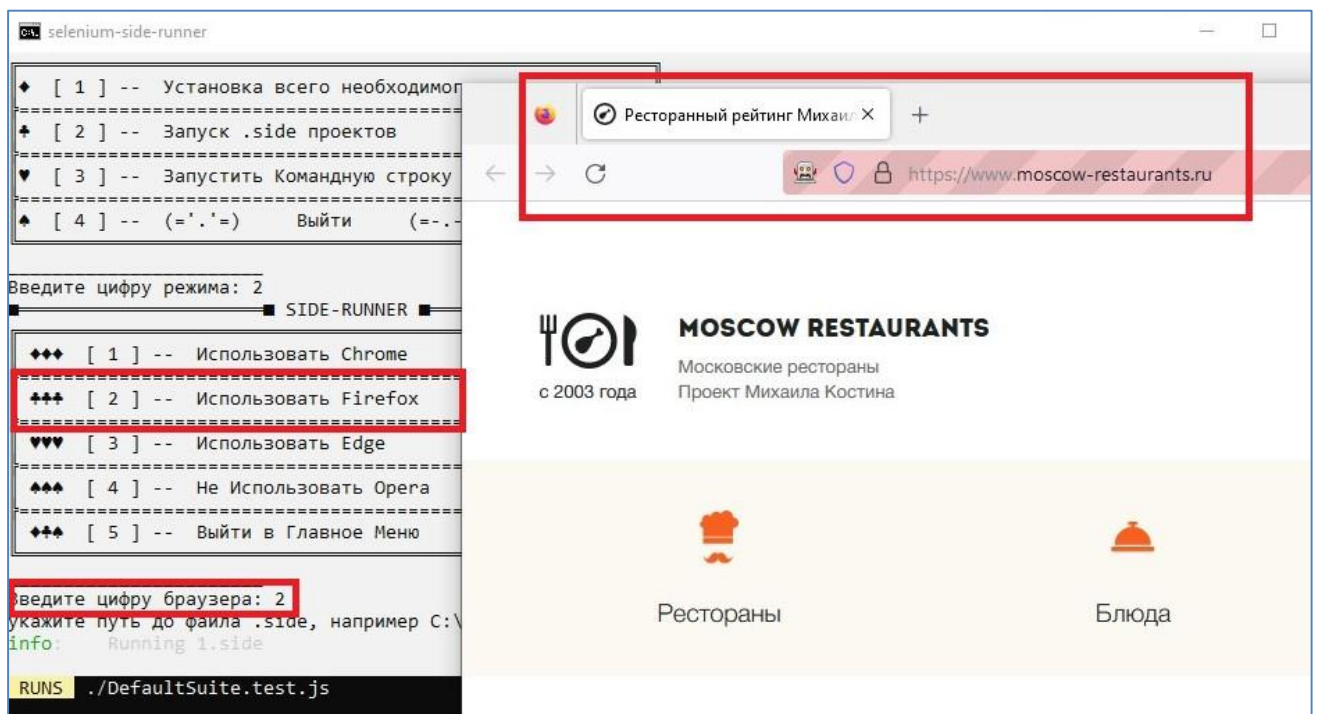
Side-runner.bat и скриншота запуска теста в браузерах



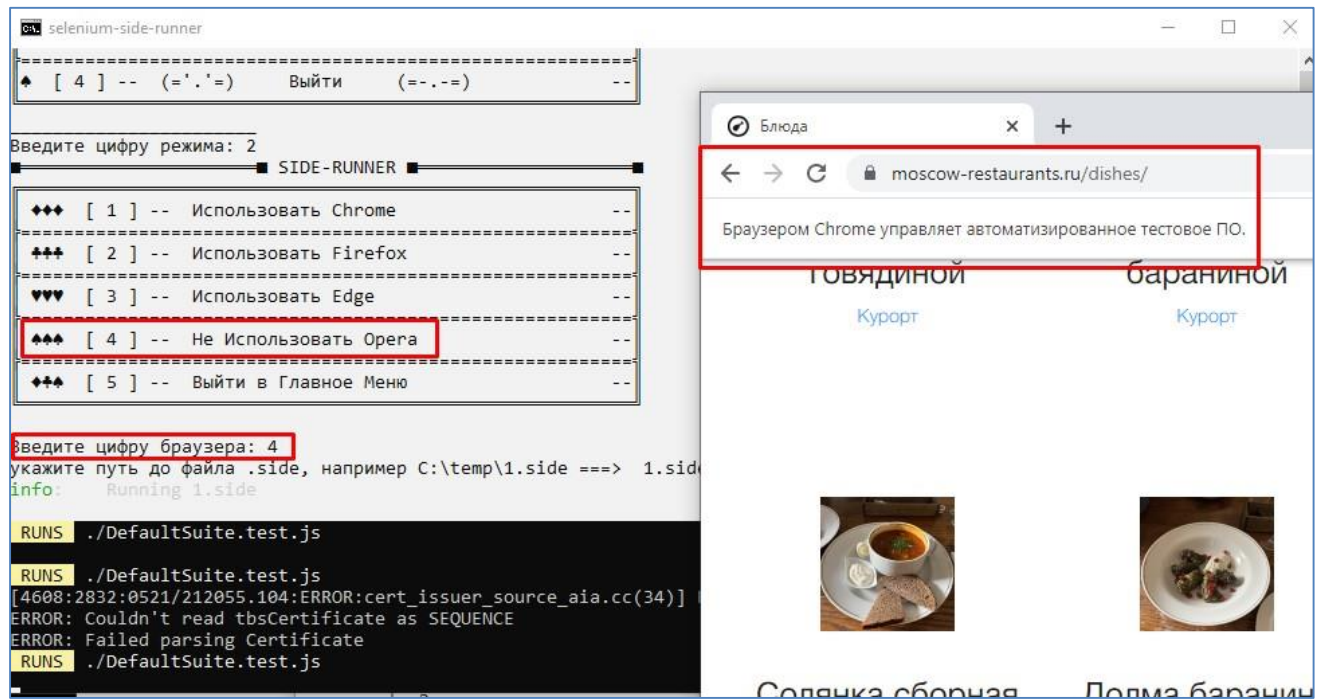
Chrome



Edge



Firefox



Opera, but Chrome

Side-runner.bat

```

1.@echo off
2.@chcp 866
3.
4.REM Запрос запуска с правами администратора
5.REM if not "%1=="am_admin" (powershell start -verb runas '%0' am_admin &
    exit /b)
6.
7.TITLE Side-runner
8.cls
9.color F0
10.     set PATH=%PATH%;C:\temp\driver
11.     set current=/d "%~dp0"
12.     cd %current%
13.
14.
15.     :mainMenu
16.
17.     echo ■■■ ГЛАВНОЕ МЕНЮ ■■■
18.     echo ════════════════════════════════════════
19.     echo
20.     ||──────────────────────────────────────────||
21. [ 1 ] -- Установка всего необходимого          --||
22.     ════════════════════════════════════════
23.     ||────────────────── [ 3 ] -- Запустить Командную
строку                      --||
24.     ════════════════════════════════════════
25.     || [ 4 ] -- (= '.' =)      Выйти      (= -. =)       --||
26.     ════════════════════════════════════════
27.
28.     GOTO insert
29.
30.
31.
32.     :insert
33.     REM Ввод данных
34.     echo _____
35.     set /p inserted="Введите цифру режима: "

```

```

36.
37.     REM Присвоение введённого значения переменной, которая может быть
        только числом
38.     set /a choise=%inserted%
39.
40.     REM Проверка данных на принадлежность к числам. Если присвоено не
        число, переменная имеет значение 0
41.     IF %choise% leq 0 (
42.         GOTO error
43.     )ELSE (
44.         GOTO action
45.     )
46.
47.     REM Условия работы программы
48.     :action
49.     IF %choise%==1 (
50.         GOTO setupMenu
51.     ) ELSE IF %choise%==2 (
52.         GOTO sideMenu
53.     ) ELSE IF %choise%==3 (
54.         GOTO cmdLine
55.     ) ELSE IF %choise%==4 (
56.         GOTO end
57.     ) ELSE GOTO mainMenu
58.
59.     REM Просмотр system32
60.     :system32
61.     set wind=C:\Windows\System32
62.     cd %wind%
63.     dir
64.     GOTO end
65.
66.     REM Сообщение об ошибке и выход
67.     :error
68.     echo
69.     echo Введённые данные некорректны. Выход из программы с ошибкой
70.     pause
71.     EXIT /b 128
72.
73.     REM Меню выбора установок и настроек

```

```

74.      :setupMenu
75.      echo  ┌───────────────────────────────────────────────────────────────────────────────────┐
76.      echo  │                                                                                   │
77.      echo  │                                                                                   │
78.      echo  │                                                                                   │
79.      echo  │ [ 2 ] -- Установить Side-runner                                     --||
80.      echo  │                                                                                   │
81.      echo  │───────────────────────────────────────────────────────────────────────────────────┐
82.      echo  │                                                                                   │
83.      echo  │ [ 4 ] -- Скопировать webdrivers .exe                                     --||
84.      echo  │                                                                                   │
85.      echo  │                                                                                   │
86.      echo  │───────────────────────────────────────────────────────────────────────────────────┐
87.      echo  │                                                                                   │
88.      GOTO setupInsert
89.
90.      REM Ввод и проверка данных пользователя
91.      :setupInsert
92.      echo  _____
93.      set /p inserted="Введите цифру установщика: "
94.
95.      REM Присвоение введённого значения переменной, которая может быть
только числом
96.      set /a chose=%inserted%
97.
98.      REM Проверка данных на принадлежность к числам. Если присвоено не
число, переменная имеет значение 0
99.      IF %chose% leq 0 (
100.         GOTO setupMenu
101.      )ELSE (
102.         GOTO setupAction
103.      )
104.
105.      REM Выбор функции установщика

```



```

106.      :setupAction
107.      IF %choise%==1 (
108.          GOTO nodeSetup
109.      ) ELSE IF %choise%==2 (
110.          GOTO sideSetup
111.      ) ELSE IF %choise%==3 (
112.          GOTO downloadWDrivers
113.      ) ELSE IF %choise%==4 (
114.          GOTO copyWDrivers
115.      ) ELSE IF %choise%==5 (
116.          GOTO mainMenu
117.      ) ELSE GOTO setupMenu
118.
119.      REM Запуск установки Node.js
120.      :nodeSetup
121.      node-v18.16.0-x64.msi
122.      GOTO setupMenu
123.
124.      REM Запуск установки selenium-side-runner
125.      :sideSetup
126.      npm install -g selenium-side-runner@3.17.0
127.      GOTO setupMenu
128.
129.      REM Скачивание веб-драйверов для Node
130.      :downloadWDrivers
131.      npm install -g chromedriver && npm install -g edgedriver && npm
install -g geckodriver && npm install -g opera && Side-runner.bat
132.
133.
134.      REM копирование скаченных ранее в папку, указанную в PATH
135.
136.      :copyWDrivers
137.      set DriverDir=C:\temp\driver
138.      IF NOT EXIST "%DriverDir%" mkdir C:\temp\driver
139.      REM set PATH=%PATH%;C:\temp\driver
140.      copy driver C:\temp\driver
141.      GOTO setupMenu
142.
143.
144.      REM Меню выбора режимов работы SIDE-RUNNER

```

```

145.      :sideMenu
146.      echo ┌───────────────────────────────────┐ SIDE-RUNNER ┌───────────────────────────────────┐
147.      echo └───────────────────────────────────┘
148.      echo ||
149.      echo ||=====||
150.      echo || [ 2 ] -- Использовать Firefox --||
151.      echo ||=====||
152.      echo ||
153.      echo ||=====||
154.      echo || [ 4 ] -- Не Использовать Opera --||
155.      echo ||=====||
156.      echo ||
157.      echo └───────────────────────────────────┘
158.
159.      GOTO sideInsert
160.
161.      :sideInsert
162.      REM Ввод данных
163.      echo _____
164.      set /p inserted="Введите цифру браузера: "
165.
166.      REM Присвоение введённого значения переменной, которая может быть
        только числом
167.      set /a choise=%inserted%
168.
169.      REM Проверка данных на принадлежность к числам. Если присвоено не
        число, переменная имеет значение 0
170.      IF %choise% leq 0 (
171.          GOTO sideMenu
172.      )ELSE (
173.          GOTO side-RunnerAction
174.      )

```

```

175.
176.
177.     :side-RunnerAction
178.     IF %choise%==1 (
179.         GOTO sideChrome
180.     ) ELSE IF %choise%==2 (
181.         GOTO sideFirefox
182.     ) ELSE IF %choise%==3 (
183.         GOTO sideEdge
184.     ) ELSE IF %choise%==4 (
185.         GOTO sideOpera
186.     ) ELSE IF %choise%==5 (
187.         GOTO mainMenu
188.     )ELSE GOTO sideMenu
189.
190.
191.
192.     :sideChrome
193.     set /p sidePath="укажите путь до файла .side, например
        C:\temp\1.side ==> "
194.     selenium-side-runner -c "browserName=chrome" %sidePath%
195.     echo Введите Side-runner.bat чтобы запустить программу снова или
        exit для выхода
196.
197.     :sideFirefox
198.     set /p sidePath="укажите путь до файла .side, например
        C:\temp\1.side ==> "
199.     selenium-side-runner -c "browserName=firefox" %sidePath%
200.     echo Введите Side-runner.bat чтобы запустить программу снова или
        exit для выхода
201.
202.     :sideEdge
203.     set /p sidePath="укажите путь до файла .side, например
        C:\temp\1.side ==> "
204.     selenium-side-runner -c "browserName=MicrosoftEdge" %sidePath%
205.
206.     :sideOpera
207.     set /p sidePath="укажите путь до файла .side, например
        C:\temp\1.side ==> "
208.     selenium-side-runner --c "browserName='opera'" %sidePath%

```

```
209.  
210.  
211.  
212.     :cmdLine  
213.     cmd  
214.  
215.  
216.     REM Завершение работы  
217.     :end  
218.     echo ████████████████████████████████████████████████████████  
219.     echo || Всем спасибо, все свободны! ||  
220.     echo ████████████████████████████████████████████████████████  
221.     pause  
222.     EXIT /b 0  
223.     echo on
```

```
1.import os
2.import shutil
3.from sys import argv
4.
5.
6.# Извлечение шагов В-файла в отдельный файл
7.def StepsFromB():
8.    os.rename('B.side', 'B.txt')
9.    b = open("B.txt", 'r').read()
10.        StepsB = open("StepsB.txt", 'w')
11.        cutB = str(b[b.index("tests"):b.index("suites:")] )
12.        cutB = str(',') + cutB[10:])
13.        StepsB.write(cutB)
14.        StepsB.close()
15.
16.    # Извлечение шагов А-файла в отдельный файл
17.    def HeadStepsFromA():
18.        os.rename('A.side', 'A.txt')
19.        a = open("A.txt", 'r').read()
20.        StepsA = open("StepsA.txt", 'w')
21.        cuttingA = str(a[:a.index("suites")])
22.        a = str(cuttingA[:-5])
23.        StepsA.write(a)
24.        StepsA.close()
25.
26.
27.    # Извлечение нижней части документа А
28.    def CutttingFlatA():
29.        a = open('A.txt', 'r').read()
30.        flat = open('Flat.txt', 'w')
31.        cutFlat = str('"' + a[a.index("suites"):])
32.        flat.write(cutFlat)
33.        flat.close()
34.
35.    #Слияние файла А и шагов из В
36.    def StepsAStepsB():
37.        stA = open('StepsA.txt', 'a+')
```

```

38.         stB = open("StepsB.txt", 'r')
39.         stA.write(stB.read())
40.         stA.close()
41.         stB.close()
42.
43.         #Слияние суммарного файла шагов А и В с нижним куском файла А
44.         def StepsAFlat():
45.             stA = open('StepsA.txt', 'a+')
46.             flat = open("Flat.txt", 'r')
47.             stA.write(flat.read())
48.             stA.close()
49.             flat.close()
50.
51.         #Создание конечного файла Final
52.         def CopyToFinal():
53.             with open('StepsA.txt', 'r') as steps, open('Final.txt', 'a+') as
                final:
54.                 for line in steps:
55.                     final.write(line)
56.
57.         # Удаление лишних файлов
58.         def DeleteTempFiles():
59.             os.remove('StepsA.txt')
60.             os.remove('StepsB.txt')
61.             os.remove('Flat.txt')
62.             os.remove('A.txt')
63.             os.remove('B.txt')
64.
65.         #Переименование конечного файла Final
66.         def RenameToSide():
67.             os.rename('Final.txt', 'Final.side')
68.
69.
70.
71.         def Main():
72.             StepsFromB()
73.             HeadStepsFromA()
74.             CutttingFlatA()
75.             StepsAStepsB()
76.             StepsAFlat()

```

```
77.      CopyToFinal ()
78.      DeleteTempFiles ()
79.      RenameToSide ()
80.
81.      Main ()
```