

Selenium - opis realizacji

12 maja 2020

Spis treści

1	Scenariusz Selenium IDE	2
2	Automatyzacja testów	4


1 Scenariusz Selenium IDE

Scenariusz wygląda następująco:

1. Driver przechodzi na stronę z **kursami walut**
2. Z kolumny *inv. 1.00 PLN* pobiera aktualne wartości walut
3. Porównuje z ostatnim wpisem do bazy danych
4. Zapisuje do bazy danych

Na potrzeby zapisu danych został utworzony prosty schemat bazy danych, z tabelami *raport*, *currency* oraz *currency_history*.


Tabela *raport* zawiera dzień oraz godzinę przeprowadzonego raportu wraz z odpowiadającym mu *id*:

Tabela:  raport

	id	day	hour
	Filtr	Filtr	Filtr
1	1	2020-05-07	13:07
2	2	2020-05-07	13:10
3	3	2020-05-07	13:18
4	4	2020-05-07	13:24

Rysunek 1: Struktura tabeli *raport*


Tabela *currency* zawiera nazwę waluty oraz jej ostatnio odnotowaną wartość:

Tabela:  currency

	name	last_value
	Filtr	Filtr
1	US Dollar	4.193516
2	Euro	4.559526
3	British Pound	5.170216
4	Indian Rupee	0.05576
5	Australian Dol...	2.732965
6	Canadian Dollar	2.996713

Rysunek 2: Struktura tabeli *currency*

Tabela *currency_history* zawiera kolumnę *currency_id*, która jest kluczem obcym z tabeli *currency* oraz numer raportu - klucz obcy z tabeli *raport* - i wartość jaka została w jego trakcie odnotowana:

Tabela:  currency_history

	currency_id	raport_id	value
	Filtr	Filtr	Filtr
1	US Dollar	1	4.215504
2	Euro	1	4.549037
3	British Pound	1	5.212572
4	Indian Rupee	1	0.055504
5	Australian Dol...	1	2.723774
6	Canadian Dollar	1	2.993453

Rysunek 3: Struktura tabeli *currency_history*

Działanie skryptu wygląda tak (mając na początku pustą bazę danych), że do bazy danych jest dodawany nowy raport i zaczytywane są kolumny z nazwą waluty oraz z aktualną wartością. Sprawdzane jest czy waluta o tej nazwie jest już przechowywana w bazie danych - jeśli nie to jest dodawana, a następnie do tabeli z historycznymi wartościami waluty dodawane są *id* raportu oraz zaczytana wartość. Następnie aktualna wartość waluty jest porównywana z ostatnią odnotowaną (*last_value* z tabeli *currency*) oraz na konsolę wyrzucany jest odpowiedni log. Po wszystkim nowa wartość zastępuje starą we wspomnianej kolumnie *last_value*.

Po przeprowadzeniu paru testów, wywołując dane zapytanie można sprawdzić historię danej waluty:

```

1 SELECT currency_id, (SELECT day FROM raport where id = raport_id) as day,
2                      (SELECT hour FROM raport where id = raport_id) as hour , value
3 FROM currency_history
4 WHERE currency_id
5 LIKE "us dollar"

```

	currency_id	day	hour	value
15	US Dollar	2020-05-09	22:04	4.200938
16	US Dollar	2020-05-09	23:55	4.20088
17	US Dollar	2020-05-09	23:56	4.200268
18	US Dollar	2020-05-09	23:57	4.200268
19	US Dollar	2020-05-09	23:58	4.200268
20	US Dollar	2020-05-09	23:59	4.200268
21	US Dollar	2020-05-11	23:03	4.218
22	US Dollar	2020-05-11	23:05	4.217892
23	US Dollar	2020-05-12	11:25	4.210353
24	US Dollar	2020-05-12	12:00	4.206208
25	US Dollar	2020-05-12	13:00	4.201147
26	US Dollar	2020-05-12	14:00	4.190906
27	US Dollar	2020-05-12	15:00	4.193355
28	US Dollar	2020-05-12	16:30	4.19878
29	US Dollar	2020-05-12	17:00	4.193516

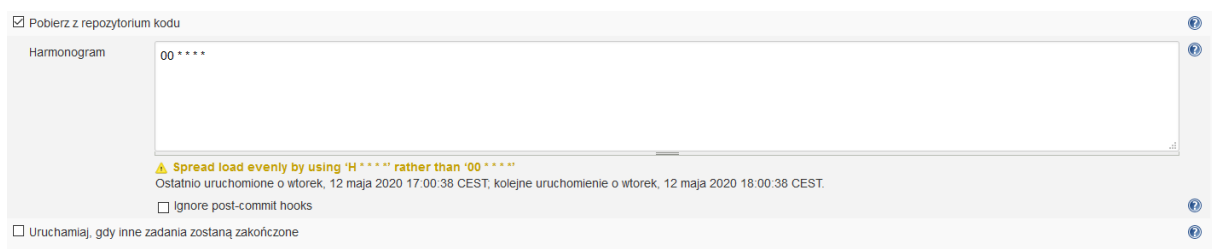
Rysunek 4: Historyczne wartości dolara w porównaniu do złotówki

2 Automatyzacja testów

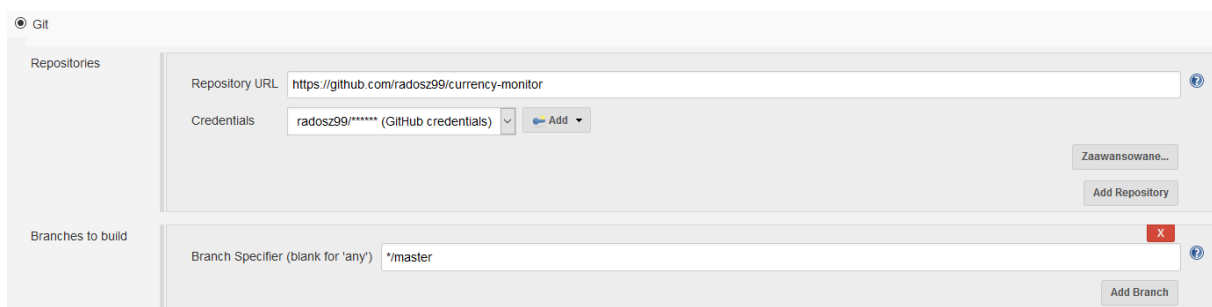
W celu automatyzacji testów powstał projekt na lokalnie postawionym Jenkinsie, który periodycznie - co godzinę - miał uruchamiać skrypt. Repozytorium zostało przeniesione na **GitHub'a**. W konfiguracji projektu w sekcji *Budowanie* został dodany nowy krok typu *Execute Windows batch command*:

```
1 python src/main.py
2 git checkout master
3 git add resources/currency_history.db
4 git commit -m "Actualize data"
5 git push
```

Po uprzednim dodaniu *credentials* do GitHub'a zostało ustawione cykliczne pobieranie z repozytorium kodu



Rysunek 5: Ustawienie cyklicznego wykonywania testów



Rysunek 6: *Credentials* do GitHub'a

Na początku pobierana jest aktualna wersja zdalnego repozytorium (wraz z bazą danych), wykonywany jest skrypt, który lokalnie aktualizuje bazę danych i dodaje ją - już zaktualizowaną - do zdalnego repozytorium, dzięki czemu nowy *pipeline* ma ciągle aktualną wersję bazy danych. Dla każdej waluty wyrzucany jest odpowiedni log z informacją czy wartość waluty uległa od czasu ostatniego testu zmianie (z reguły tak).

Na następnej stronie zamieszczony jest log konsoli z przebiegu wybranego *pipeline'a*.

```



















1 Uruchomione przez zmian w repozytorium kodu
2 Running as SYSTEM
3 Building in workspace E:\Jenkins\workspace\Selenium_regression_testing
4 using credential GitHub
5 Cloning the remote Git repository
6 Cloning repository https://github.com/radosz99/currency-monitor
7 > git.exe init E:\Jenkins\workspace\Selenium_regression_testing # timeout=10
8 Fetching upstream changes from https://github.com/radosz99/currency-monitor
9 > git.exe --version # timeout=10
10 using GIT_ASKPASS to set credentials GitHub credentials
11 > git.exe fetch --tags --force --progress -- https://github.com/radosz99/currency-monitor +refs
    /heads/*:refs/remotes/origin/* # timeout=10
12 > git.exe config remote.origin.url https://github.com/radosz99/currency-monitor # timeout=10
13 > git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
14 > git.exe config remote.origin.url https://github.com/radosz99/currency-monitor # timeout=10
15 Fetching upstream changes from https://github.com/radosz99/currency-monitor
16 using GIT_ASKPASS to set credentials GitHub credentials
17 > git.exe fetch --tags --force --progress -- https://github.com/radosz99/currency-monitor +refs
    /heads/*:refs/remotes/origin/* # timeout=10
18 > git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
19 > git.exe rev-parse "refs/remotes/origin/origin/master^{commit}" # timeout=10
20 Checking out Revision 0fc4400560c245cd3d08e5f13cbc5a9011c19271 (refs/remotes/origin/master)
21 > git.exe config core.sparsecheckout # timeout=10
22 > git.exe checkout -f 0fc4400560c245cd3d08e5f13cbc5a9011c19271 # timeout=10
23 Commit message: "Actualize data"
24 > git.exe rev-list --no-walk ea9d4fbcf0afc138a1be7a238b7b39f622fff143 # timeout=10
25 [Selenium_regression_testing] $ cmd /c call C:\Users\Radek\AppData\Local\Temp\
    jenkins1719702516192694482.bat
26
27 E:\Jenkins\workspace\Selenium_regression_testing>python src/main.py
28
29 CHANGE OCCURED | Currency US Dollar, old - 4.19878, new - 4.193516
30 CHANGE OCCURED | Currency Euro, old - 4.561469, new - 4.559526
31 CHANGE OCCURED | Currency British Pound, old - 5.172549, new - 5.170216
32 CHANGE OCCURED | Currency Indian Rupee, old - 0.0558, new - 0.05576
33 CHANGE OCCURED | Currency Australian Dollar, old - 2.731665, new - 2.732965
34 CHANGE OCCURED | Currency Canadian Dollar, old - 2.998059, new - 2.996713
35 CHANGE OCCURED | Currency Singapore Dollar, old - 2.965174, new - 2.962291
36 CHANGE OCCURED | Currency Swiss Franc, old - 4.337318, new - 4.336133
37 CHANGE OCCURED | Currency Malaysian Ringgit, old - 0.970156, new - 0.968963
38 CHANGE OCCURED | Currency Japanese Yen, old - 0.039118, new - 0.039105
39 CHANGE OCCURED | Currency Argentine Peso, old - 0.06225, new - 0.062166
40 CHANGE OCCURED | Currency Bahraini Dinar, old - 11.166969, new - 11.152969
41 CHANGE OCCURED | Currency Botswana Pula, old - 0.34518, new - 0.344667
42 CHANGE OCCURED | Currency Brazilian Real, old - 0.727723, new - 0.726066
43 CHANGE OCCURED | Currency Bruneian Dollar, old - 2.965174, new - 2.962291
44 CHANGE OCCURED | Currency Bulgarian Lev, old - 2.332242, new - 2.331249
45 NO CHANGE | Currency Chilean Peso, old - 0.005101, new - 0.005101
46 CHANGE OCCURED | Currency Chinese Yuan Renminbi, old - 0.592801, new - 0.592211
47 CHANGE OCCURED | Currency Colombian Peso, old - 0.001083, new - 0.001081
48 CHANGE OCCURED | Currency Croatian Kuna, old - 0.602926, new - 0.60264
49 CHANGE OCCURED | Currency Czech Koruna, old - 0.166255, new - 0.166572
50 CHANGE OCCURED | Currency Danish Krone, old - 0.611757, new - 0.611488
51 CHANGE OCCURED | Currency Emirati Dirham, old - 1.143303, new - 1.14187
52 CHANGE OCCURED | Currency Hong Kong Dollar, old - 0.541729, new - 0.541033
53 CHANGE OCCURED | Currency Hungarian Forint, old - 0.013001, new - 0.012994
54 CHANGE OCCURED | Currency Icelandic Krona, old - 0.028741, new - 0.028733
55 NO CHANGE | Currency Indonesian Rupiah, old - 0.000282, new - 0.000282
56 NO CHANGE | Currency Iranian Rial, old - 0.0001, new - 0.0001
57 CHANGE OCCURED | Currency Israeli Shekel, old - 1.197339, new - 1.194137
58 CHANGE OCCURED | Currency Kazakhstani Tenge, old - 0.009974, new - 0.009962
59 CHANGE OCCURED | Currency Kuwaiti Dinar, old - 13.575989, new - 13.5596
60 CHANGE OCCURED | Currency Libyan Dinar, old - 2.957574, new - 2.953865
61 CHANGE OCCURED | Currency Mauritian Rupee, old - 0.105475, new - 0.105187
62 CHANGE OCCURED | Currency Mexican Peso, old - 0.174627, new - 0.17453
63 CHANGE OCCURED | Currency Nepalese Rupee, old - 0.034712, new - 0.034688

```

```

64 CHANGE OCCURED | Currency New Zealand Dollar, old - 2.562314, new - 2.56316
65 CHANGE OCCURED | Currency Norwegian Krone, old - 0.41302, new - 0.413163
66 CHANGE OCCURED | Currency Omani Rial, old - 10.920105, new - 10.906414
67 CHANGE OCCURED | Currency Pakistani Rupee, old - 0.026174, new - 0.026149
68 CHANGE OCCURED | Currency Philippine Peso, old - 0.08369, new - 0.083546
69 CHANGE OCCURED | Currency Qatari Riyal, old - 1.153511, new - 1.152065
70 CHANGE OCCURED | Currency Romanian New Leu, old - 0.94421, new - 0.943792
71 CHANGE OCCURED | Currency Russian Ruble, old - 0.057381, new - 0.057387
72 CHANGE OCCURED | Currency Saudi Arabian Riyal, old - 1.119675, new - 1.118271
73 CHANGE OCCURED | Currency South African Rand, old - 0.230115, new - 0.230037
74 CHANGE OCCURED | Currency South Korean Won, old - 0.003431, new - 0.003429
75 CHANGE OCCURED | Currency Sri Lankan Rupee, old - 0.022307, new - 0.022263
76 CHANGE OCCURED | Currency Swedish Krona, old - 0.430915, new - 0.431498
77 CHANGE OCCURED | Currency Taiwan New Dollar, old - 0.140403, new - 0.140219
78 CHANGE OCCURED | Currency Thai Baht, old - 0.130693, new - 0.130713
79 CHANGE OCCURED | Currency Trinidadian Dollar, old - 0.627658, new - 0.624649
80 CHANGE OCCURED | Currency Turkish Lira, old - 0.600401, new - 0.599827
81 CHANGE OCCURED | Currency Venezuelan Bolivar, old - 0.420404, new - 0.419876
82
83 E:\Jenkins\workspace\Selenium_regression_testing>git checkout master
84 M resources/currency_history.db
85 Branch 'master' set up to track remote branch 'master' from 'origin'.
86 Switched to a new branch 'master'
87
88 E:\Jenkins\workspace\Selenium_regression_testing>git add resources/currency_history.db
89 E:\Jenkins\workspace\Selenium_regression_testing>git commit -m "Actualize data"
90 [master c34c9fb] Actualize data
91 1 file changed, 0 insertions(+), 0 deletions(-)
92
93 E:\Jenkins\workspace\Selenium_regression_testing>git push
94 To https://github.com/radosz99/currency-monitor
95 0fc4400..c34c9fb master -> master
96
97 E:\Jenkins\workspace\Selenium_regression_testing>exit 0
98 Process leaked file descriptors. See https://jenkins.io/redirect/troubleshooting/process-leaked-
99 file-descriptors for more information
100
101 using credential GitHub
102 [WS-CLEANUP] Deleting project workspace...
103
104 [WS-CLEANUP] Deferred wipeout is used...
105
106 Finished: SUCCESS

```

Actualize data  radosz99 committed 1 hour ago	 c34c9fb	
Actualize data  radosz99 committed 2 hours ago	 0fc4400	
Actualize data  radosz99 committed 3 hours ago	 ea9d4fb	
Actualize data  radosz99 committed 4 hours ago	 7f79109	
Actualize data  radosz99 committed 5 hours ago	 b7cfa90	
Actualize data  radosz99 committed 6 hours ago	 fec4140	

Rysunek 7: Committed dokonane przez Jenkinsa