MINI LINK

mini link

Data CoLab Test Task

Task Report

Faranak Heydari

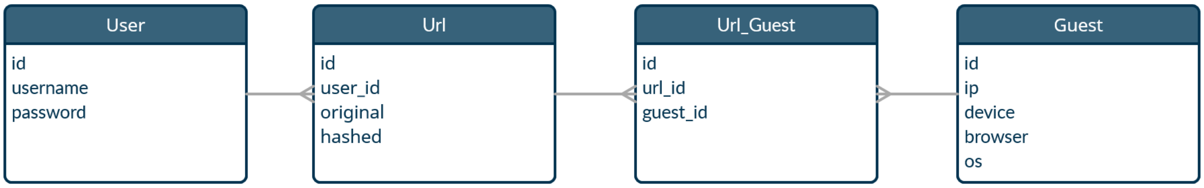
What is Mini Link?

The idea is a program that makes your URLs shorter. Basically, you enter a URL and receive a short version of it. What you need to do and what you get:

- First, you need to sign up, so you can create a short URL.
- Second, you will create your short URLs.
- After sharing them, you can get reports based on different parameters.

Behind the Scenes

Having in mind that our language is Python and our framework is Django Rest, in my opinion, the first step of the system design is designing its database. Based on the sign-up and analytics which are two use cases of our system, we know that there is a relation between our data so I considered an RDB (here I utilized from PostgreSQL) and designed an ERD for it:



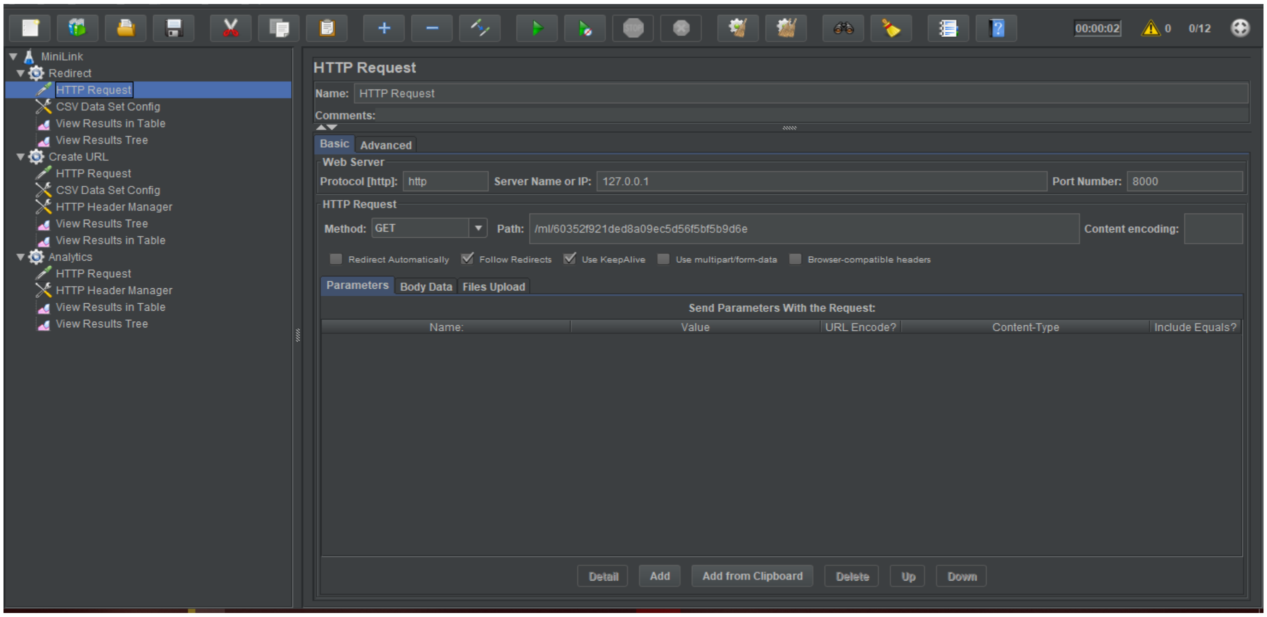
Then, I focused on the main use-case i.e. Redirecting. This is the most requested API among all APIs in Mini Link. Knowing that redirecting is nothing but a simple key-value like:

hashed_ur ---> original_url

I decided to use Redis for caching (storing) this key-value to speed up the redirecting process. The implementation is explained in the code.

Benchmarks and Improvements

To test Mini Link, I used Apache Jmeter with the following configuration:



- MiniLink
  - Redirect
    - HTTP Request
    - CSV Data Set Config
    - View Results in Table
    - View Results Tree
  - Create URL
    - HTTP Request
    - CSV Data Set Config
    - HTTP Header Manager
    - View Results Tree
    - View Results in Table
  - Analytics
    - HTTP Request
    - HTTP Header Manager
    - View Results in Table
    - View Results Tree

HTTP Request

Name: HTTP Request

Comments:

Web Server

Protocol [http]: http   Server Name or IP: 127.0.0.1   Port Number: 8000

HTTP Request

Method: GET   Path: /ml/60352f921ded8a09ec5d56f5bf5b9d6e   Content encoding:

□ Redirect Automatically ☑ Follow Redirects ☑ Use KeepAlive □ Use multipart/form-data □ Browser-compatible headers

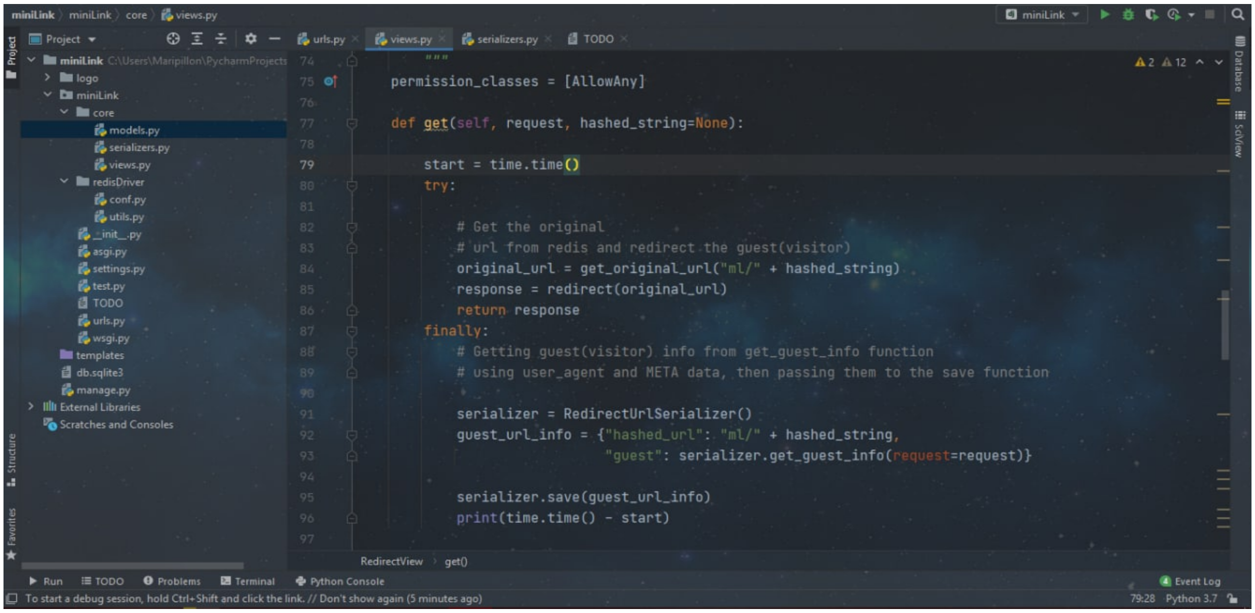
Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?

I set these configurations for each of the Thread Groups to send requests in one second (3 main use cases):

Redirect: 102 requests
Create Url: 6 requests
Analytics: 8 requests

To test the system, first of all, I used simple sequential coding for redirecting.

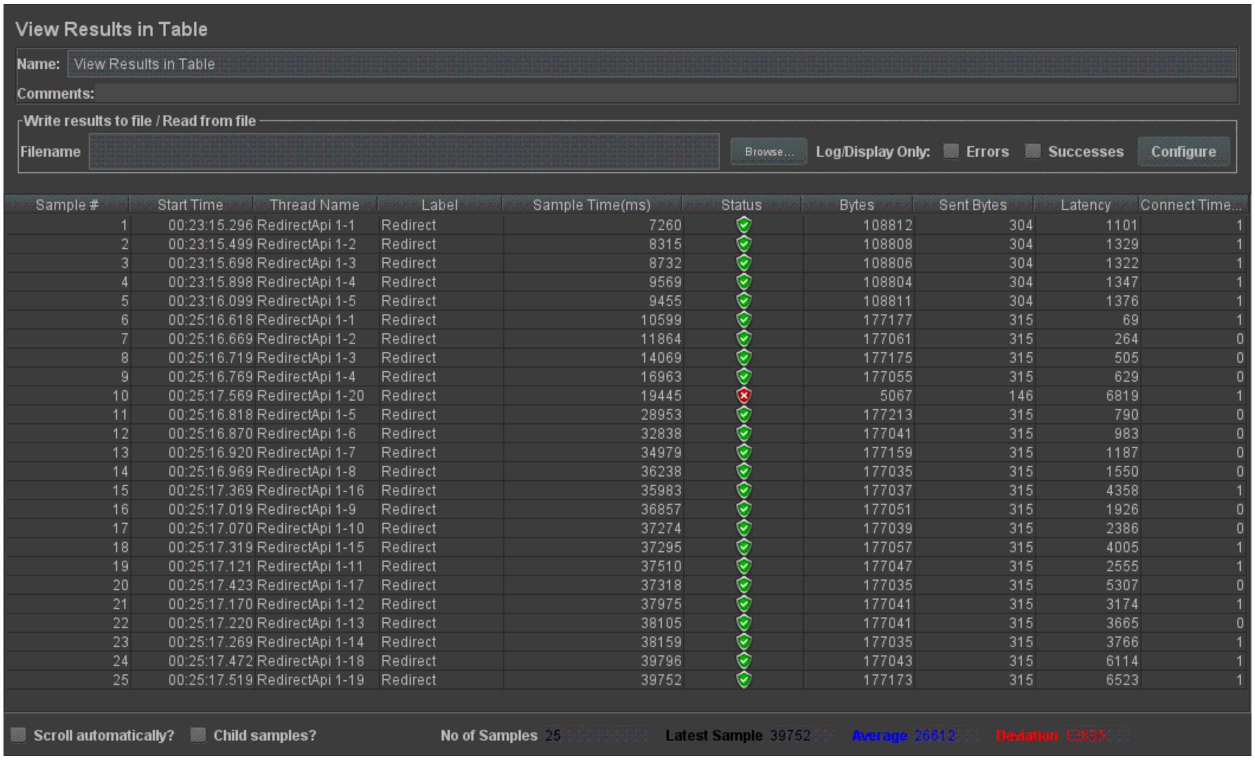


```
        """
    permission_classes = [AllowAny]

    def get(self, request, hashed_string=None):

        start = time.time()
        try:

            # Get the original
            # url from redis and redirect the guest(visitor)
            original_url = get_original_url("ml/" + hashed_string)
            response = redirect(original_url)
            return response
        finally:
            # Getting guest(visitor) info from get_guest_info function
            # using user_agent and META data, then passing them to the save function

            serializer = RedirectUrlSerializer()
            guest_url_info = {"hashed_url": "ml/" + hashed_string,
                              "guest": serializer.get_guest_info(request=request)}

            serializer.save(guest_url_info)
            print(time.time() - start)
```

To start a debug session, hold Ctrl+Shift and click the link. // Don't show again (5 minutes ago)

In the beginning, I did not get good results even before using the Jmeter (I just simply tested the API with Postman)
After that, I tried Multi-Threading. Still, it did not seem like a good solution because I did not have to use Multi-Threading and waste system resources just to simply write to the database.
I also tried Async IO, but I did not realize how I am going to use it in my problem.

In the mentioned sequential method, it took the program 1.29 seconds for one individual URL to redirect. Although It seemed like a good plan when I increased the number of requests, it got stuck and it took my program 7 to 113 seconds to redirect!
You can see the results in the following screenshots of Jmeter:



View Results in Table

Name: View Results in Table

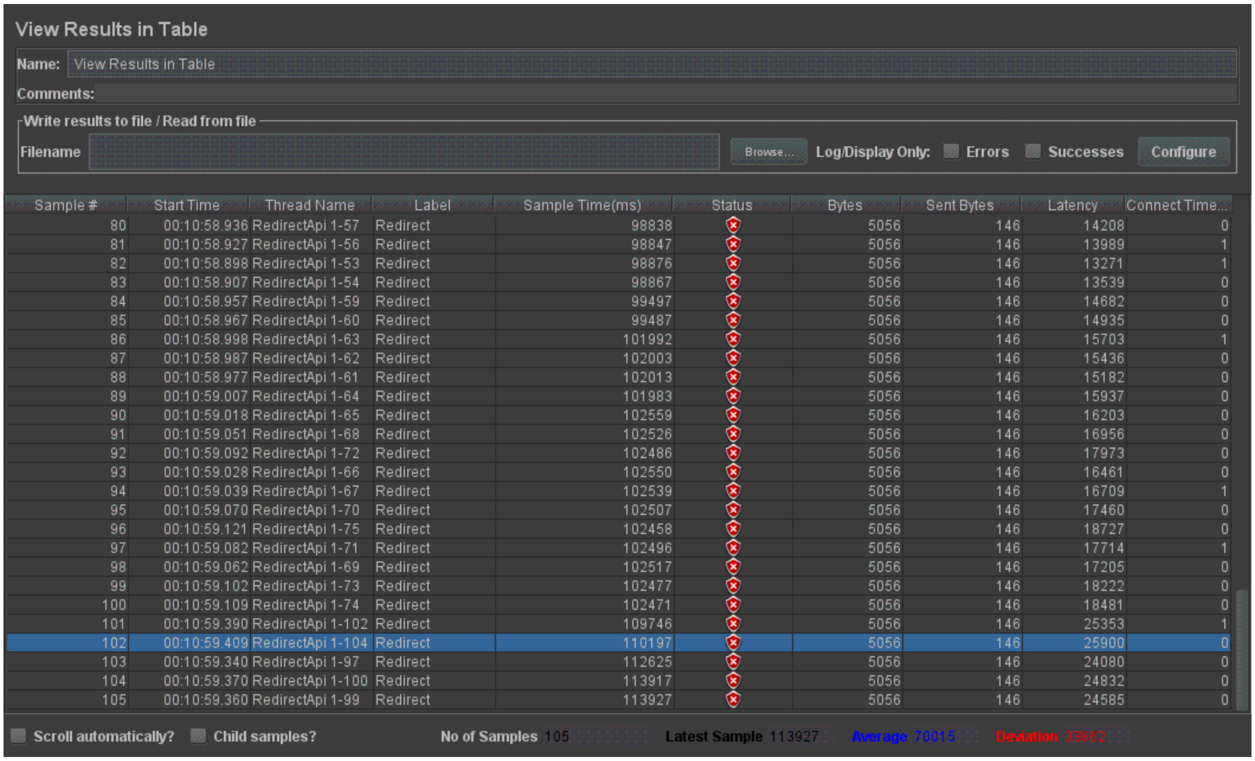
Comments:

Write results to file / Read from file

Filename

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time...
1	00:23:15.296	RedirectApi 1-1	Redirect	7260	✓	108812	304	1101	1
2	00:23:15.499	RedirectApi 1-2	Redirect	8315	✓	108808	304	1329	1
3	00:23:15.698	RedirectApi 1-3	Redirect	8732	✓	108806	304	1322	1
4	00:23:15.898	RedirectApi 1-4	Redirect	9569	✓	108804	304	1347	1
5	00:23:16.099	RedirectApi 1-5	Redirect	9455	✓	108811	304	1376	1
6	00:25:16.618	RedirectApi 1-1	Redirect	10599	✓	177177	315	69	1
7	00:25:16.669	RedirectApi 1-2	Redirect	11864	✓	177061	315	264	0
8	00:25:16.719	RedirectApi 1-3	Redirect	14069	✓	177175	315	505	0
9	00:25:16.769	RedirectApi 1-4	Redirect	16963	✓	177055	315	629	0
10	00:25:17.569	RedirectApi 1-20	Redirect	19445	✗	5067	146	6819	1
11	00:25:16.818	RedirectApi 1-5	Redirect	28953	✓	177213	315	790	0
12	00:25:16.870	RedirectApi 1-6	Redirect	32838	✓	177041	315	983	0
13	00:25:16.920	RedirectApi 1-7	Redirect	34979	✓	177159	315	1187	0
14	00:25:16.969	RedirectApi 1-8	Redirect	36238	✓	177035	315	1550	0
15	00:25:17.369	RedirectApi 1-16	Redirect	35983	✓	177037	315	4358	1
16	00:25:17.019	RedirectApi 1-9	Redirect	36857	✓	177051	315	1926	0
17	00:25:17.070	RedirectApi 1-10	Redirect	37274	✓	177039	315	2386	0
18	00:25:17.319	RedirectApi 1-15	Redirect	37295	✓	177057	315	4005	1
19	00:25:17.121	RedirectApi 1-11	Redirect	37510	✓	177047	315	2555	1
20	00:25:17.423	RedirectApi 1-17	Redirect	37318	✓	177035	315	5307	0
21	00:25:17.170	RedirectApi 1-12	Redirect	37975	✓	177041	315	3174	1
22	00:25:17.220	RedirectApi 1-13	Redirect	38105	✓	177041	315	3665	0
23	00:25:17.269	RedirectApi 1-14	Redirect	38159	✓	177035	315	3766	1
24	00:25:17.472	RedirectApi 1-18	Redirect	39796	✓	177043	315	6114	1
25	00:25:17.519	RedirectApi 1-19	Redirect	39752	✓	177173	315	6523	1

□ Scroll automatically? □ Child samples? No of Samples 25 Latest Sample 39752 Average 26612 Deviation 12635



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

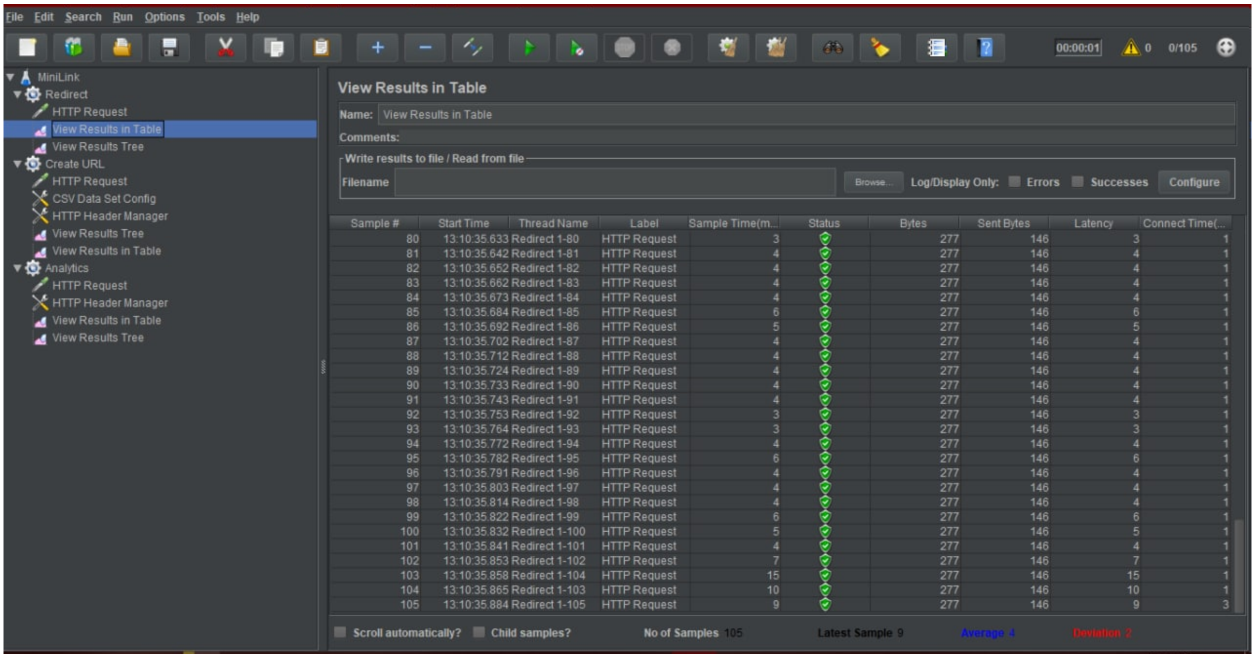
Filename

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time...
80	00:10:58.936	RedirectApi 1-57	Redirect	98838	✗	5056	146	14208	0
81	00:10:58.927	RedirectApi 1-56	Redirect	98847	✗	5056	146	13989	1
82	00:10:58.898	RedirectApi 1-53	Redirect	98876	✗	5056	146	13271	1
83	00:10:58.907	RedirectApi 1-54	Redirect	98867	✗	5056	146	13539	0
84	00:10:58.957	RedirectApi 1-59	Redirect	99497	✗	5056	146	14682	0
85	00:10:58.967	RedirectApi 1-60	Redirect	99487	✗	5056	146	14935	0
86	00:10:58.998	RedirectApi 1-63	Redirect	101992	✗	5056	146	15703	1
87	00:10:58.987	RedirectApi 1-62	Redirect	102003	✗	5056	146	15436	0
88	00:10:58.977	RedirectApi 1-61	Redirect	102013	✗	5056	146	15182	0
89	00:10:59.007	RedirectApi 1-64	Redirect	101983	✗	5056	146	15937	0
90	00:10:59.018	RedirectApi 1-65	Redirect	102559	✗	5056	146	16203	0
91	00:10:59.051	RedirectApi 1-68	Redirect	102526	✗	5056	146	16956	0
92	00:10:59.092	RedirectApi 1-72	Redirect	102486	✗	5056	146	17973	0
93	00:10:59.028	RedirectApi 1-66	Redirect	102550	✗	5056	146	16461	0
94	00:10:59.039	RedirectApi 1-67	Redirect	102539	✗	5056	146	16709	1
95	00:10:59.070	RedirectApi 1-70	Redirect	102507	✗	5056	146	17460	0
96	00:10:59.121	RedirectApi 1-75	Redirect	102458	✗	5056	146	18727	0
97	00:10:59.082	RedirectApi 1-71	Redirect	102496	✗	5056	146	17714	1
98	00:10:59.062	RedirectApi 1-69	Redirect	102517	✗	5056	146	17205	0
99	00:10:59.102	RedirectApi 1-73	Redirect	102477	✗	5056	146	18222	0
100	00:10:59.109	RedirectApi 1-74	Redirect	102471	✗	5056	146	18481	0
101	00:10:59.390	RedirectApi 1-102	Redirect	109746	✗	5056	146	25353	1
102	00:10:59.409	RedirectApi 1-104	Redirect	110197	✗	5056	146	25900	0
103	00:10:59.340	RedirectApi 1-97	Redirect	112625	✗	5056	146	24080	0
104	00:10:59.370	RedirectApi 1-100	Redirect	113917	✗	5056	146	24832	0
105	00:10:59.360	RedirectApi 1-99	Redirect	113927	✗	5056	146	24585	0

□ Scroll automatically? □ Child samples? No of Samples 105 Latest Sample 113927 Average 70015 Deviation 33862

It occurred to me that maybe the issue is on the database side. Therefore, I commented the Postgres part of my code and checked the results again. However, it did not change much.

Next, I thought it could be the Redis part of the code And I tried to use pooling as a solution. But before that, I commented all parts of the view and tested just the Redis part and I got these results:



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

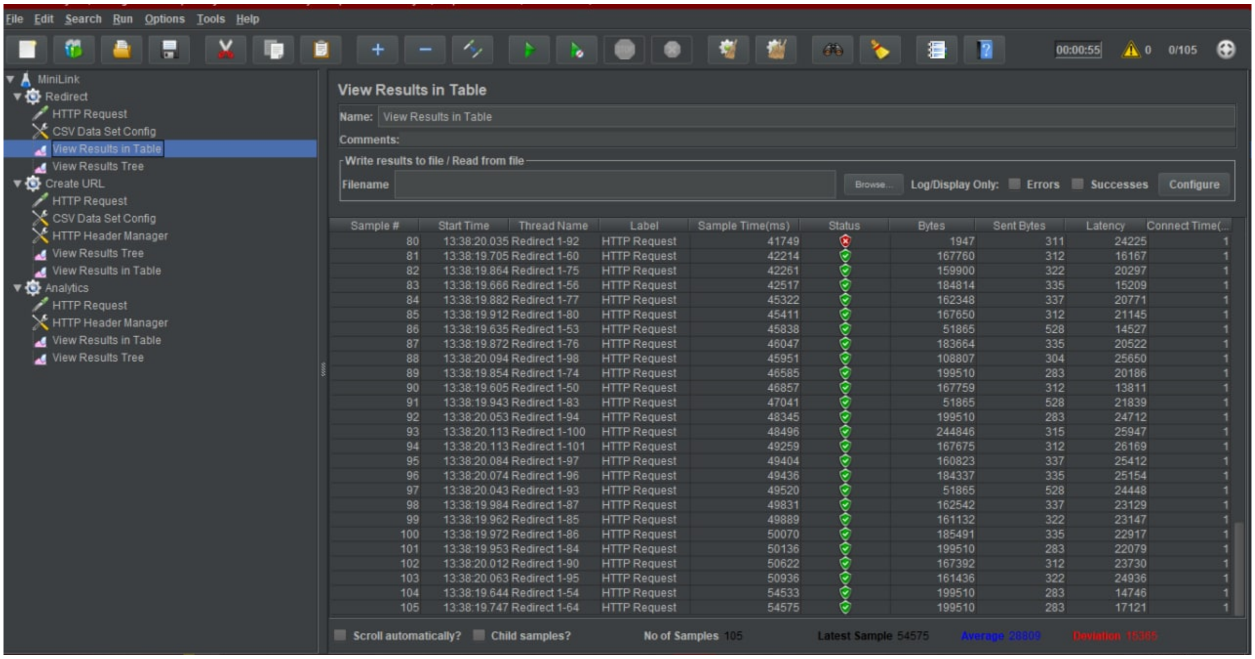
Filename

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(m...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
80	13:10:35.633	Redirect 1-80	HTTP Request	3	✓	277	146	3	1
81	13:10:35.642	Redirect 1-81	HTTP Request	4	✓	277	146	4	1
82	13:10:35.652	Redirect 1-82	HTTP Request	4	✓	277	146	4	1
83	13:10:35.662	Redirect 1-83	HTTP Request	4	✓	277	146	4	1
84	13:10:35.673	Redirect 1-84	HTTP Request	4	✓	277	146	4	1
85	13:10:35.684	Redirect 1-85	HTTP Request	6	✓	277	146	6	1
86	13:10:35.692	Redirect 1-86	HTTP Request	5	✓	277	146	5	1
87	13:10:35.702	Redirect 1-87	HTTP Request	4	✓	277	146	4	1
88	13:10:35.712	Redirect 1-88	HTTP Request	4	✓	277	146	4	1
89	13:10:35.724	Redirect 1-89	HTTP Request	4	✓	277	146	4	1
90	13:10:35.733	Redirect 1-90	HTTP Request	4	✓	277	146	4	1
91	13:10:35.743	Redirect 1-91	HTTP Request	4	✓	277	146	4	1
92	13:10:35.753	Redirect 1-92	HTTP Request	3	✓	277	146	3	1
93	13:10:35.764	Redirect 1-93	HTTP Request	3	✓	277	146	3	1
94	13:10:35.772	Redirect 1-94	HTTP Request	4	✓	277	146	4	1
95	13:10:35.782	Redirect 1-95	HTTP Request	6	✓	277	146	6	1
96	13:10:35.791	Redirect 1-96	HTTP Request	4	✓	277	146	4	1
97	13:10:35.803	Redirect 1-97	HTTP Request	4	✓	277	146	4	1
98	13:10:35.814	Redirect 1-98	HTTP Request	4	✓	277	146	4	1
99	13:10:35.822	Redirect 1-99	HTTP Request	6	✓	277	146	6	1
100	13:10:35.832	Redirect 1-100	HTTP Request	5	✓	277	146	5	1
101	13:10:35.841	Redirect 1-101	HTTP Request	4	✓	277	146	4	1
102	13:10:35.853	Redirect 1-102	HTTP Request	7	✓	277	146	7	1
103	13:10:35.858	Redirect 1-104	HTTP Request	15	✓	277	146	15	1
104	13:10:35.865	Redirect 1-103	HTTP Request	10	✓	277	146	10	1
105	13:10:35.884	Redirect 1-105	HTTP Request	9	✓	277	146	9	3

□ Scroll automatically? □ Child samples? No of Samples 105 Latest Sample 9 Average 4 Deviation 2

Seeing these results, made me realize that there is something wrong with the redirecting part of the program. After a few attempts, I realized that I have been requesting a fixed URL 105 times (LOL) and that was the bottleneck. As a solution, I made a dynamic URL and the results were 7 to 54 seconds for 105 requests.



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename

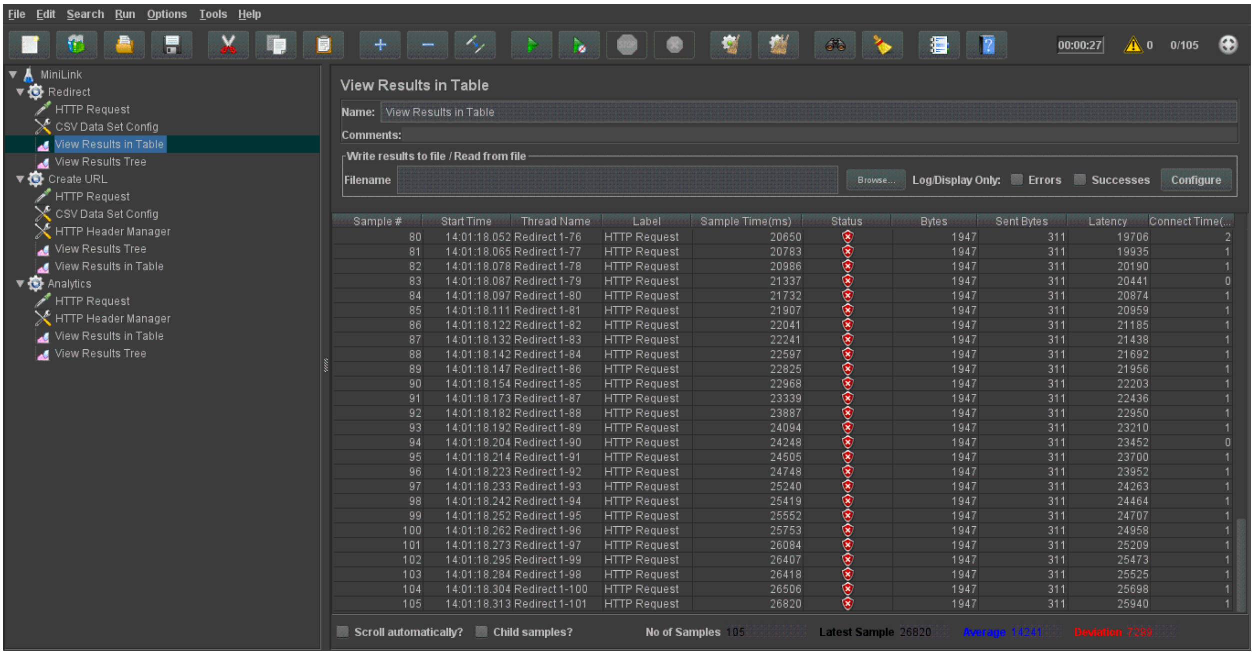
Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(...
80	13:38:20.035	Redirect 1-92	HTTP Request	41749	✗	1947	311	24225	1
81	13:38:19.705	Redirect 1-60	HTTP Request	42214	✓	167760	312	16167	1
82	13:38:19.864	Redirect 1-75	HTTP Request	42261	✓	159900	322	20297	1
83	13:38:19.666	Redirect 1-56	HTTP Request	42517	✓	184814	335	15209	1
84	13:38:19.882	Redirect 1-77	HTTP Request	45322	✓	162348	337	20771	1
85	13:38:19.912	Redirect 1-80	HTTP Request	45411	✓	167650	312	21145	1
86	13:38:19.635	Redirect 1-53	HTTP Request	45838	✓	51865	528	14527	1
87	13:38:19.872	Redirect 1-76	HTTP Request	46047	✓	183664	335	20522	1
88	13:38:20.094	Redirect 1-98	HTTP Request	45951	✓	108807	304	25650	1
89	13:38:19.854	Redirect 1-74	HTTP Request	46585	✓	199510	283	20186	1
90	13:38:19.605	Redirect 1-50	HTTP Request	46857	✓	167759	312	13811	1
91	13:38:19.943	Redirect 1-83	HTTP Request	47041	✓	51865	528	21839	1
92	13:38:20.053	Redirect 1-94	HTTP Request	48345	✓	199510	283	24712	1
93	13:38:20.113	Redirect 1-100	HTTP Request	48496	✓	244846	315	25947	1
94	13:38:20.113	Redirect 1-101	HTTP Request	49259	✓	167675	312	26169	1
95	13:38:20.084	Redirect 1-97	HTTP Request	49404	✓	160823	337	25412	1
96	13:38:20.074	Redirect 1-96	HTTP Request	49436	✓	184337	335	25154	1
97	13:38:20.043	Redirect 1-93	HTTP Request	49520	✓	51865	528	24448	1
98	13:38:19.984	Redirect 1-87	HTTP Request	49831	✓	162542	337	23129	1
99	13:38:19.962	Redirect 1-85	HTTP Request	49889	✓	161132	322	23147	1
100	13:38:19.972	Redirect 1-86	HTTP Request	50070	✓	185491	335	22917	1
101	13:38:19.953	Redirect 1-84	HTTP Request	50136	✓	199510	283	22079	1
102	13:38:20.012	Redirect 1-90	HTTP Request	50622	✓	167392	312	23730	1
103	13:38:20.063	Redirect 1-95	HTTP Request	50936	✓	161436	322	24936	1
104	13:38:19.644	Redirect 1-54	HTTP Request	54533	✓	199510	283	14746	1
105	13:38:19.747	Redirect 1-64	HTTP Request	54575	✓	199510	283	17121	1

□ Scroll automatically? □ Child samples? No of Samples 105 Latest Sample 54575 Average 28809 Deviation 15365

These results made me realize there is something wrong with some URLs. Because some of them were responding well at first but after some requests, they took a lot of time to respond.
It occurred to me that maybe some of these websites are taking too much time to load completely and because Jmeter waits to load the completed page, I decided to ignore the loading part of the sites and just test some test URLs; for example:
“http://redirect-faranak-test.test/”.

The duration of redirects considerably improved: 2-28 seconds.



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(...
80	14:01:18.052	Redirect 1-76	HTTP Request	20650	⊗	1947	311	19706	2
81	14:01:18.065	Redirect 1-77	HTTP Request	20783	⊗	1947	311	19935	1
82	14:01:18.078	Redirect 1-78	HTTP Request	20986	⊗	1947	311	20190	1
83	14:01:18.087	Redirect 1-79	HTTP Request	21337	⊗	1947	311	20441	0
84	14:01:18.097	Redirect 1-80	HTTP Request	21732	⊗	1947	311	20874	1
85	14:01:18.111	Redirect 1-81	HTTP Request	21907	⊗	1947	311	20959	1
86	14:01:18.122	Redirect 1-82	HTTP Request	22041	⊗	1947	311	21185	1
87	14:01:18.132	Redirect 1-83	HTTP Request	22241	⊗	1947	311	21438	1
88	14:01:18.142	Redirect 1-84	HTTP Request	22597	⊗	1947	311	21692	1
89	14:01:18.147	Redirect 1-86	HTTP Request	22825	⊗	1947	311	21956	1
90	14:01:18.154	Redirect 1-85	HTTP Request	22968	⊗	1947	311	22203	1
91	14:01:18.173	Redirect 1-87	HTTP Request	23339	⊗	1947	311	22436	1
92	14:01:18.182	Redirect 1-88	HTTP Request	23887	⊗	1947	311	22950	1
93	14:01:18.192	Redirect 1-89	HTTP Request	24094	⊗	1947	311	23210	1
94	14:01:18.204	Redirect 1-90	HTTP Request	24248	⊗	1947	311	23452	0
95	14:01:18.214	Redirect 1-91	HTTP Request	24505	⊗	1947	311	23700	1
96	14:01:18.223	Redirect 1-92	HTTP Request	24748	⊗	1947	311	23952	1
97	14:01:18.233	Redirect 1-93	HTTP Request	25240	⊗	1947	311	24263	1
98	14:01:18.242	Redirect 1-94	HTTP Request	25419	⊗	1947	311	24464	1
99	14:01:18.252	Redirect 1-95	HTTP Request	25552	⊗	1947	311	24707	1
100	14:01:18.262	Redirect 1-96	HTTP Request	25753	⊗	1947	311	24958	1
101	14:01:18.273	Redirect 1-97	HTTP Request	26084	⊗	1947	311	25209	1
102	14:01:18.295	Redirect 1-99	HTTP Request	26407	⊗	1947	311	25473	1
103	14:01:18.284	Redirect 1-98	HTTP Request	26418	⊗	1947	311	25525	1
104	14:01:18.304	Redirect 1-100	HTTP Request	26506	⊗	1947	311	25698	1
105	14:01:18.313	Redirect 1-101	HTTP Request	26820	⊗	1947	311	25940	1

□ Scroll automatically? □ Child samples? No of Samples 105 Latest Sample 26820 Average 14241 Deviation 7299

Despite the improvements compared with the previous methods, the speed up was not enough. After some research, I came up with this solution where I could use a queue for sending and running my Postgres writings through the queue in the background and make the URL redirections faster.

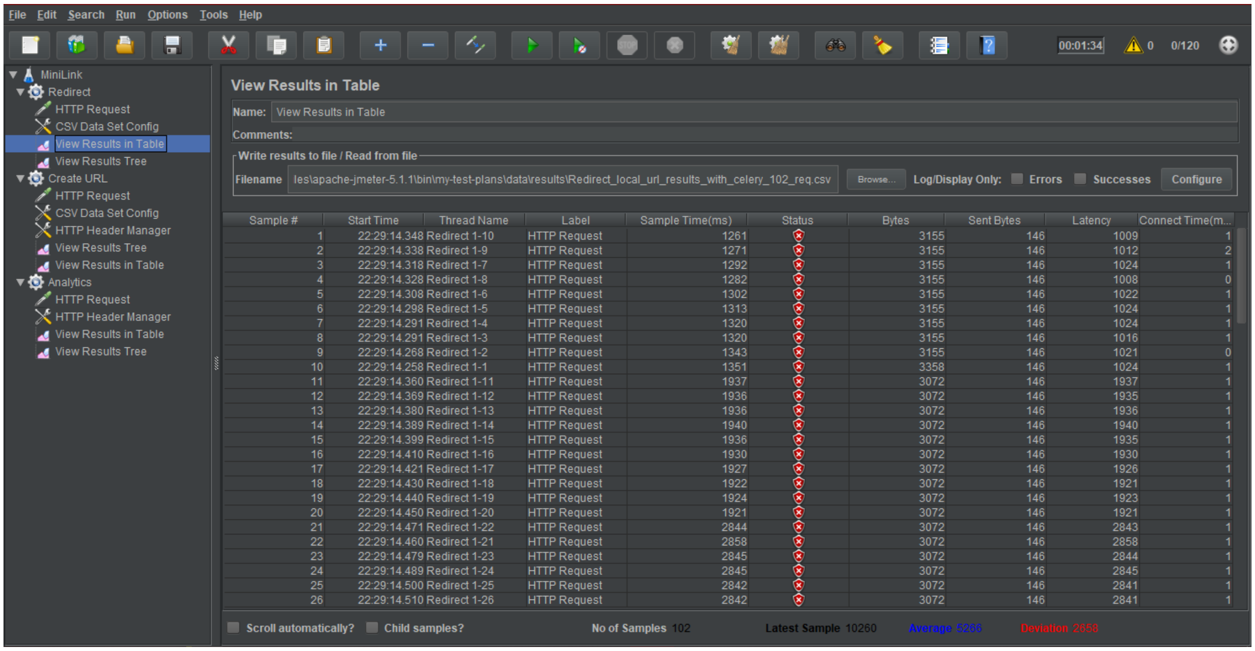
My options were Django-Carrot and Celery. Due to the lack of resources on Django-Carrot and the fact that its whole methods were new to me, I decided to go with Celery which I had experience working with.

Celery made a good improvement in the redirection process. The analytics part also could be implemented with Celery. But there are not many requests for that use case. Therefore, I let it be for now.

Here are the results of all requests using Celery:

Redirect: 1-10 seconds
Create Url: 0.7-4 seconds
Analytics: 0.1-4 seconds

Redirect:



View Results in Table

Name: View Results in Table

Comments:

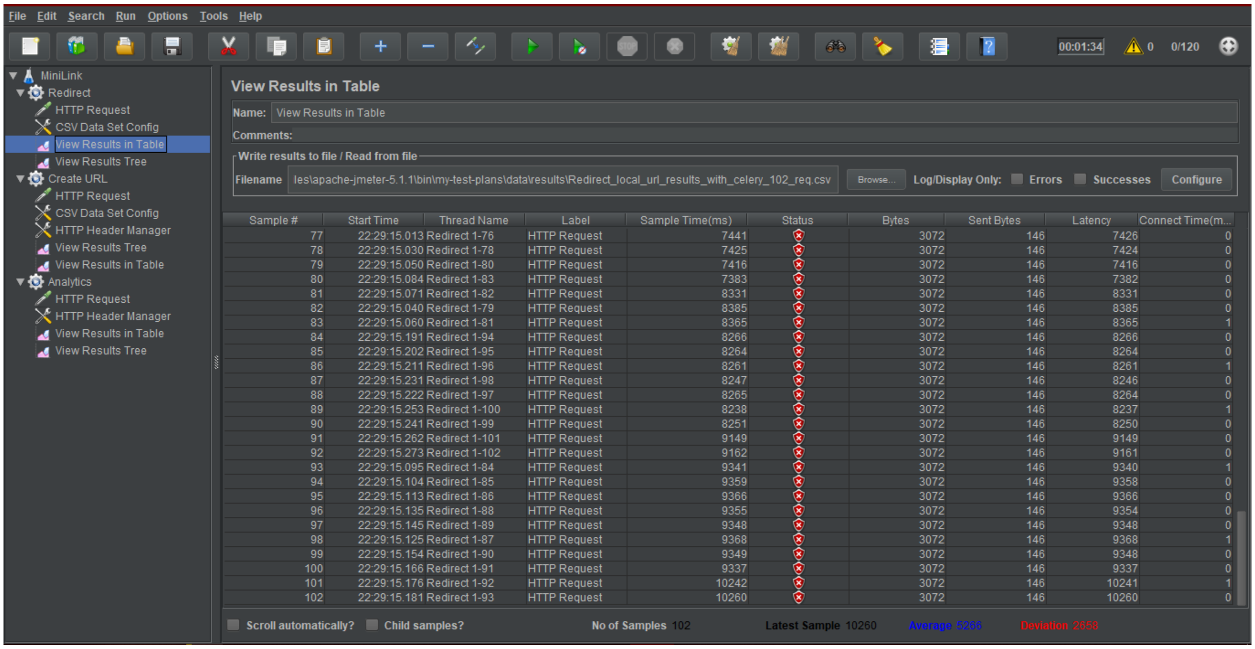
Write results to file / Read from file

Filename les\apache-jmeter-5.1.1\bin\my-test-plans\data\results\Redirect_local_url_results_with_celery_102_req.csv

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(m...
1	22:29:14.348	Redirect 1-10	HTTP Request	1261	☒	3155	146	1009	1
2	22:29:14.338	Redirect 1-9	HTTP Request	1271	☒	3155	146	1012	2
3	22:29:14.318	Redirect 1-7	HTTP Request	1292	☒	3155	146	1024	1
4	22:29:14.328	Redirect 1-8	HTTP Request	1282	☒	3155	146	1008	0
5	22:29:14.308	Redirect 1-6	HTTP Request	1302	☒	3155	146	1022	1
6	22:29:14.298	Redirect 1-5	HTTP Request	1313	☒	3155	146	1024	1
7	22:29:14.291	Redirect 1-4	HTTP Request	1320	☒	3155	146	1024	1
8	22:29:14.291	Redirect 1-3	HTTP Request	1320	☒	3155	146	1016	1
9	22:29:14.268	Redirect 1-2	HTTP Request	1343	☒	3155	146	1021	0
10	22:29:14.258	Redirect 1-1	HTTP Request	1351	☒	3358	146	1024	1
11	22:29:14.360	Redirect 1-11	HTTP Request	1937	☒	3072	146	1937	1
12	22:29:14.369	Redirect 1-12	HTTP Request	1936	☒	3072	146	1935	1
13	22:29:14.380	Redirect 1-13	HTTP Request	1936	☒	3072	146	1936	1
14	22:29:14.389	Redirect 1-14	HTTP Request	1940	☒	3072	146	1940	1
15	22:29:14.399	Redirect 1-15	HTTP Request	1936	☒	3072	146	1935	1
16	22:29:14.410	Redirect 1-16	HTTP Request	1930	☒	3072	146	1930	1
17	22:29:14.421	Redirect 1-17	HTTP Request	1927	☒	3072	146	1926	1
18	22:29:14.430	Redirect 1-18	HTTP Request	1922	☒	3072	146	1921	1
19	22:29:14.440	Redirect 1-19	HTTP Request	1924	☒	3072	146	1923	1
20	22:29:14.450	Redirect 1-20	HTTP Request	1921	☒	3072	146	1921	1
21	22:29:14.471	Redirect 1-22	HTTP Request	2844	☒	3072	146	2843	1
22	22:29:14.460	Redirect 1-21	HTTP Request	2858	☒	3072	146	2858	1
23	22:29:14.479	Redirect 1-23	HTTP Request	2845	☒	3072	146	2844	1
24	22:29:14.489	Redirect 1-24	HTTP Request	2845	☒	3072	146	2845	1
25	22:29:14.500	Redirect 1-25	HTTP Request	2842	☒	3072	146	2841	1
26	22:29:14.510	Redirect 1-26	HTTP Request	2842	☒	3072	146	2841	1

□ Scroll automatically? □ Child samples? No of Samples 102 Latest Sample 10260 Average 5266 Deviation 2658



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

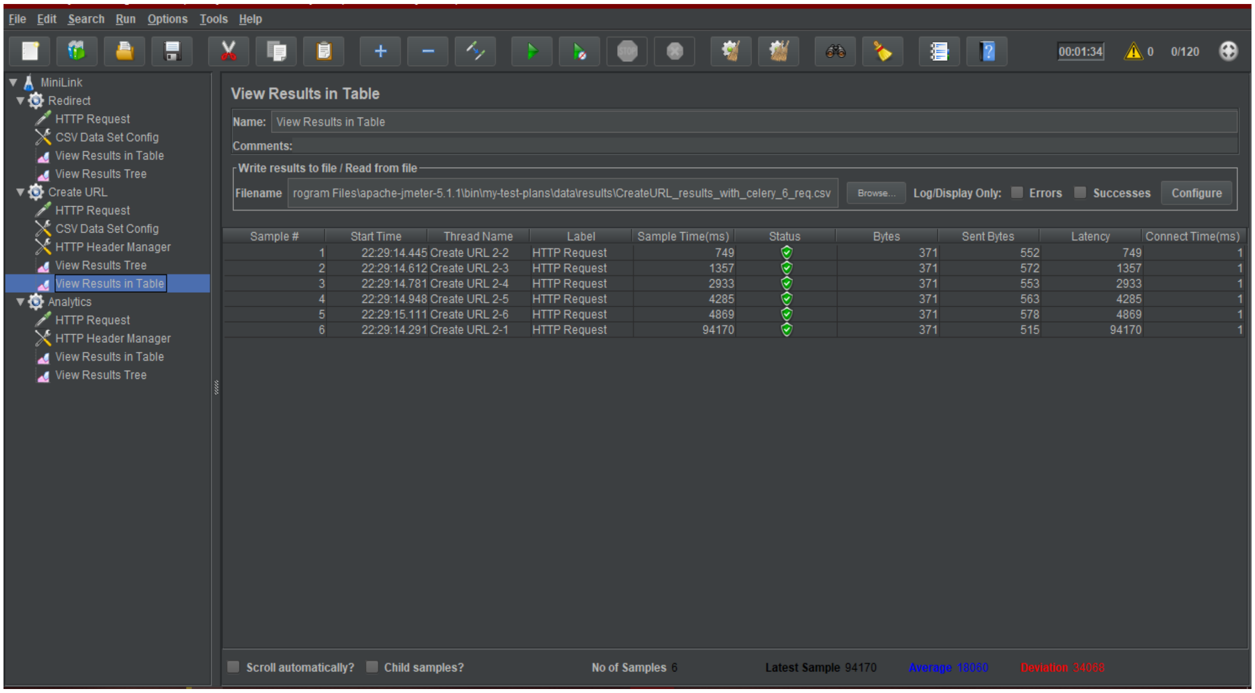
Filename les\apache-jmeter-5.1.1\bin\my-test-plans\data\results\Redirect_local_url_results_with_celery_102_req.csv

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(m...
77	22:29:15.013	Redirect 1-76	HTTP Request	7441	☒	3072	146	7426	0
78	22:29:15.030	Redirect 1-78	HTTP Request	7425	☒	3072	146	7424	0
79	22:29:15.050	Redirect 1-80	HTTP Request	7416	☒	3072	146	7416	0
80	22:29:15.084	Redirect 1-83	HTTP Request	7383	☒	3072	146	7382	0
81	22:29:15.071	Redirect 1-82	HTTP Request	8331	☒	3072	146	8331	0
82	22:29:15.040	Redirect 1-79	HTTP Request	8385	☒	3072	146	8385	0
83	22:29:15.060	Redirect 1-81	HTTP Request	8365	☒	3072	146	8365	1
84	22:29:15.191	Redirect 1-94	HTTP Request	8266	☒	3072	146	8266	0
85	22:29:15.202	Redirect 1-95	HTTP Request	8264	☒	3072	146	8264	0
86	22:29:15.211	Redirect 1-96	HTTP Request	8261	☒	3072	146	8261	1
87	22:29:15.231	Redirect 1-98	HTTP Request	8247	☒	3072	146	8246	0
88	22:29:15.222	Redirect 1-97	HTTP Request	8265	☒	3072	146	8264	0
89	22:29:15.253	Redirect 1-100	HTTP Request	8238	☒	3072	146	8237	1
90	22:29:15.241	Redirect 1-99	HTTP Request	8251	☒	3072	146	8250	0
91	22:29:15.262	Redirect 1-101	HTTP Request	9149	☒	3072	146	9149	0
92	22:29:15.273	Redirect 1-102	HTTP Request	9162	☒	3072	146	9161	0
93	22:29:15.095	Redirect 1-84	HTTP Request	9341	☒	3072	146	9340	1
94	22:29:15.104	Redirect 1-85	HTTP Request	9359	☒	3072	146	9358	0
95	22:29:15.113	Redirect 1-86	HTTP Request	9366	☒	3072	146	9366	0
96	22:29:15.135	Redirect 1-88	HTTP Request	9355	☒	3072	146	9354	0
97	22:29:15.145	Redirect 1-89	HTTP Request	9348	☒	3072	146	9348	0
98	22:29:15.125	Redirect 1-87	HTTP Request	9368	☒	3072	146	9368	1
99	22:29:15.154	Redirect 1-90	HTTP Request	9349	☒	3072	146	9348	0
100	22:29:15.166	Redirect 1-91	HTTP Request	9337	☒	3072	146	9337	0
101	22:29:15.176	Redirect 1-92	HTTP Request	10242	☒	3072	146	10241	1
102	22:29:15.181	Redirect 1-93	HTTP Request	10260	☒	3072	146	10260	0

□ Scroll automatically? □ Child samples? No of Samples 102 Latest Sample 10260 Average 5266 Deviation 2658

CreateUrl:



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

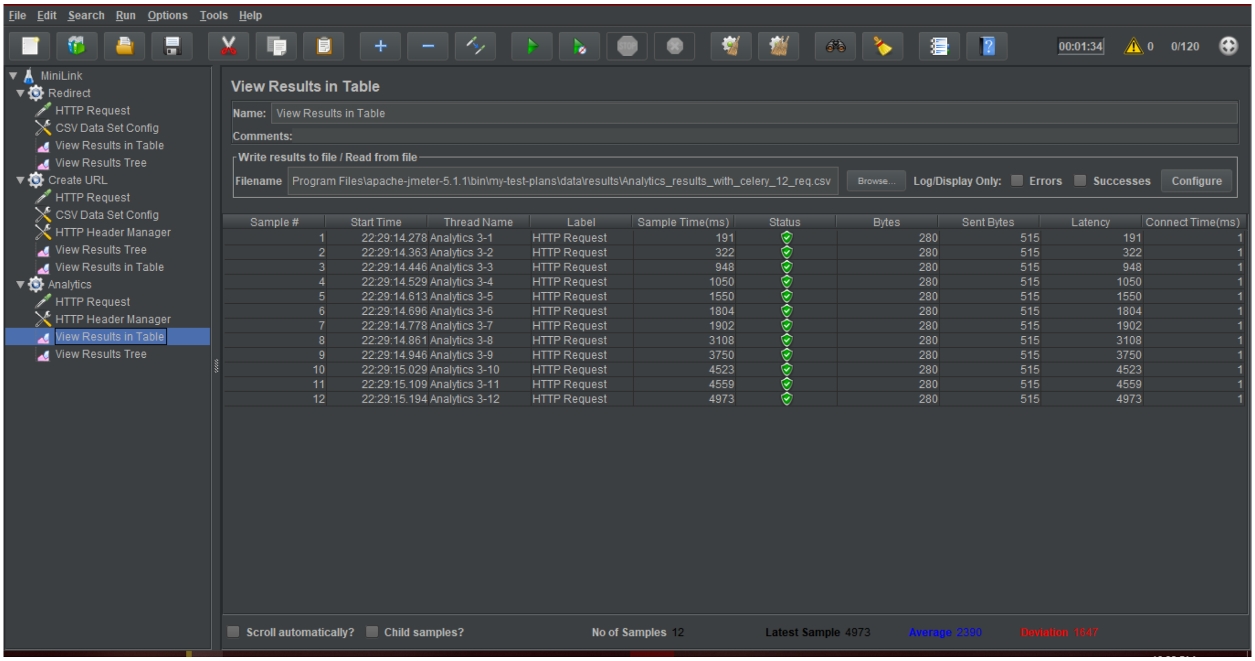
Filename: rogram Files\apache-jmeter-5.1.1\bin\my-test-plans\data\results\CreateURL_results_with_celery_6_req.csv

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	22:29:14.445	Create URL 2-2	HTTP Request	749	✓	371	552	749	1
2	22:29:14.612	Create URL 2-3	HTTP Request	1357	✓	371	572	1357	1
3	22:29:14.781	Create URL 2-4	HTTP Request	2933	✓	371	553	2933	1
4	22:29:14.948	Create URL 2-5	HTTP Request	4285	✓	371	563	4285	1
5	22:29:15.111	Create URL 2-6	HTTP Request	4869	✓	371	578	4869	1
6	22:29:14.291	Create URL 2-1	HTTP Request	94170	✓	371	515	94170	1

□ Scroll automatically? □ Child samples? No of Samples 6 Latest Sample 94170 Average 18060 Deviation 34068

Analytics:



View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Program Files\apache-jmeter-5.1.1\bin\my-test-plans\data\results\Analytics_results_with_celery_12_req.csv

Log/Display Only: □ Errors □ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	22:29:14.278	Analytics 3-1	HTTP Request	191	✓	280	515	191	1
2	22:29:14.363	Analytics 3-2	HTTP Request	322	✓	280	515	322	1
3	22:29:14.446	Analytics 3-3	HTTP Request	948	✓	280	515	948	1
4	22:29:14.529	Analytics 3-4	HTTP Request	1050	✓	280	515	1050	1
5	22:29:14.613	Analytics 3-5	HTTP Request	1550	✓	280	515	1550	1
6	22:29:14.696	Analytics 3-6	HTTP Request	1804	✓	280	515	1804	1
7	22:29:14.778	Analytics 3-7	HTTP Request	1902	✓	280	515	1902	1
8	22:29:14.861	Analytics 3-8	HTTP Request	3108	✓	280	515	3108	1
9	22:29:14.946	Analytics 3-9	HTTP Request	3750	✓	280	515	3750	1
10	22:29:15.029	Analytics 3-10	HTTP Request	4523	✓	280	515	4523	1
11	22:29:15.109	Analytics 3-11	HTTP Request	4559	✓	280	515	4559	1
12	22:29:15.194	Analytics 3-12	HTTP Request	4973	✓	280	515	4973	1

□ Scroll automatically? □ Child samples? No of Samples 12 Latest Sample 4973 Average 2390 Deviation 1647

Note: You can find the Report files in email attachments (CSV format).