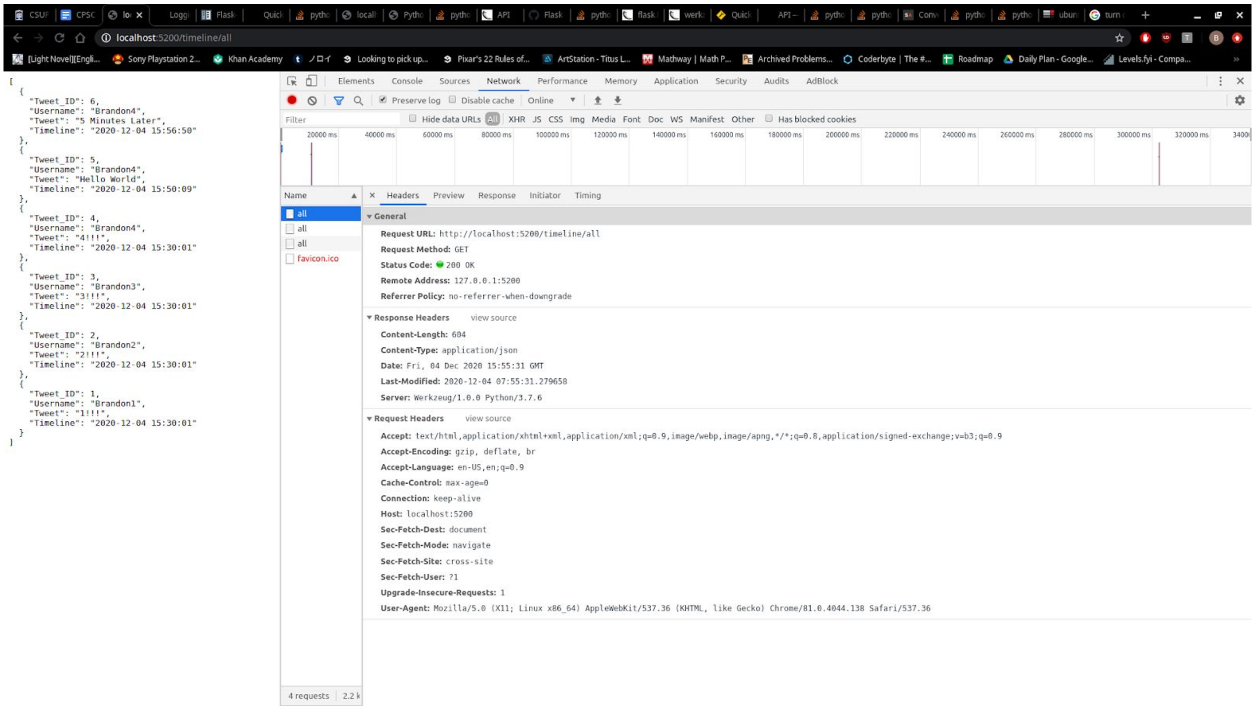
Screenshots

Test HTTP caching using your Browser’s Developer Tools to verify that:

1. The initial call to `getPublicTimeline()` loads JSON from the server



```
[
  {
    "Tweet_ID": 6,
    "Username": "Brandon4",
    "Tweet": "5 Minutes Later",
    "Timeline": "2020-12-04 15:56:50"
  },
  {
    "Tweet_ID": 5,
    "Username": "Brandon4",
    "Tweet": "Hello World",
    "Timeline": "2020-12-04 15:50:09"
  },
  {
    "Tweet_ID": 4,
    "Username": "Brandon4",
    "Tweet": "4!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 3,
    "Username": "Brandon3",
    "Tweet": "3!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 2,
    "Username": "Brandon2",
    "Tweet": "2!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 1,
    "Username": "Brandon1",
    "Tweet": "1!!!",
    "Timeline": "2020-12-04 15:30:01"
  }
]
```

▼ General

Request URL: http://localhost:5200/timeline/all
Request Method: GET
Status Code: ● 200 OK
Remote Address: 127.0.0.1:5200
Referrer Policy: no-referrer-when-downgrade

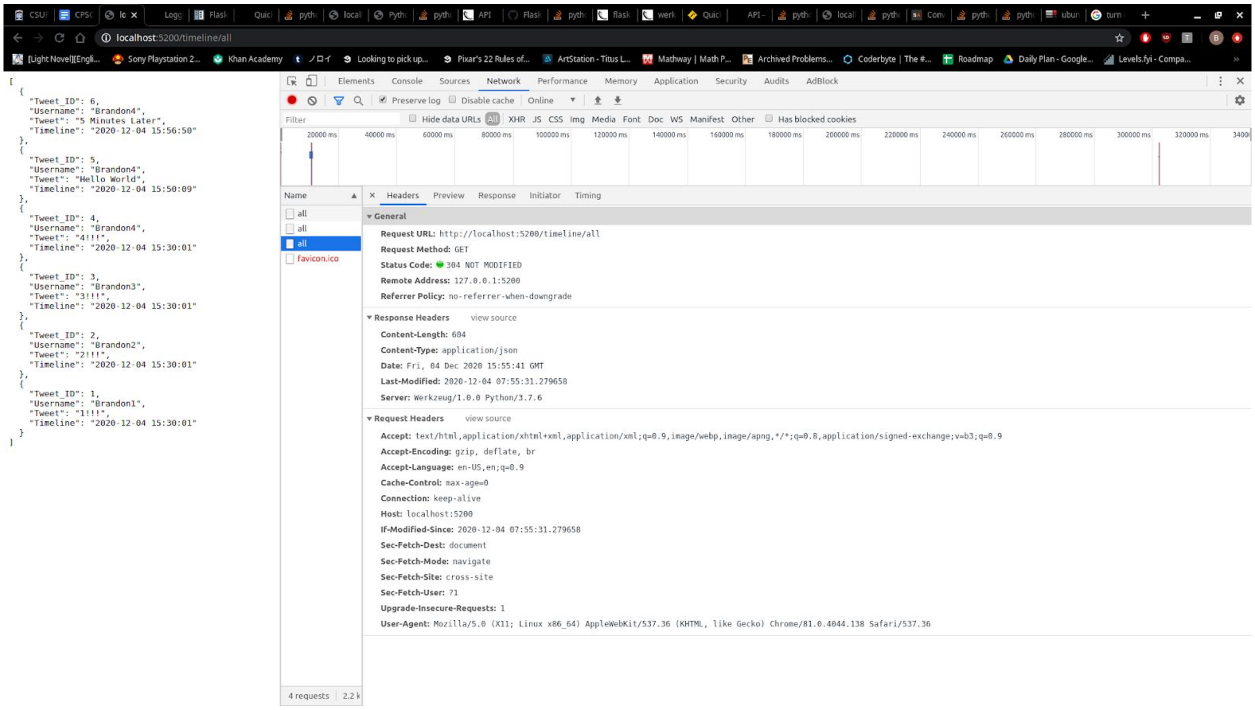
▼ Response Headers view source

Content-Length: 604
Content-Type: application/json
Date: Fri, 04 Dec 2020 15:55:31 GMT
Last-Modified: 2020-12-04 07:55:31.279658
Server: Werkzeug/1.0.0 Python/3.7.6

▼ Request Headers view source

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cache-Control: max-age=0
Connection: keep-alive
Host: localhost:5200
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: cross-site
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.138 Safari/537.36

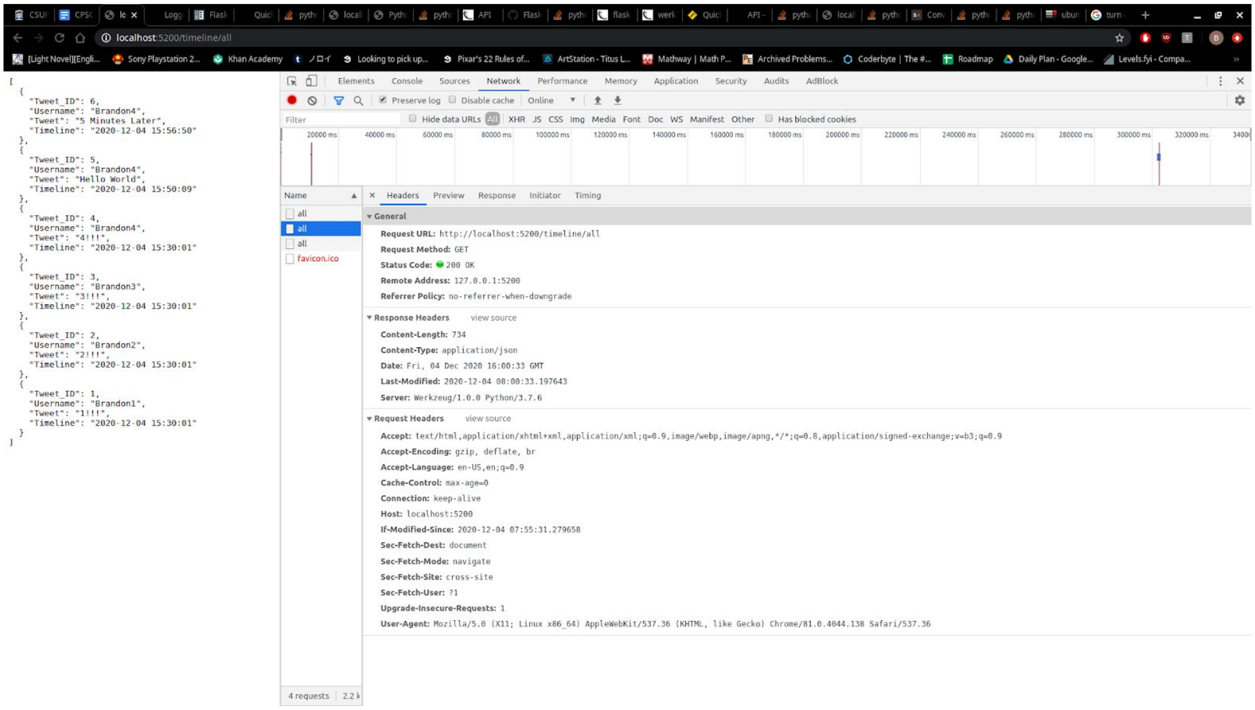
2. Refreshing the page within a 5 minute window results in `HTTP 304 Not Modified` with no payload.



```
[
  {
    "Tweet_ID": 6,
    "Username": "Brandon4",
    "Tweet": "5 Minutes Later",
    "Timeline": "2020-12-04 15:56:50"
  },
  {
    "Tweet_ID": 5,
    "Username": "Brandon4",
    "Tweet": "Hello World",
    "Timeline": "2020-12-04 15:50:09"
  },
  {
    "Tweet_ID": 4,
    "Username": "Brandon4",
    "Tweet": "4!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 3,
    "Username": "Brandon3",
    "Tweet": "3!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 2,
    "Username": "Brandon2",
    "Tweet": "2!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 1,
    "Username": "Brandon1",
    "Tweet": "1!!!",
    "Timeline": "2020-12-04 15:30:01"
  }
]
```

```
▼ General
    Request URL: http://localhost:5200/timeline/all
    Request Method: GET
    Status Code: ● 304 NOT MODIFIED
    Remote Address: 127.0.0.1:5200
    Referrer Policy: no-referrer-when-downgrade
▼ Response Headers      view source
    Content-Length: 604
    Content-Type: application/json
    Date: Fri, 04 Dec 2020 15:55:41 GMT
    Last-Modified: 2020-12-04 07:55:31.279658
    Server: Werkzeug/1.0.0 Python/3.7.6
▼ Request Headers      view source
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    Cache-Control: max-age=0
    Connection: keep-alive
    Host: localhost:5200
    If-Modified-Since: 2020-12-04 07:55:31.279658
    Sec-Fetch-Dest: document
    Sec-Fetch-Mode: navigate
    Sec-Fetch-Site: cross-site
    Sec-Fetch-User: ?1
    Upgrade-Insecure-Requests: 1
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.138 Safari/537.36
```

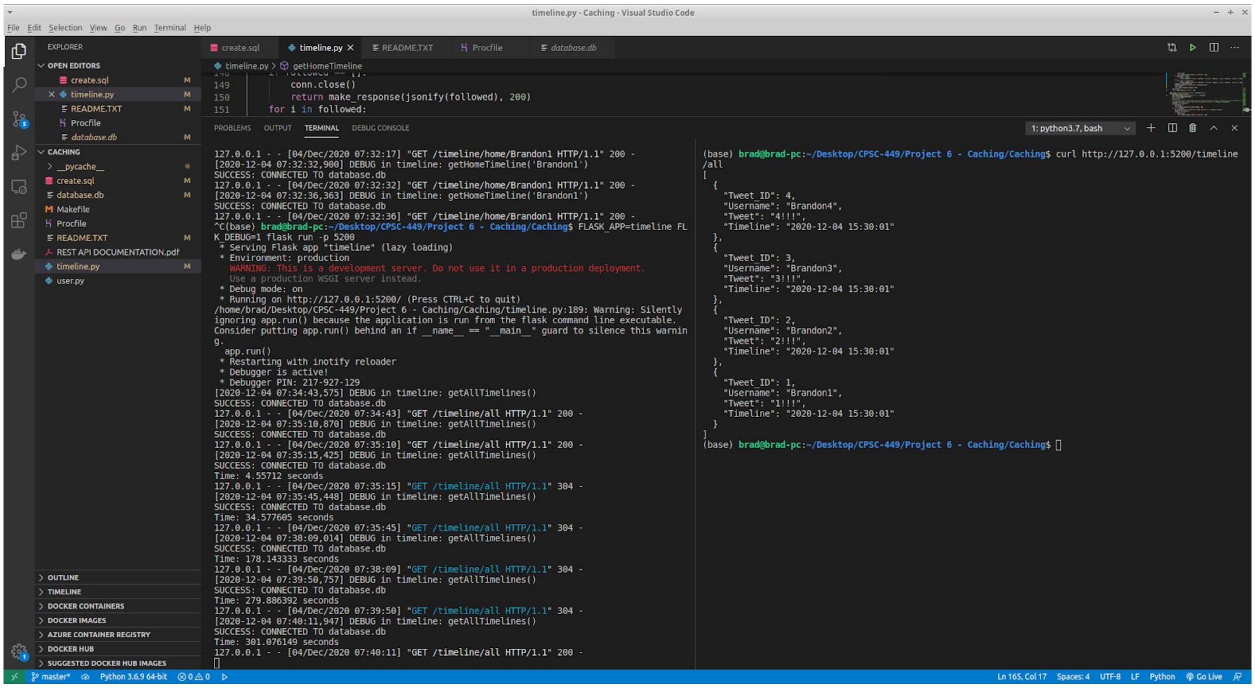
3. Calls outside the 5 minute window reload the JSON for the timeline.



```
[
  {
    "Tweet_ID": 6,
    "Username": "Brandon4",
    "Tweet": "5 Minutes Later",
    "Timeline": "2020-12-04 15:56:50"
  },
  {
    "Tweet_ID": 5,
    "Username": "Brandon4",
    "Tweet": "Hello World",
    "Timeline": "2020-12-04 15:50:09"
  },
  {
    "Tweet_ID": 4,
    "Username": "Brandon4",
    "Tweet": "4!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 3,
    "Username": "Brandon3",
    "Tweet": "3!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 2,
    "Username": "Brandon2",
    "Tweet": "2!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 1,
    "Username": "Brandon1",
    "Tweet": "1!!!",
    "Timeline": "2020-12-04 15:30:01"
  }
]
```

```
▼ General
    Request URL: http://localhost:5200/timeline/all
    Request Method: GET
    Status Code: ● 200 OK
    Remote Address: 127.0.0.1:5200
    Referrer Policy: no-referrer-when-downgrade
▼ Response Headers      view source
    Content-Length: 734
    Content-Type: application/json
    Date: Fri, 04 Dec 2020 16:00:33 GMT
    Last-Modified: 2020-12-04 08:00:33.197643
    Server: Werkzeug/1.0.0 Python/3.7.6
▼ Request Headers      view source
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    Cache-Control: max-age=0
    Connection: keep-alive
    Host: localhost:5200
    If-Modified-Since: 2020-12-04 07:55:31.279658
    Sec-Fetch-Dest: document
    Sec-Fetch-Mode: navigate
    Sec-Fetch-Site: cross-site
    Sec-Fetch-User: ?1
    Upgrade-Insecure-Requests: 1
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.138 Safari/537.36
```

4. Logged Screenshots



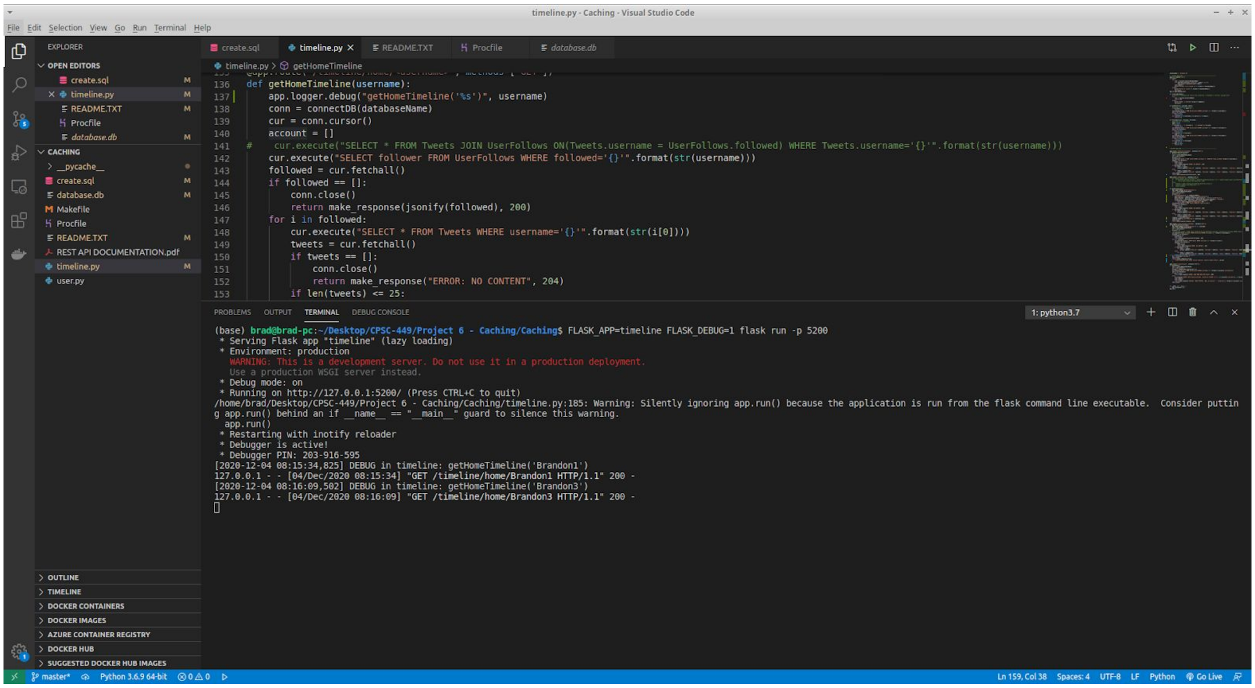
```
            conn.close()
            return make_response(jsonify(followed), 200)
        for i in followed:
```

```
127.0.0.1 - - [04/Dec/2020 07:32:17] "GET /timeline/home/Brandon1 HTTP/1.1" 200 -
[2020-12-04 07:32:32,900] DEBUG in timeline: getHomeTimeline('Brandon1')
SUCCESS: CONNECTED TO database.db
127.0.0.1 - - [04/Dec/2020 07:32:32] "GET /timeline/home/Brandon1 HTTP/1.1" 200 -
[2020-12-04 07:32:36,363] DEBUG in timeline: getHomeTimeline('Brandon1')
SUCCESS: CONNECTED TO database.db
127.0.0.1 - - [04/Dec/2020 07:32:36] "GET /timeline/home/Brandon1 HTTP/1.1" 200 -
^C(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$ FLASK_APP=timeline FL
K_DEBUG=1 flask run -p 5200
 * Serving Flask app "timeline" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5200/ (Press CTRL+C to quit)
/home/brad/Desktop/CPSC-449/Project 6 - Caching/Caching/timeline.py:189: Warning: Silently
ignoring app.run() because the application is run from the flask command line executable.
Consider putting app.run() behind an if __name__ == "__main__" guard to silence this warnin
g.
  app.run()
 * Restarting with inotify reloader
 * Debugger is active!
 * Debugger PIN: 217-927-129
[2020-12-04 07:34:43,575] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
127.0.0.1 - - [04/Dec/2020 07:34:43] "GET /timeline/all HTTP/1.1" 200 -
[2020-12-04 07:35:10,870] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
127.0.0.1 - - [04/Dec/2020 07:35:10] "GET /timeline/all HTTP/1.1" 200 -
[2020-12-04 07:35:15,425] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
Time: 4.55712 seconds
127.0.0.1 - - [04/Dec/2020 07:35:15] "GET /timeline/all HTTP/1.1" 304 -
[2020-12-04 07:35:45,448] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
Time: 34.577605 seconds
127.0.0.1 - - [04/Dec/2020 07:35:45] "GET /timeline/all HTTP/1.1" 304 -
[2020-12-04 07:38:09,014] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
Time: 178.143333 seconds
127.0.0.1 - - [04/Dec/2020 07:38:09] "GET /timeline/all HTTP/1.1" 304 -
[2020-12-04 07:39:50,757] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
Time: 279.886392 seconds
127.0.0.1 - - [04/Dec/2020 07:39:50] "GET /timeline/all HTTP/1.1" 304 -
[2020-12-04 07:40:11,947] DEBUG in timeline: getAllTimelines()
SUCCESS: CONNECTED TO database.db
Time: 301.076149 seconds
127.0.0.1 - - [04/Dec/2020 07:40:11] "GET /timeline/all HTTP/1.1" 200 -
```

```
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$ curl http://127.0.0.1:5200/timeline
/all
[
  {
    "Tweet_ID": 4,
    "Username": "Brandon4",
    "Tweet": "4!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 3,
    "Username": "Brandon3",
    "Tweet": "3!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 2,
    "Username": "Brandon2",
    "Tweet": "2!!!",
    "Timeline": "2020-12-04 15:30:01"
  },
  {
    "Tweet_ID": 1,
    "Username": "Brandon1",
    "Tweet": "1!!!",
    "Timeline": "2020-12-04 15:30:01"
  }
]
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$
```

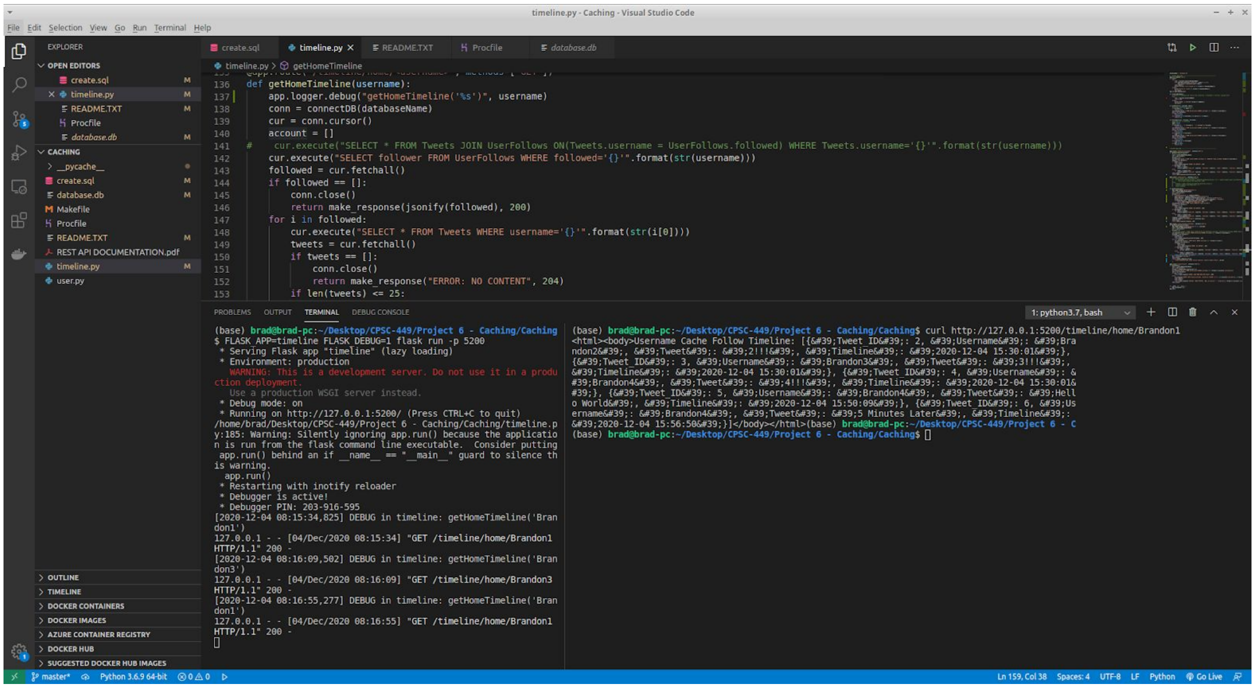
Test object caching by:

1. Using `app.logger.debug()` to determine when a request for a user's timeline is fulfilled from cache and when it results in a call to the original data source.



```python
def getHomeTimeline(username):
    app.logger.debug("getHomeTimeline('%s')", username)
    conn = connectDB(databaseName)
    cur = conn.cursor()
    account = []
#    cur.execute("SELECT * FROM Tweets JOIN UserFollows ON(Tweets.username = UserFollows.followed) WHERE Tweets.username='{}'".format(str(username)))
    cur.execute("SELECT follower FROM UserFollows WHERE followed='{}'".format(str(username)))
    followed = cur.fetchall()
    if followed == []:
        conn.close()
        return make_response(jsonify(followed), 200)
    for i in followed:
        cur.execute("SELECT * FROM Tweets WHERE username='{}'".format(str(i[0])))
        tweets = cur.fetchall()
        if tweets == []:
            conn.close()
            return make_response("ERROR: NO CONTENT", 204)
        if len(tweets) <= 25:
```

```
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$ FLASK_APP=timeline FLASK_DEBUG=1 flask run -p 5200
 * Serving Flask app "timeline" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5200/ (Press CTRL+C to quit)
/home/brad/Desktop/CPSC-449/Project 6 - Caching/Caching/timeline.py:185: Warning: Silently ignoring app.run() because the application is run from the flask command line executable.  Consider puttin
g app.run() behind an if __name__ == "__main__" guard to silence this warning.
  app.run()
 * Restarting with inotify reloader
 * Debugger is active!
 * Debugger PIN: 203-916-595
[2020-12-04 08:15:34,825] DEBUG in timeline: getHomeTimeline('Brandon1')
127.0.0.1 - - [04/Dec/2020 08:15:34] "GET /timeline/home/Brandon1 HTTP/1.1" 200 -
[2020-12-04 08:16:09,502] DEBUG in timeline: getHomeTimeline('Brandon3')
127.0.0.1 - - [04/Dec/2020 08:16:09] "GET /timeline/home/Brandon3 HTTP/1.1" 200 -
```

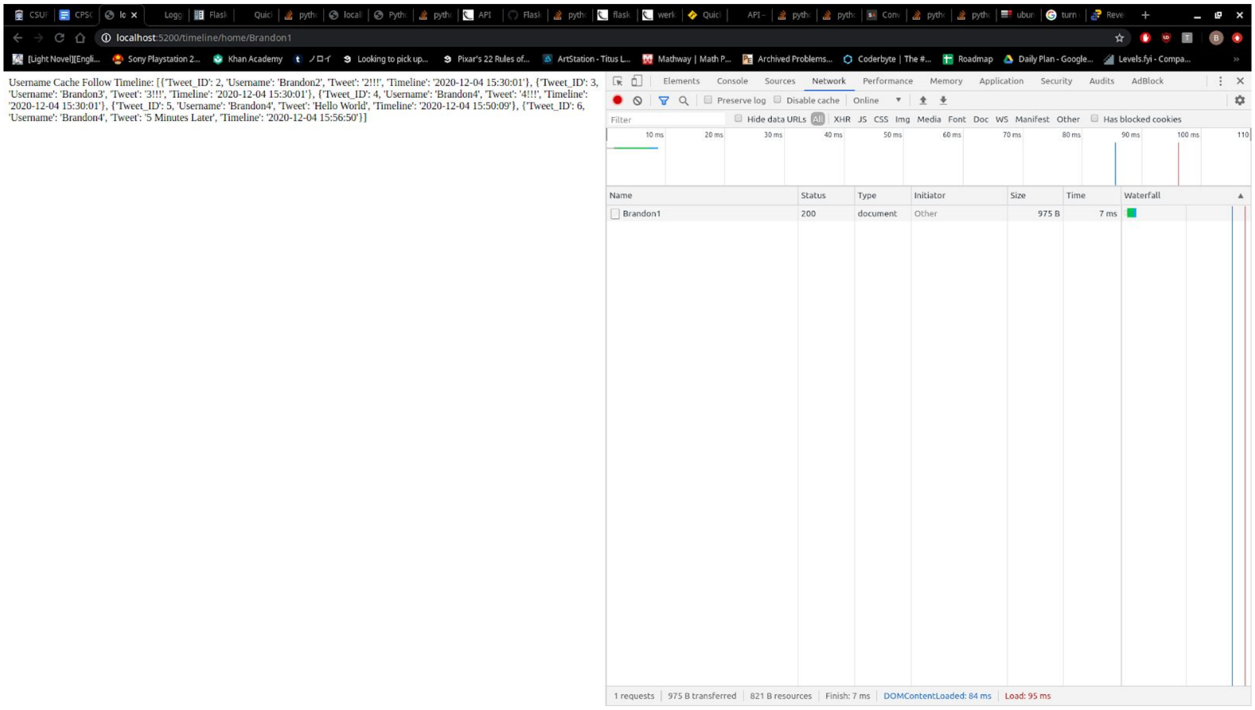


```python
def getHomeTimeline(username):
    app.logger.debug("getHomeTimeline('%s')", username)
    conn = connectDB(databaseName)
    cur = conn.cursor()
    account = []
#    cur.execute("SELECT * FROM Tweets JOIN UserFollows ON(Tweets.username = UserFollows.followed) WHERE Tweets.username='{}'".format(str(username)))
    cur.execute("SELECT follower FROM UserFollows WHERE followed='{}'".format(str(username)))
    followed = cur.fetchall()
    if followed == []:
        conn.close()
        return make_response(jsonify(followed), 200)
    for i in followed:
        cur.execute("SELECT * FROM Tweets WHERE username='{}'".format(str(i[0])))
        tweets = cur.fetchall()
        if tweets == []:
            conn.close()
            return make_response("ERROR: NO CONTENT", 204)
        if len(tweets) <= 25:
```

```
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching
$ FLASK_APP=timeline FLASK_DEBUG=1 flask run -p 5200
 * Serving Flask app "timeline" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a produ
ction deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5200/ (Press CTRL+C to quit)
/home/brad/Desktop/CPSC-449/Project 6 - Caching/Caching/timeline.p
y:185: Warning: Silently ignoring app.run() because the applicatio
n is run from the flask command line executable.  Consider putting
 app.run() behind an if __name__ == "__main__" guard to silence th
is warning.
  app.run()
 * Restarting with inotify reloader
 * Debugger is active!
 * Debugger PIN: 203-916-595
[2020-12-04 08:15:34,825] DEBUG in timeline: getHomeTimeline('Bran
don1')
127.0.0.1 - - [04/Dec/2020 08:15:34] "GET /timeline/home/Brandon1
HTTP/1.1" 200 -
[2020-12-04 08:16:09,502] DEBUG in timeline: getHomeTimeline('Bran
don3')
127.0.0.1 - - [04/Dec/2020 08:16:09] "GET /timeline/home/Brandon3
HTTP/1.1" 200 -
[2020-12-04 08:16:55,277] DEBUG in timeline: getHomeTimeline('Bran
don1')
127.0.0.1 - - [04/Dec/2020 08:16:55] "GET /timeline/home/Brandon1
HTTP/1.1" 200 -
```

```
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$ curl http://127.0.0.1:5200/timeline/home/Brandon1
<html><body>Username Cache Follow Timeline: [{&#39;Tweet_ID&#39;: 2, &#39;Username&#39;: &#39;Bra
ndon2&#39;, &#39;Tweet&#39;: &#39;2!!!&#39;, &#39;Timeline&#39;: &#39;2020-12-04 15:30:01&#39;},
{&#39;Tweet_ID&#39;: 3, &#39;Username&#39;: &#39;Brandon3&#39;, &#39;Tweet&#39;: &#39;3!!!&#39;,
&#39;Timeline&#39;: &#39;2020-12-04 15:30:01&#39;}, {&#39;Tweet_ID&#39;: 4, &#39;Username&#39;: &
#39;Brandon4&#39;, &#39;Tweet&#39;: &#39;4!!!&#39;, &#39;Timeline&#39;: &#39;2020-12-04 15:30:01&
#39;}, {&#39;Tweet_ID&#39;: 5, &#39;Username&#39;: &#39;Brandon4&#39;, &#39;Tweet&#39;: &#39;Hell
o World&#39;, &#39;Timeline&#39;: &#39;2020-12-04 15:50:09&#39;}, {&#39;Tweet_ID&#39;: 6, &#39;Us
ername&#39;: &#39;Brandon4&#39;, &#39;Tweet&#39;: &#39;5 Minutes Later&#39;, &#39;Timeline&#39;:
&#39;2020-12-04 15:56:50&#39;}]</body></html>(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - C
(base) brad@brad-pc:~/Desktop/CPSC-449/Project 6 - Caching/Caching$
```

2. Calling `getHomeTimeline(username)` for a user and verifying its contents.



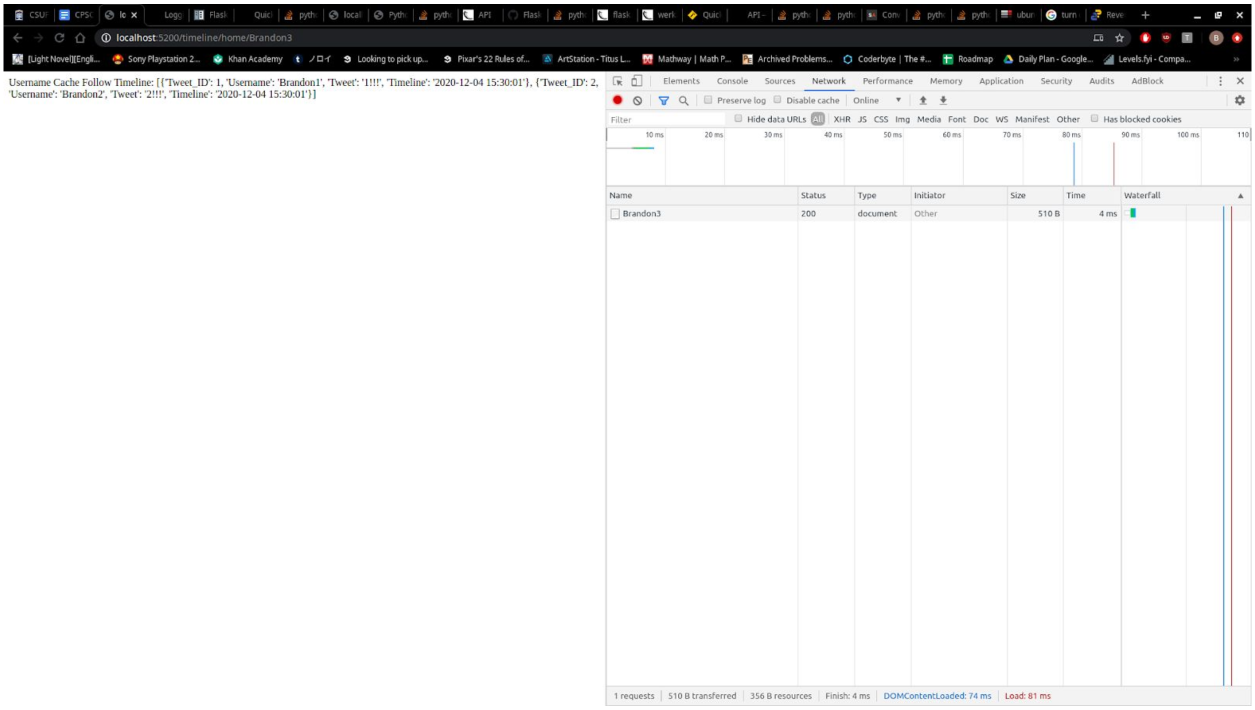
localhost:5200/timeline/home/Brandon1

```
Username Cache Follow Timeline: [{'Tweet_ID': 2, 'Username': 'Brandon2', 'Tweet': '2!!!', 'Timeline': '2020-12-04 15:30:01'}, {'Tweet_ID': 3, 'Username': 'Brandon3', 'Tweet': '3!!!', 'Timeline': '2020-12-04 15:30:01'}, {'Tweet_ID': 4, 'Username': 'Brandon4', 'Tweet': '4!!!', 'Timeline': '2020-12-04 15:30:01'}, {'Tweet_ID': 5, 'Username': 'Brandon4', 'Tweet': 'Hello World', 'Timeline': '2020-12-04 15:50:09'}, {'Tweet_ID': 6, 'Username': 'Brandon4', 'Tweet': '5 Minutes Later', 'Timeline': '2020-12-04 15:56:50'}]
```

Name	Status	Type	Initiator	Size	Time
Brandon1	200	document	Other	975 B	7 ms

1 requests | 975 B transferred | 821 B resources | Finish: 7 ms | DOMContentLoaded: 84 ms | Load: 95 ms

3. Calling `getHomeTimeline(username)` for a different user who follows some of the same users and verifying that the requests for those users' timelines are fulfilled from cache.



localhost:5200/timeline/home/Brandon3

```
Username Cache Follow Timeline: [{'Tweet_ID': 1, 'Username': 'Brandon1', 'Tweet': '1!!!', 'Timeline': '2020-12-04 15:30:01'}, {'Tweet_ID': 2, 'Username': 'Brandon2', 'Tweet': '2!!!', 'Timeline': '2020-12-04 15:30:01'}]
```

Name	Status	Type	Initiator	Size	Time
Brandon3	200	document	Other	510 B	4 ms

1 requests | 510 B transferred | 356 B resources | Finish: 4 ms | DOMContentLoaded: 74 ms | Load: 81 ms