```
1  //history of timestamps
2  var timestampList = [];
3  //accumulate the average of all timestamps in the session
4  var averageGapBetweenTimestamps = 0;
5
6  exports.handler = async (event) => {
7
8      //get current timestamp from date object (milliseconds)
9      const currentTime = new Date();
10     const currentTimestamp = currentTime.getTime();
11
12      //read parameters from url
13     if(event.queryStringParameters != null)
14     {
15         var cmd = event.queryStringParameters.cmd;
16         //if command is to reset, clear the history of time stamps and reset the
   average to begin at 0
17         if(cmd === "RESET")
18         {
19             timestampList = [];
20             averageGapBetweenTimestamps = 0;
21         }
22     }
23
24     //if this is the first timestamp being added, add it to the timestamp list and
   output the timestamp
25     if(timestampList.length === 0)
26     {
27         timestampList.push(currentTimestamp);
28         const response = {
29             statusCode: 200,
30             body: JSON.stringify({ThisInvocation: currentTime.toLocaleString()})
31         }
32
33         return response;
34     }
35
36     //if it is not the first timestamp being added, calculate the time since last
   timestamp and average
37     var TotalInvocationsOnThisContainer = timestampList.length;
38     var TimeSinceLast = currentTimestamp -
   timestampList[TotalInvocationsOnThisContainer-1];
39     averageGapBetweenTimestamps = ((averageGapBetweenTimestamps *
   (TotalInvocationsOnThisContainer-1)+ TimeSinceLast)
   /(TotalInvocationsOnThisContainer++);
40
41     //add the timestamp to the history of timestamps
42     timestampList.push(currentTimestamp);
43
44     //body of the response to return is in the form of a JSON object
45     const responseBody = {
46         ThisInvocation: currentTime.toLocaleString(),
   //convert timestamp to string
47         TimeSinceLast : Math.floor(TimeSinceLast/1000),
48         TotalInvocationsOnThisContainer : TotalInvocationsOnThisContainer,
49         AverageGapBetweenInvocations : Math.floor(averageGapBetweenTimestamps/1000)
50     }
51
52     const response = {
53             statusCode: 200,
```

```
54              body: JSON.stringify(responseBody)
55          }
56
57          return response;
58  }
59
```