Performance

Memi Lavi
www.memilavi.com

Google: 53% of mobile users abandon sites that take over 3 seconds to load

Source: https://www.marketingdive.com/news/google-53-of-mobile-users-abandon-sites-that-take-over-3-seconds-to-load/426070/

Performance

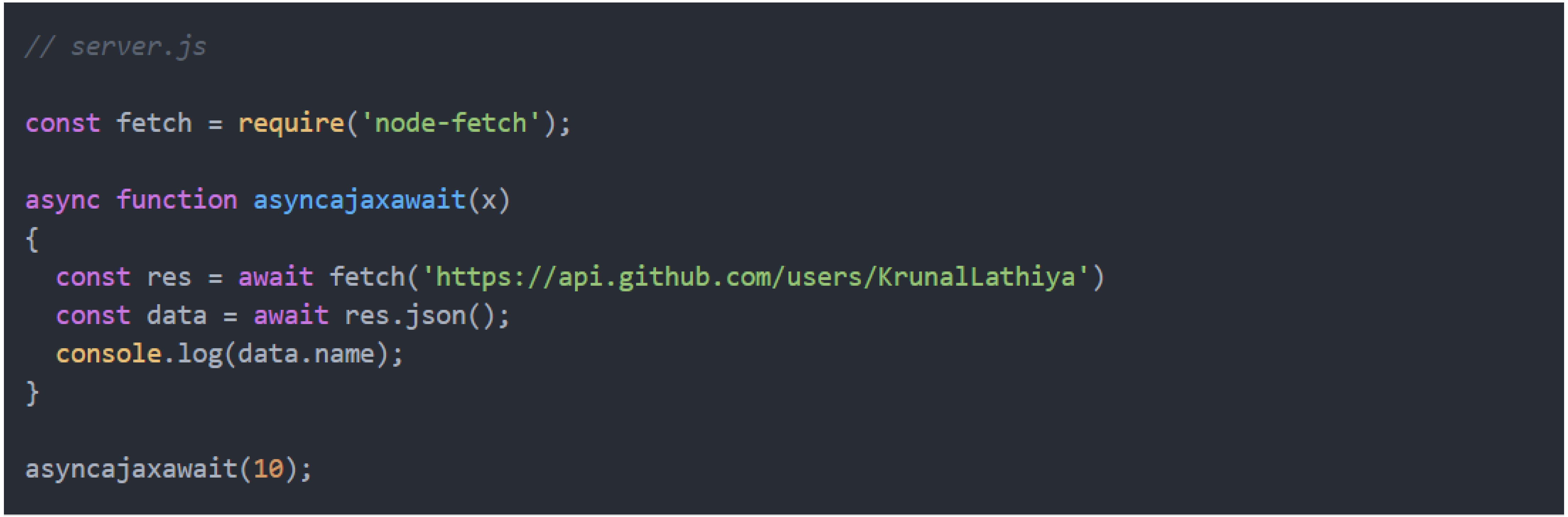
- Your API must be fast
- Performance is not related to the API, but to the underlying code

Async Operations

- Don’t wait for long operations to complete
- Relevant mainly for IO operations (files, database access, networking, etc.)
- Almost every platform supports this concept

Async Operations

async / await in node.js



```
// server.js

const fetch = require('node-fetch');

async function asyncajaxawait(x)
{
  const res = await fetch('https://api.github.com/users/KrunalLathiya')
  const data = await res.json();
  console.log(data.name);
}

asyncajaxawait(10);
```

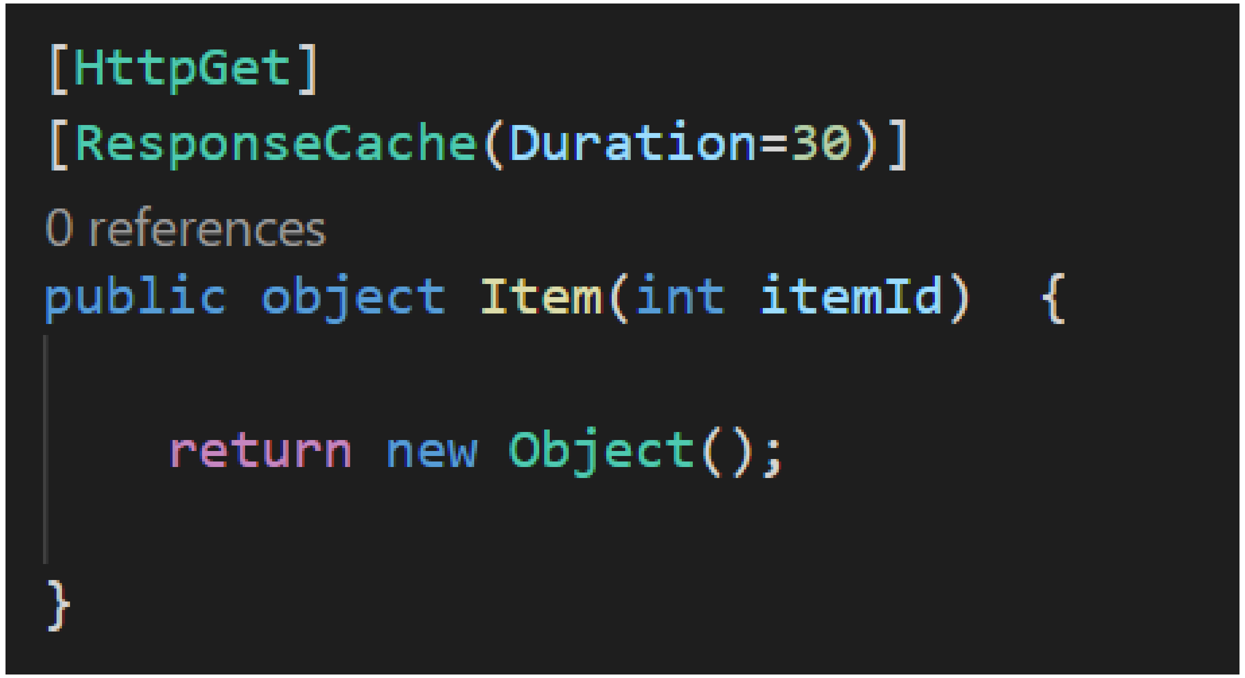
Source: https://appdividend.com/2018/08/28/node-async-await-example-tutorial/

Caching

- Store frequently accessed data close to the API
  - Usually in-Memory
- Set expiration and invalidation

Caching

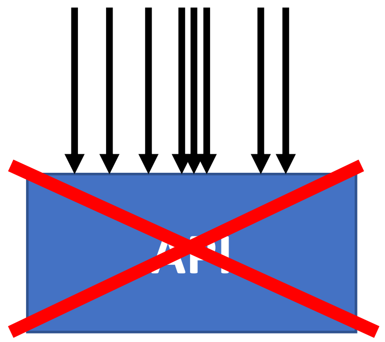
Caching in ASP.NET Core



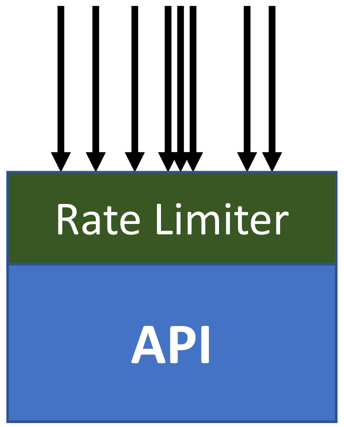
```
[HttpGet]
[ResponseCache(Duration=30)]
0 references
public object Item(int itemId)  {

    return new Object();

}
```

Rate Limit

- Limit the maximum concurrent requests the API handles

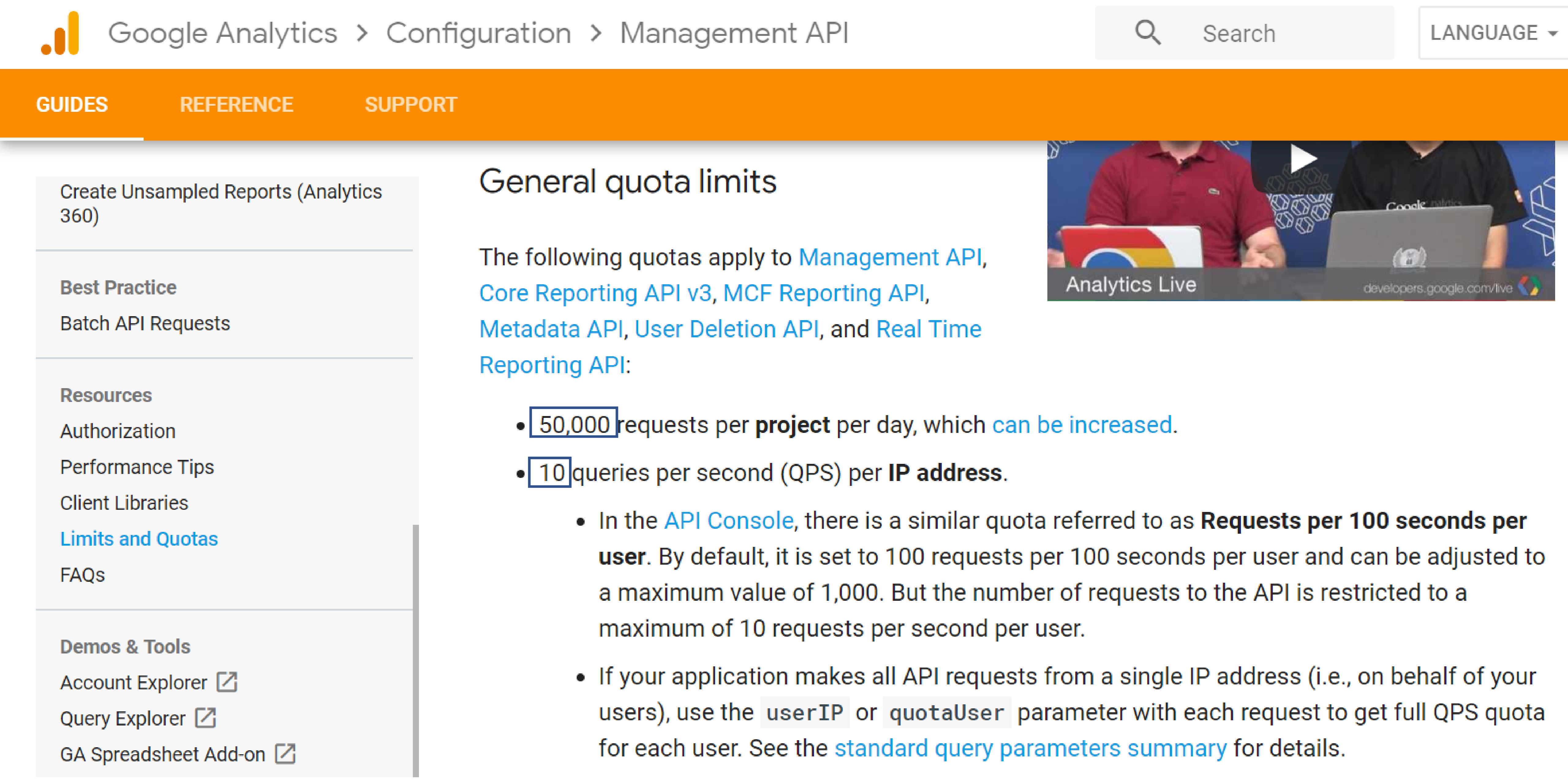


Rate Limit



Quota

- Limits the number of requests a specific client can make
- Minimizes risk of heavy load



Google Analytics > Configuration > Management API

Search

LANGUAGE

GUIDES REFERENCE SUPPORT

Create Unsampled Reports (Analytics 360)

Best Practice

Batch API Requests

Resources

Authorization

Performance Tips

Client Libraries

Limits and Quotas

FAQs

Demos & Tools

Account Explorer

Query Explorer

GA Spreadsheet Add-on

General quota limits

The following quotas apply to Management API, Core Reporting API v3, MCF Reporting API, Metadata API, User Deletion API, and Real Time Reporting API:

- 50,000 requests per **project** per day, which can be increased.
- 10 queries per second (QPS) per **IP address**.
  - In the API Console, there is a similar quota referred to as **Requests per 100 seconds per user**. By default, it is set to 100 requests per 100 seconds per user and can be adjusted to a maximum value of 1,000. But the number of requests to the API is restricted to a maximum of 10 requests per second per user.
  - If your application makes all API requests from a single IP address (i.e., on behalf of your users), use the `userIP` or `quotaUser` parameter with each request to get full QPS quota for each user. See the standard query parameters summary for details.