

Promsie

RWTH Aachen University

1st Papop Lekhapanyaporn 2nd Tobias Piatek 3rd Tobias Waerder

Abstract

In Javascript there are two ways to handle asynchronous code. The first way is to use callbacks. The second way is to use promises. This paper will explain the concept of promises and how to use them. The paper will also compare the two ways to handle asynchronous code.

I. EVENTLOOP

In Javascript, a single threaded programming language, It is possible to handle thousands of operation symaltaniously by using eventloop. Eventloop is created to fix the single threaded problem that program can only run one function at a time. In practice, the program will be unusable without event loop.

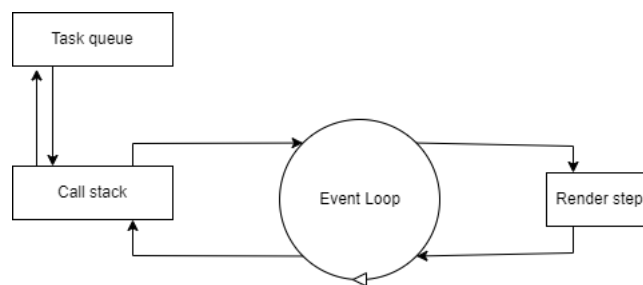


Figure 1. eventloop

in figure 1 the eventloop will constantly check the task queue. If there is a task in the task queue, the first task in the queue will be removed and put into the call stack. The eventloop will then take a detour to the call stack and execute the task that was in the callstack. After a certain amount of time has passed the event loop will then go through the render step which contain styles generation, layout positioning , and pixel data generation that will be displayed on the browser.

II. CALLBACKS

III. PROMISES

Promise is a object use to represent a task that will eventually be either be pending, fulfilled or rejected which will be handle by *then* or *catch* method function.

A. *then*

B. *catch*

IV. ASYNCHRONOUS METHODS

A. *async*

B. *await*

V. CONCLUSION

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer

tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

REFERENCES

- [1] Author, Title *Bookname*, Band and publication number, Page, Year.
- [2] Author, Title *Conference Name*, Page, Year
- [3] Author, *Title*, Publisher, Year