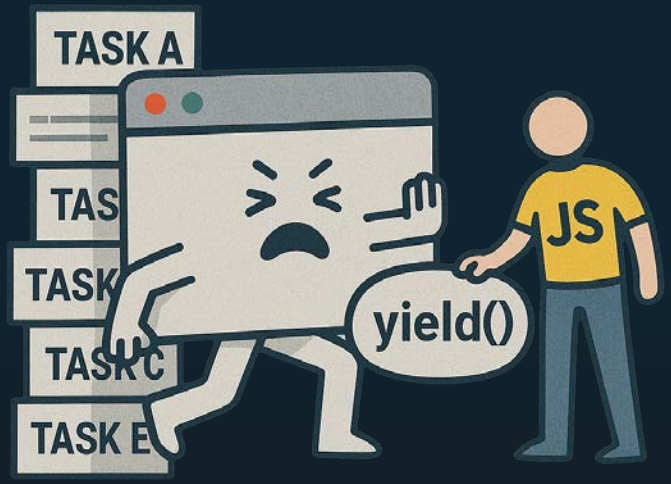
Let Your Browser Take a Breather



OLEKSANDR TKACHENKO

ABOUT ME:

- Frontend developer with 6+ years of experience.
- Author of technical and scientific articles.
- Judge and mentor at international hackathons.
- Speaker at global conferences.
- Open-source contributor.
- Creator of the "Skeleton Mammoth" open-source CSS library.

What is scheduler.yield()?

TERMINOLOGY

Main Thread

This is the central place where the browser does most of its work. It handles rendering, layout, and runs most of your JavaScript code.

Long task

This is any JavaScript task that keeps the **Main Thread** busy for too long – usually more than 50 milliseconds. When that happens, the page can freeze or feel unresponsive.

Blocking task

Is a synchronous operation on the **Main Thread** that prevents the browser from processing other important things, like responding to clicks or updating the UI. Usually, long tasks are blocking tasks.

TERMINOLOGY

Main Thread

This is the central place where the browser does most of its work. It handles rendering, layout, and runs most of your JavaScript code.

Long task

This is any JavaScript task that keeps the **Main Thread** busy for too long – usually more than 50 milliseconds. When that happens, the page can freeze or feel unresponsive.

Blocking task

Is a synchronous operation on the **Main Thread** that prevents the browser from processing other important things, like responding to clicks or updating the UI. Usually, long tasks are blocking tasks.

TERMINOLOGY

Main Thread

This is the central place where the browser does most of its work. It handles rendering, layout, and runs most of your JavaScript code.

Long task

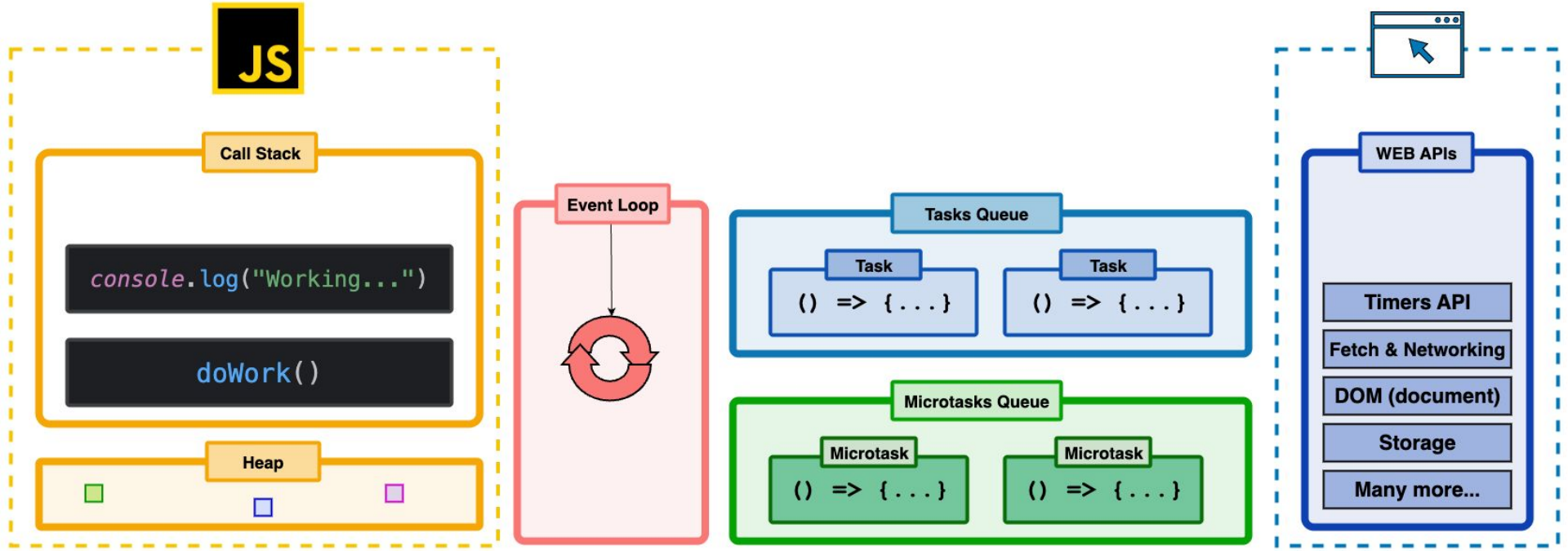
This is any JavaScript task that keeps the **Main Thread** busy for too long – usually more than 50 milliseconds. When that happens, the page can freeze or feel unresponsive.

Blocking task

Is a synchronous operation on the **Main Thread** that prevents the browser from processing other important things, like responding to clicks or updating the UI. Usually, long tasks are blocking tasks.

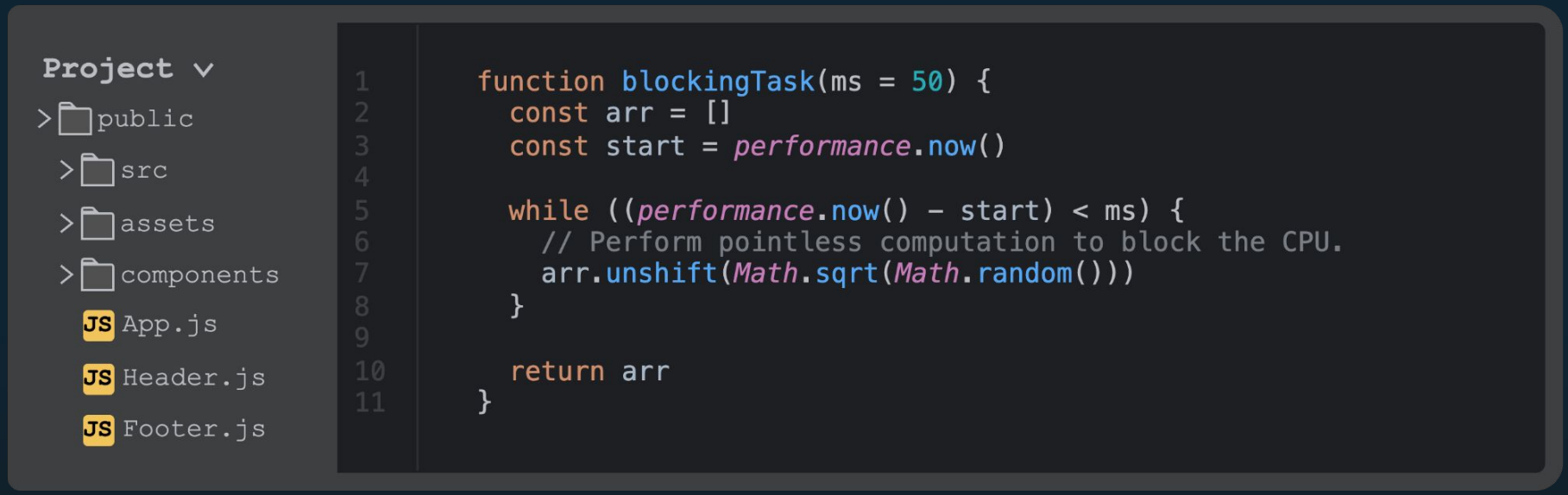
The Problem

Task Processing in the Browser

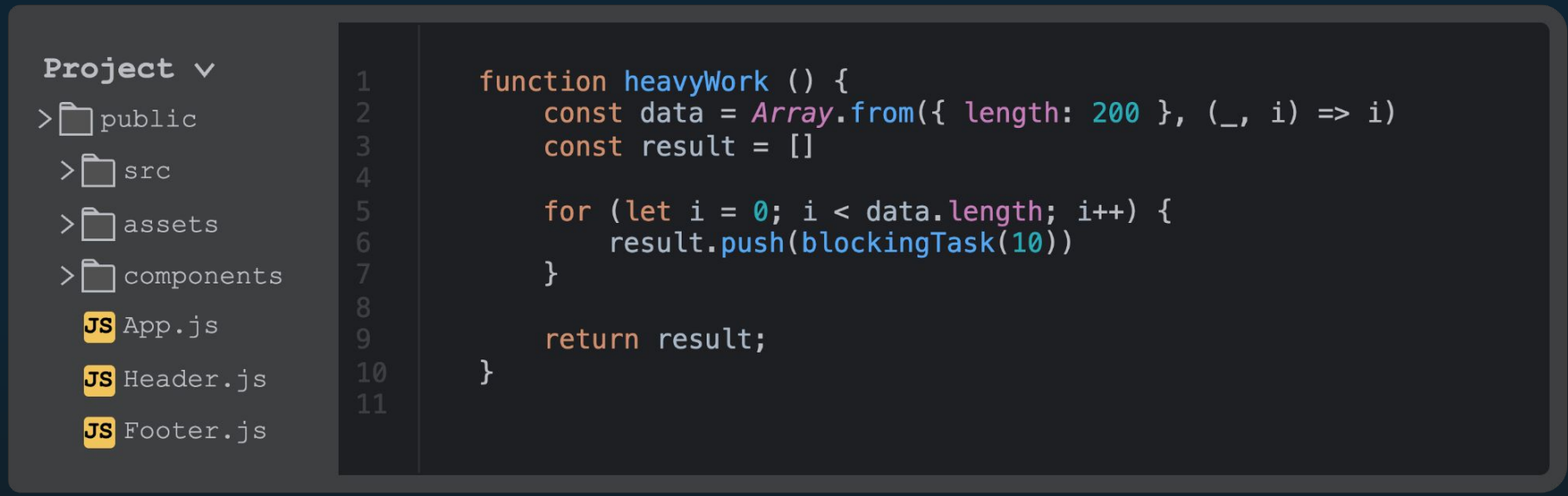


The Problem Description

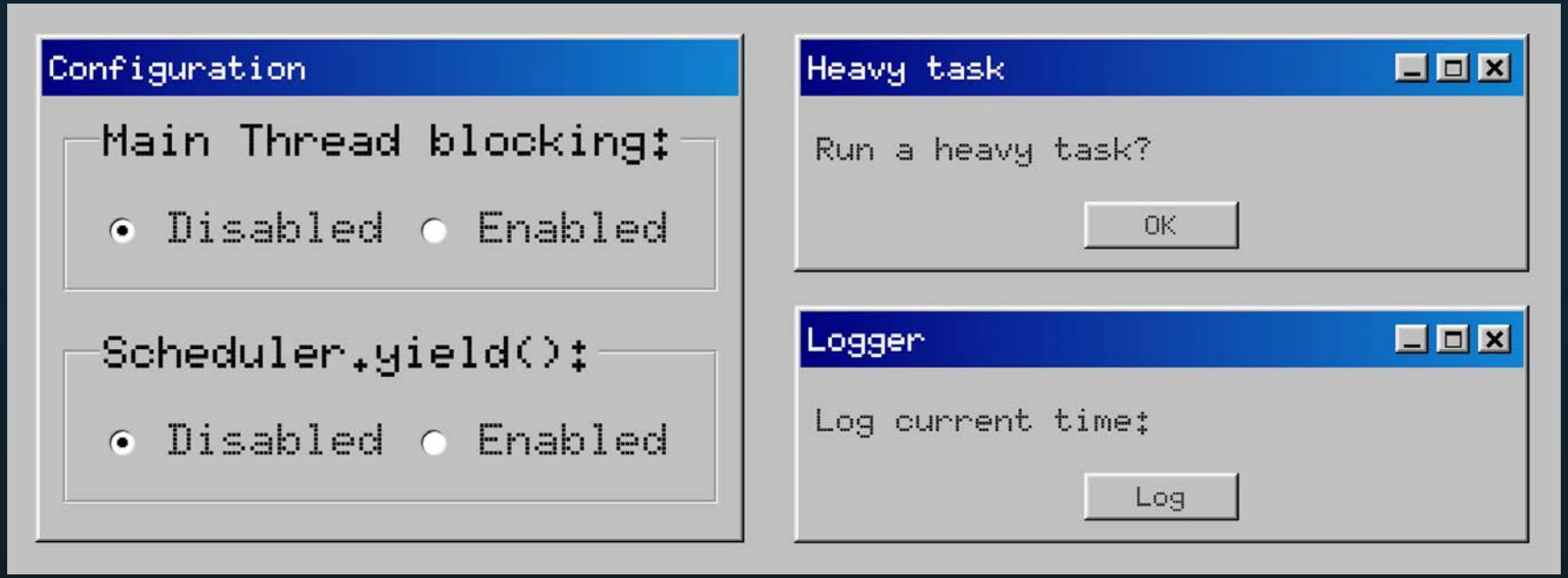
blockingTask



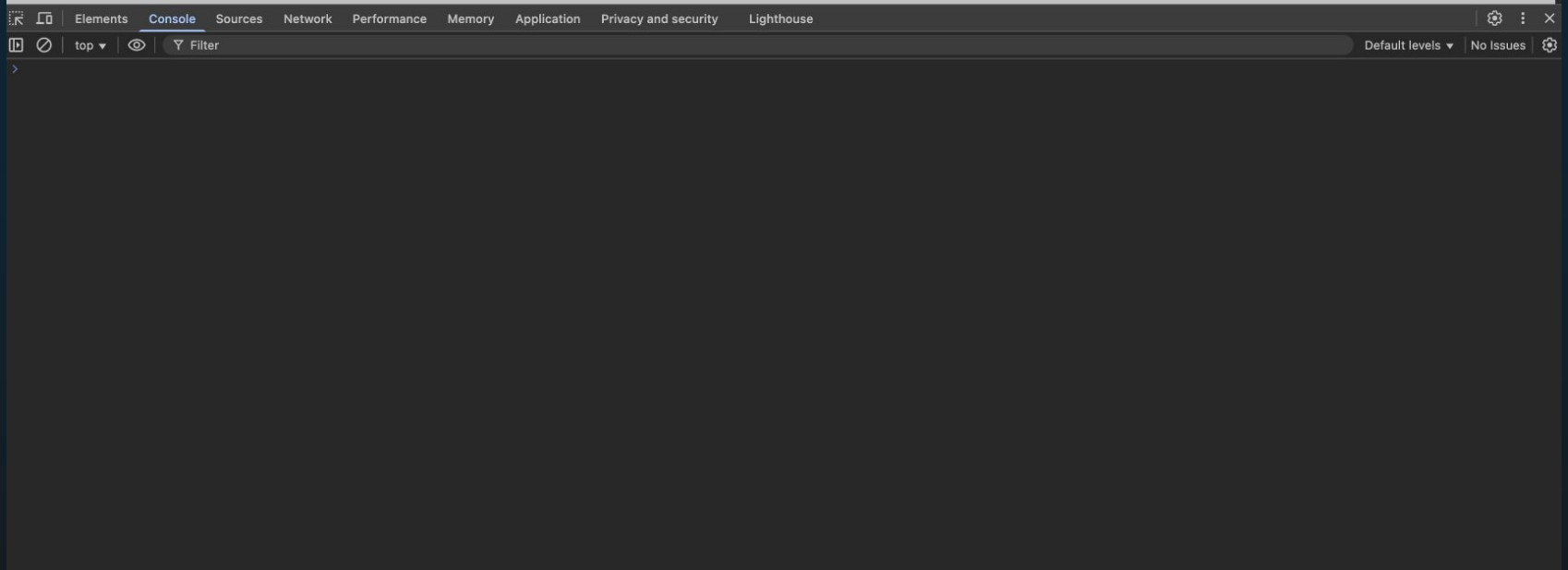
heavyWork


Project
public
src
assets
components
App.js
Header.js
Footer.js
function heavyWork () {
const data = Array.from({ length: 200 }, (_, i) => i)
const result = []
for (let i = 0; i < data.length; i++) {
result.push(blockingTask(10))
}
return result;
}

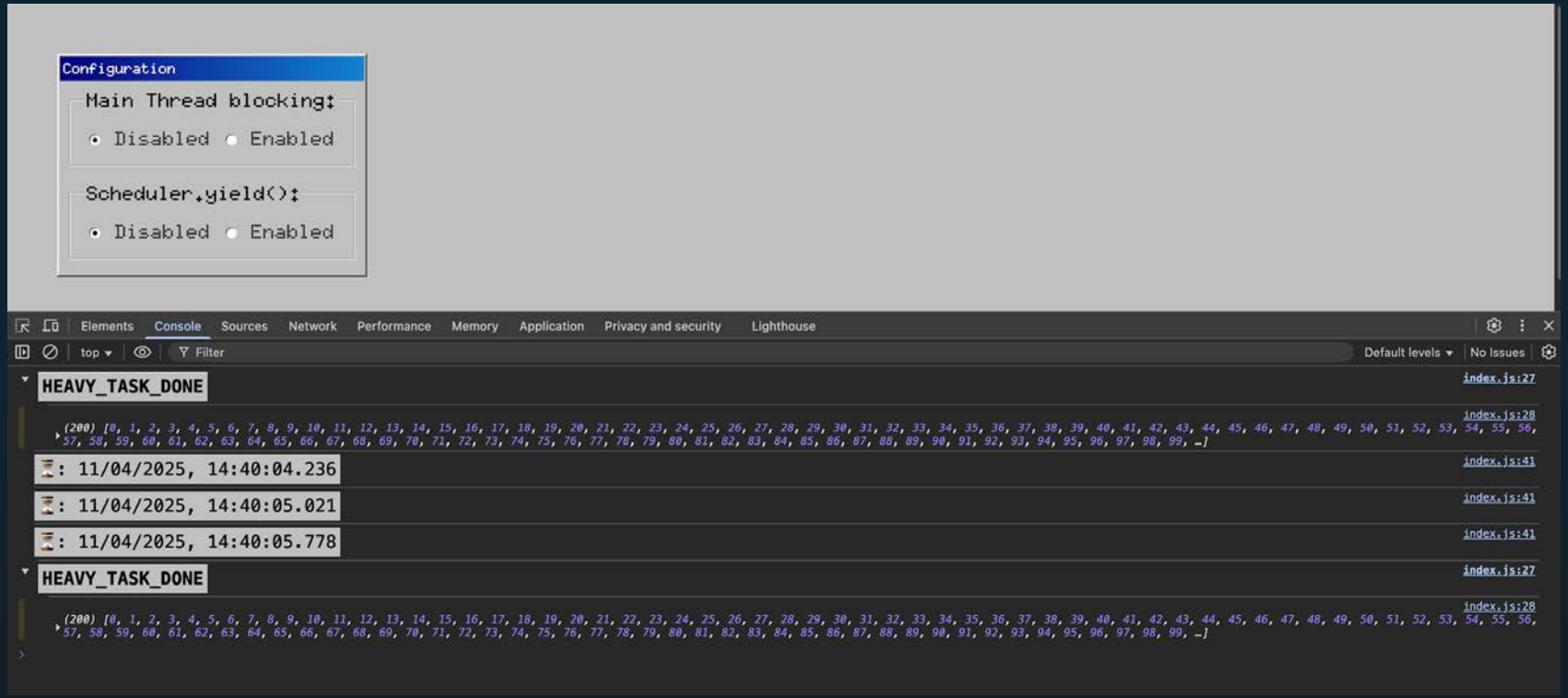
The Problem Demonstration



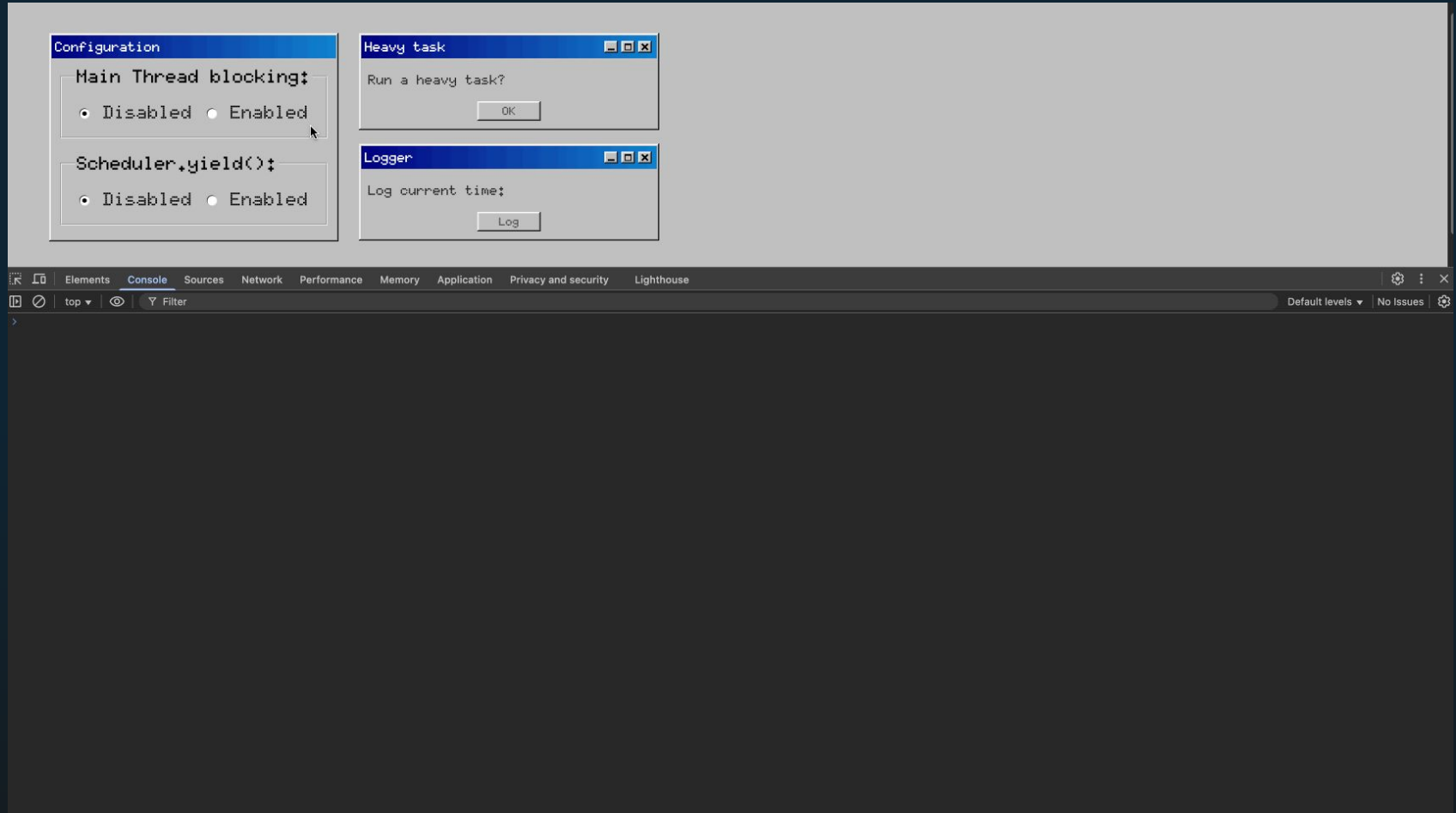
The Problem Demonstration



The Problem Demonstration

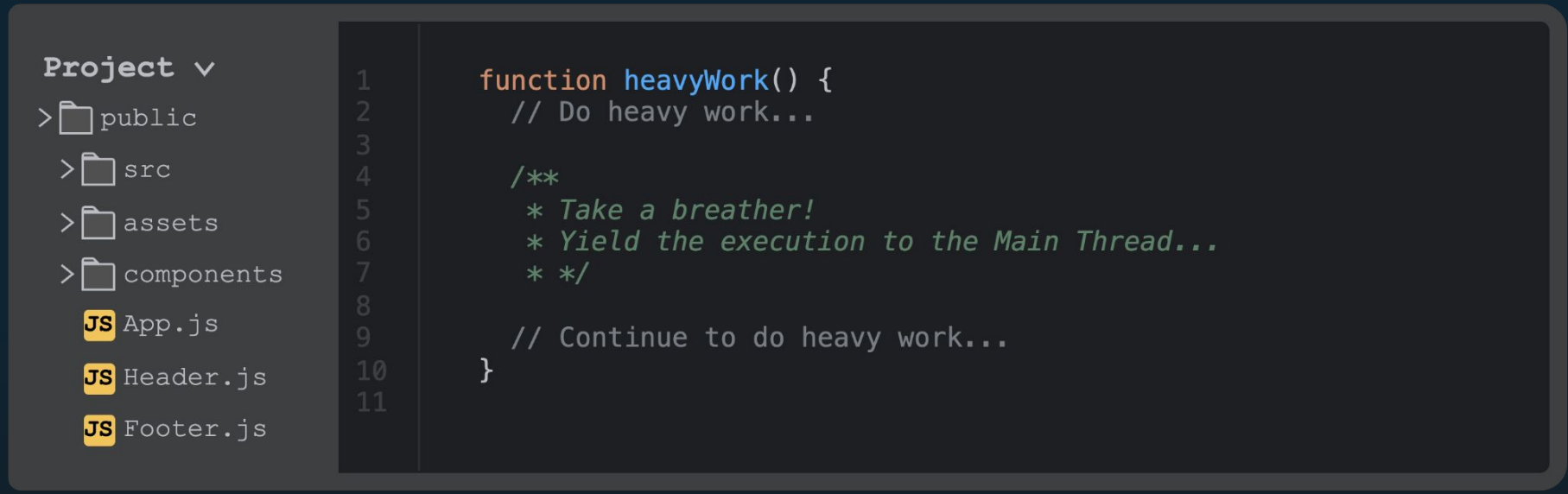


The Problem Demonstration

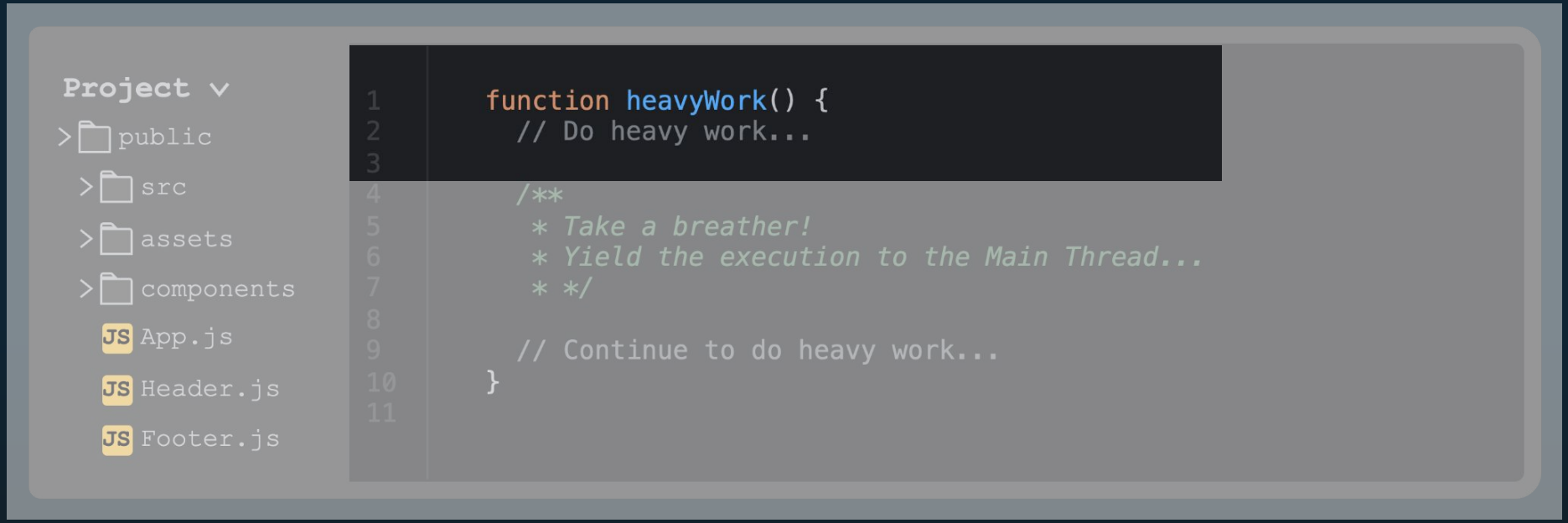


The Problem Solution

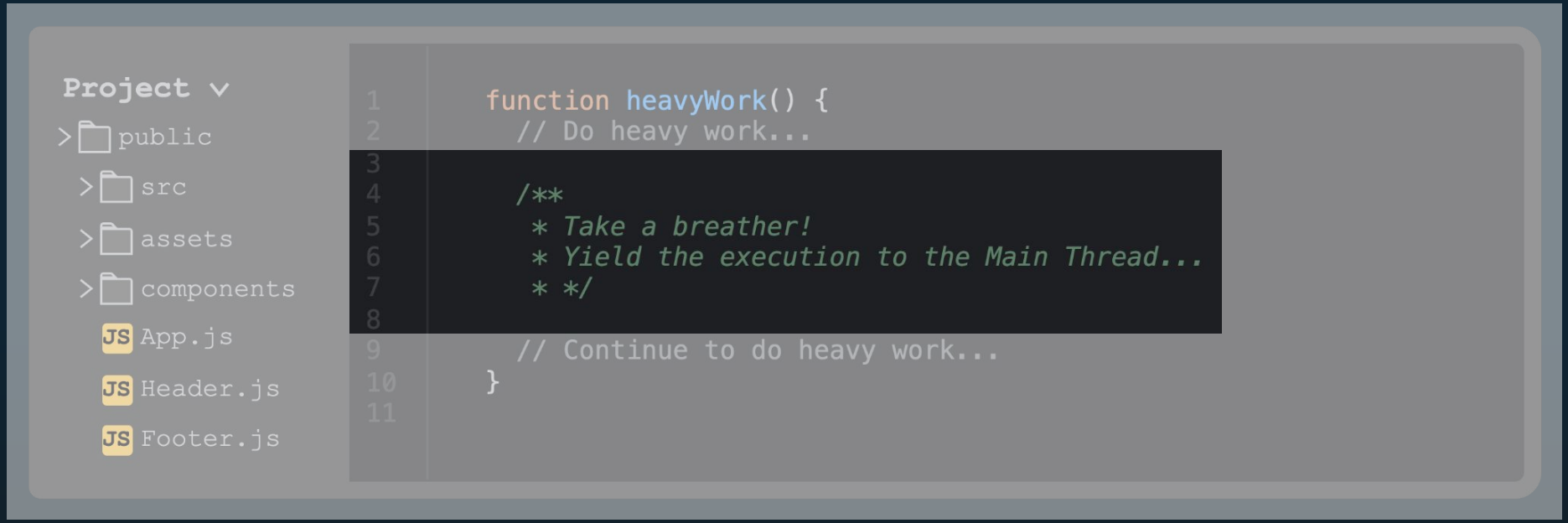
heavyWork - Splitted



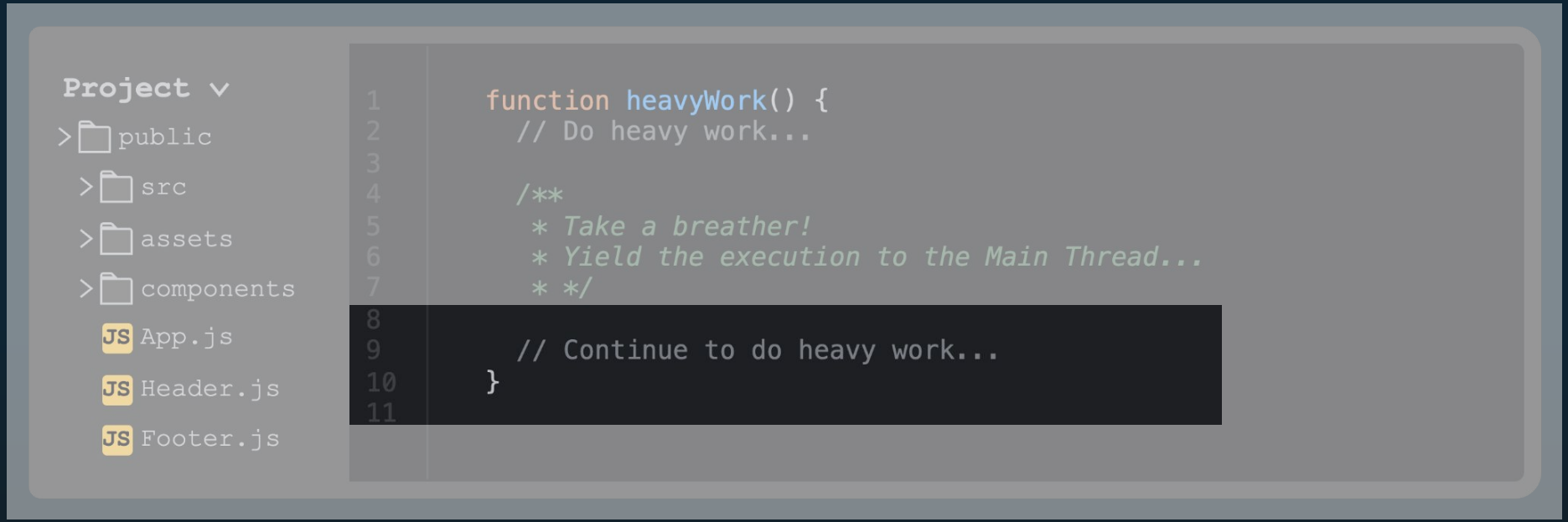
heavyWork — Splitted



heavyWork - Splitted

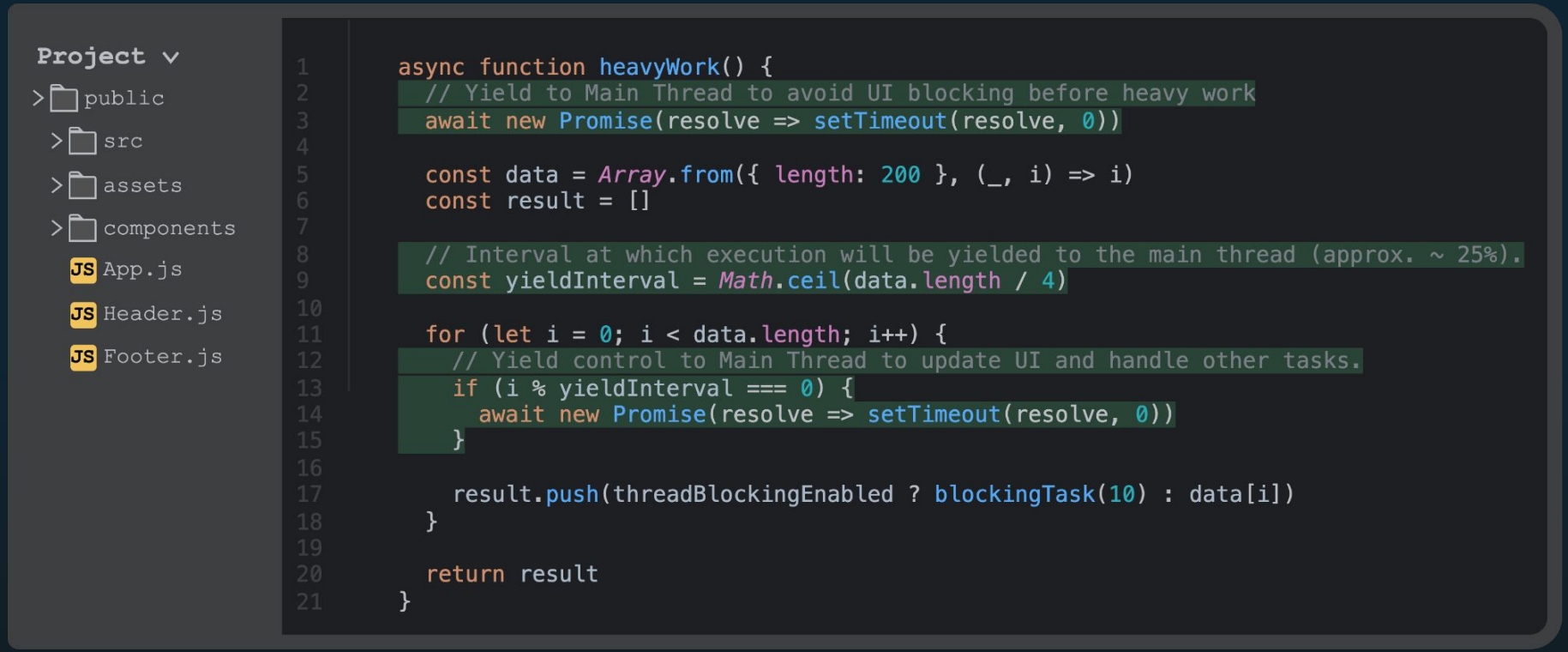


heavyWork - Splitted

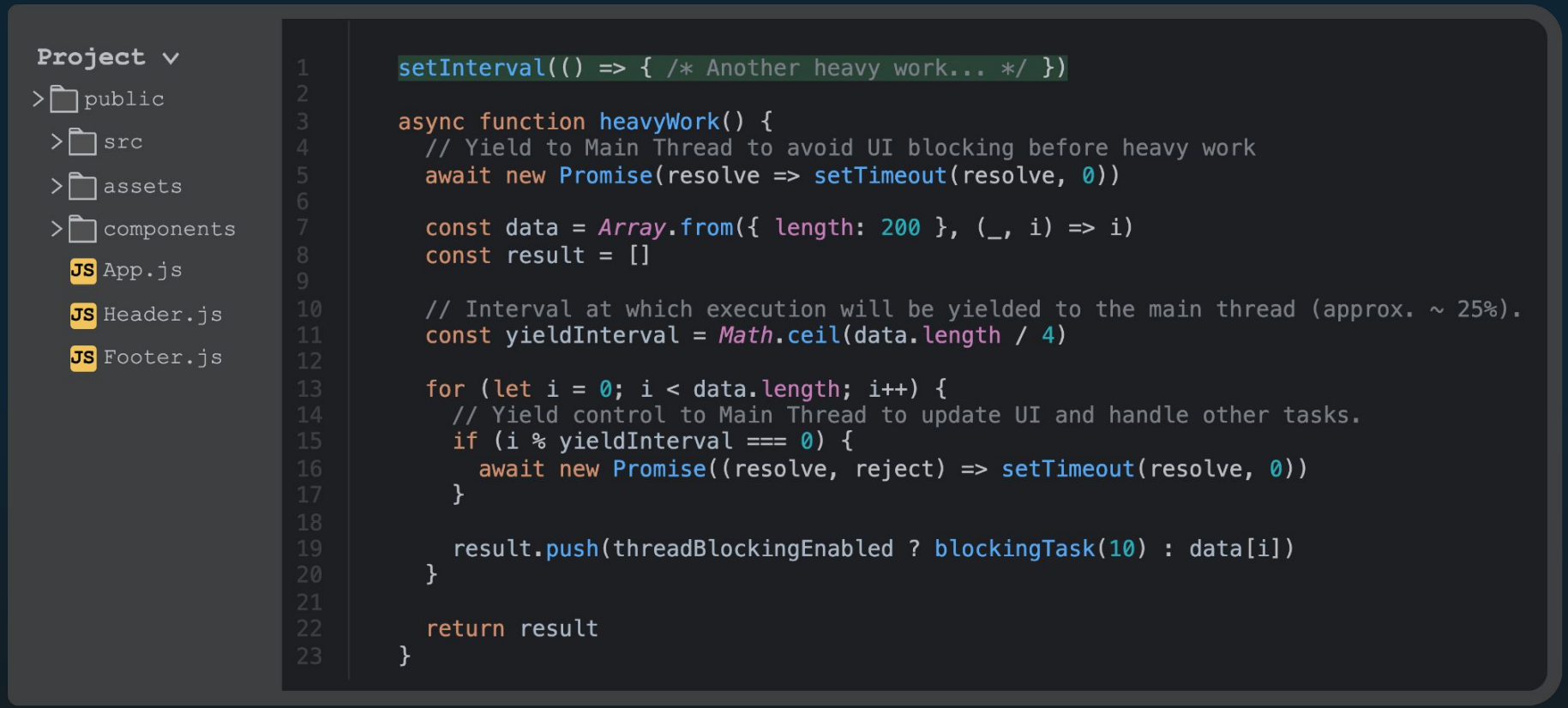


Old Problem-Solving Approaches

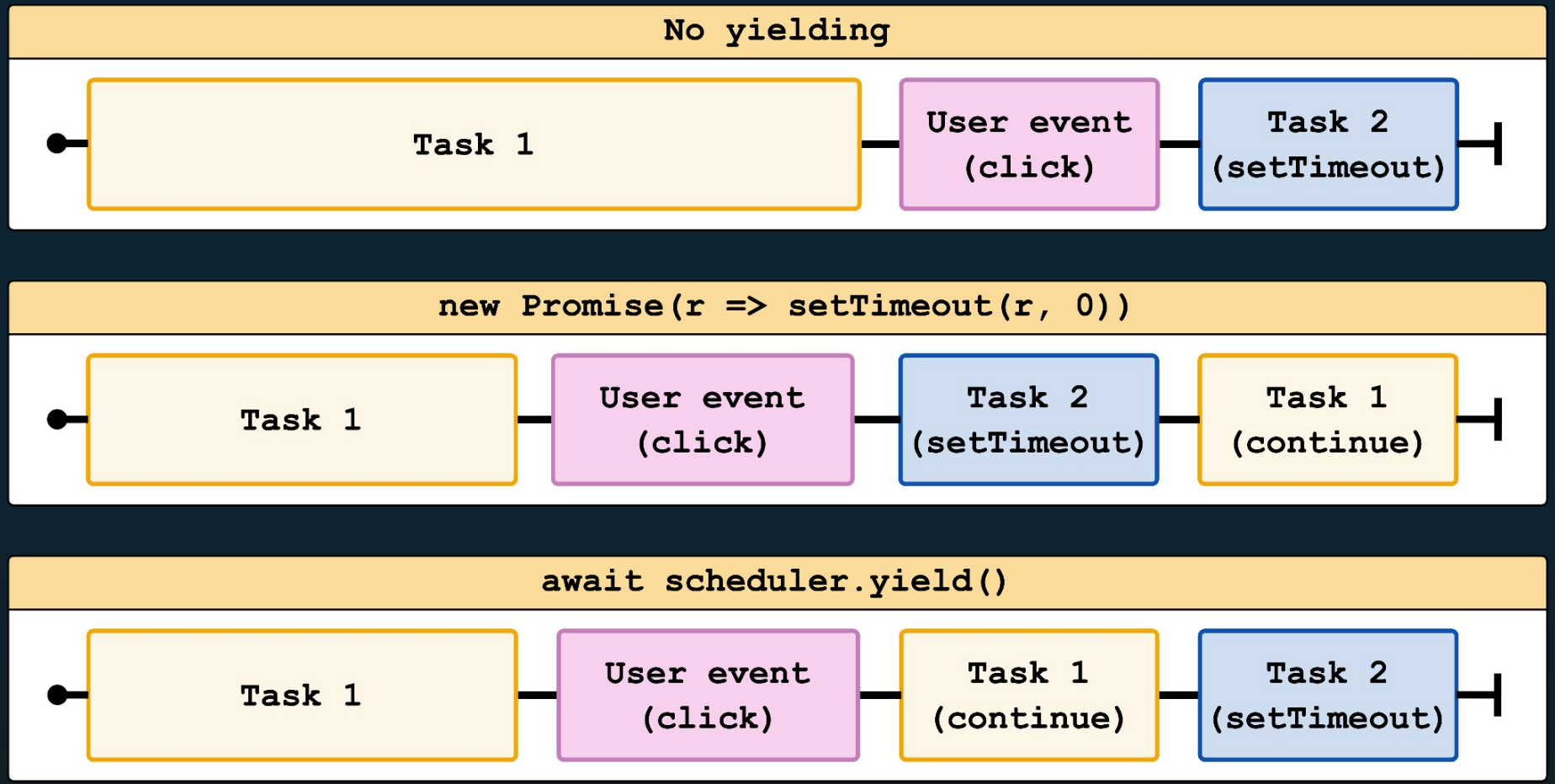
heavyWork — setTimeout()

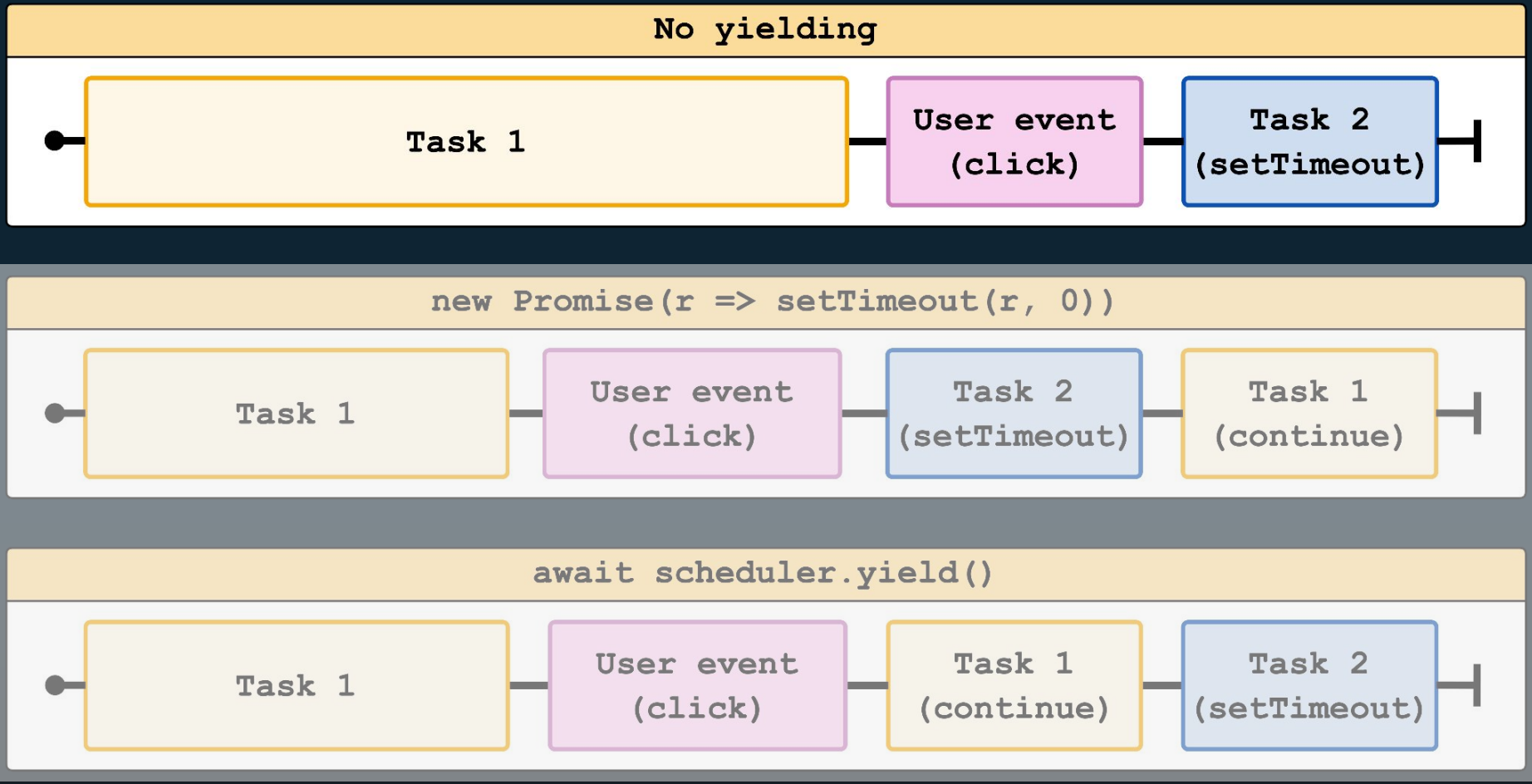


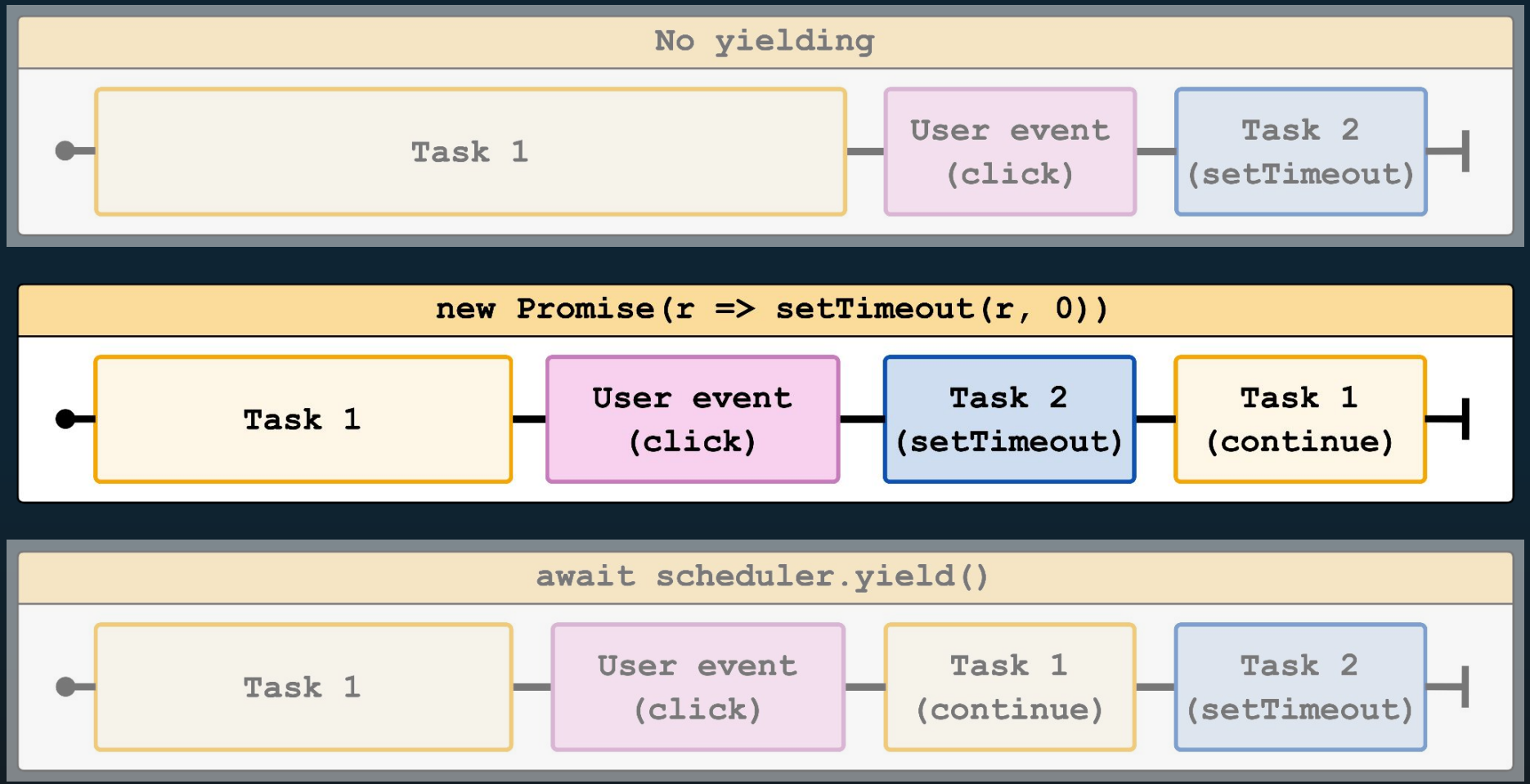
heavyWork — setTimeout()

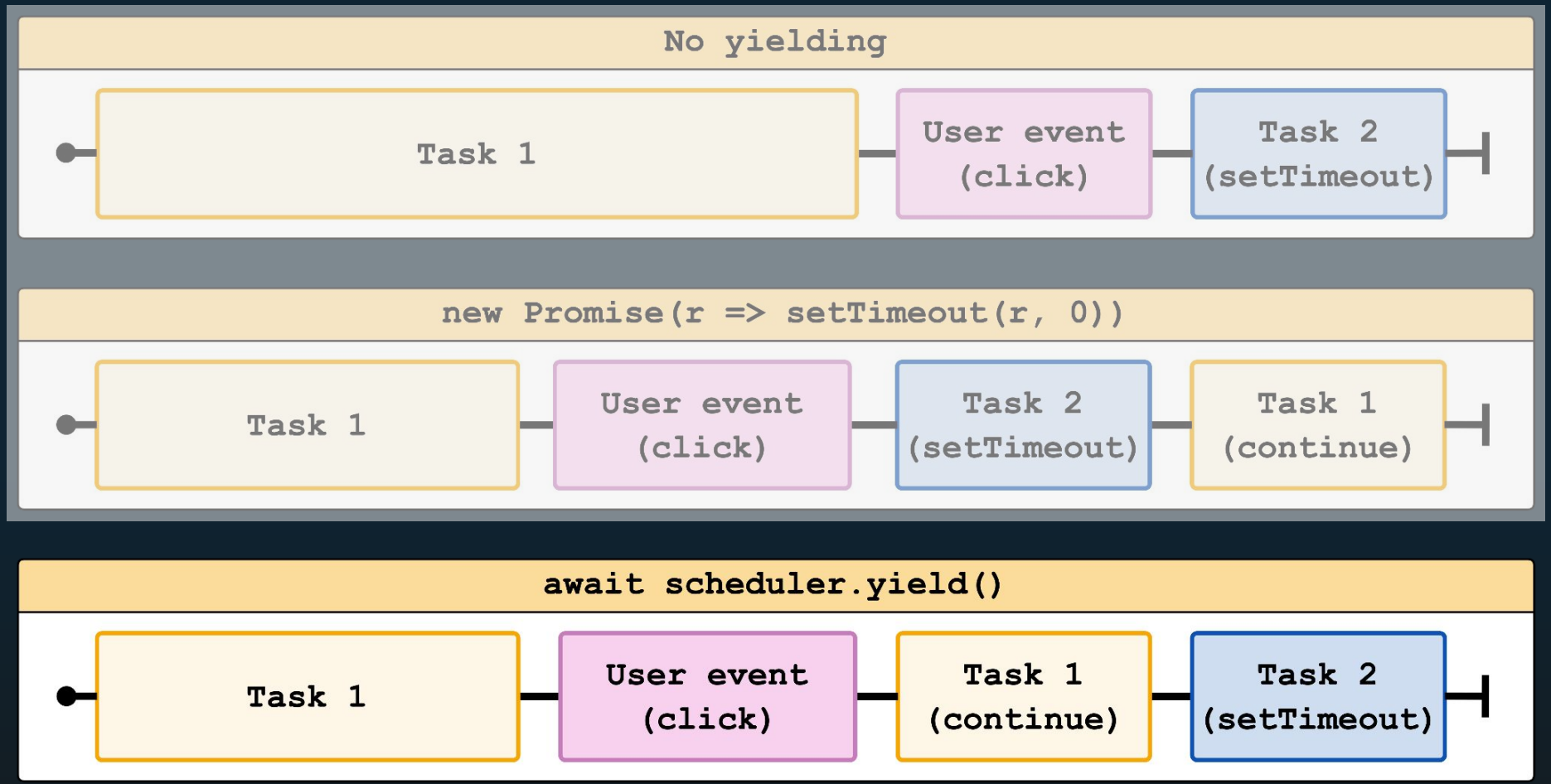


Scheduler.yield()


No yielding
Task 1
User event (click)
Task 2 (setTimeout)
new Promise(r => setTimeout(r, 0))
Task 1
User event (click)
Task 2 (setTimeout)
Task 1 (continue)
await scheduler.yield()
Task 1
User event (click)
Task 1 (continue)
Task 2 (setTimeout)


No yielding
Task 1
User event (click)
Task 2 (setTimeout)
new Promise(r => setTimeout(r, 0))
Task 1
User event (click)
Task 2 (setTimeout)
Task 1 (continue)
await scheduler.yield()
Task 1
User event (click)
Task 1 (continue)
Task 2 (setTimeout)


No yielding
Task 1
User event (click)
Task 2 (setTimeout)
new Promise(r => setTimeout(r, 0))
Task 1
User event (click)
Task 2 (setTimeout)
Task 1 (continue)
await scheduler.yield()
Task 1
User event (click)
Task 1 (continue)
Task 2 (setTimeout)


No yielding
Task 1
User event (click)
Task 2 (setTimeout)
new Promise(r => setTimeout(r, 0))
Task 1
User event (click)
Task 2 (setTimeout)
Task 1 (continue)
await scheduler.yield()
Task 1
User event (click)
Task 1 (continue)
Task 2 (setTimeout)

Priorities

Priorities

1

"user-blocking"

The highest priority tasks that directly affect user interaction, such as handling clicks, taps, and critical UI operations.

Priorities

1

"user-blocking"

The highest priority tasks that directly affect user interaction, such as handling clicks, taps, and critical UI operations.

2

"user-visible"

Tasks that affect UI visibility or content, but are not critical for input.

Priorities

1

"user-blocking"

The highest priority tasks that directly affect user interaction, such as handling clicks, taps, and critical UI operations.

2

"user-visible"

Tasks that affect UI visibility or content, but are not critical for input.

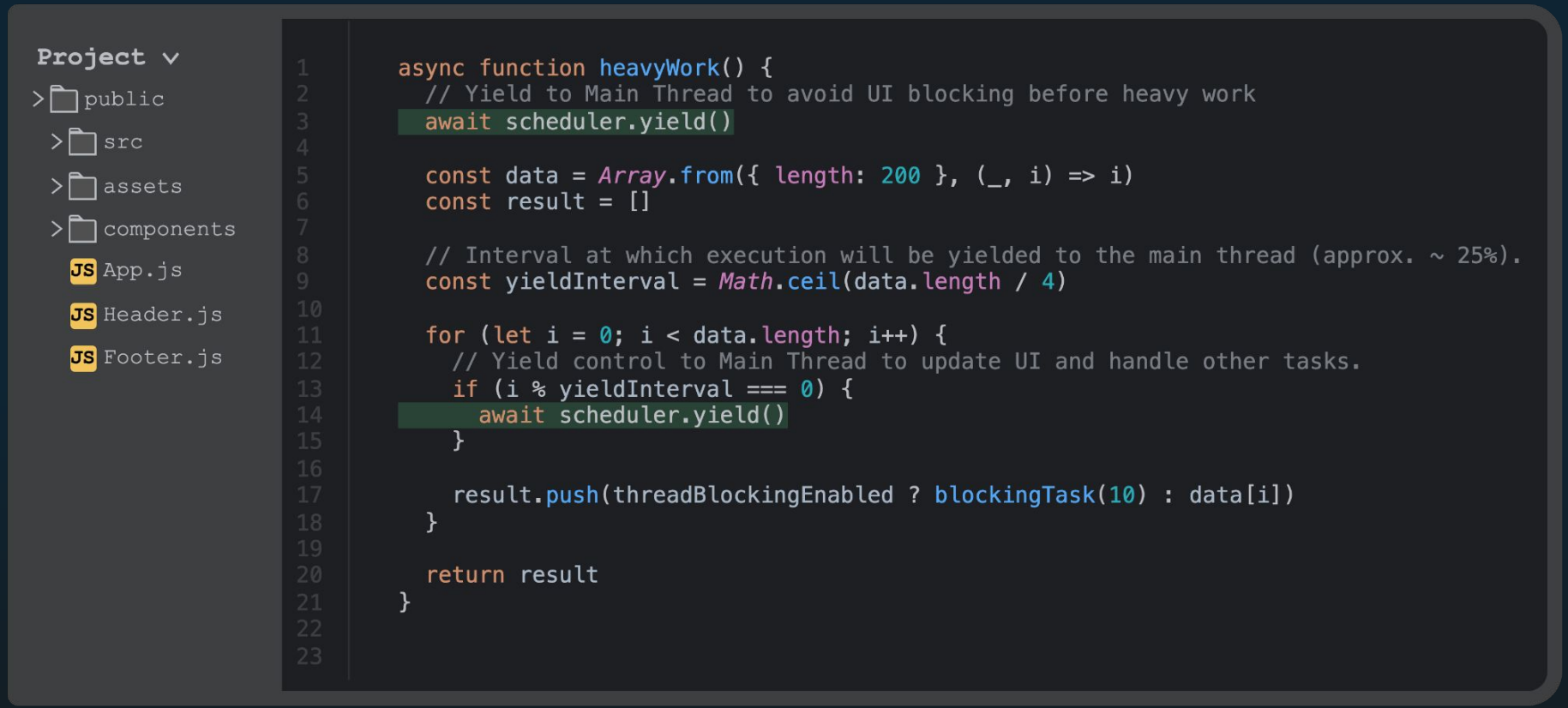
3

"background"

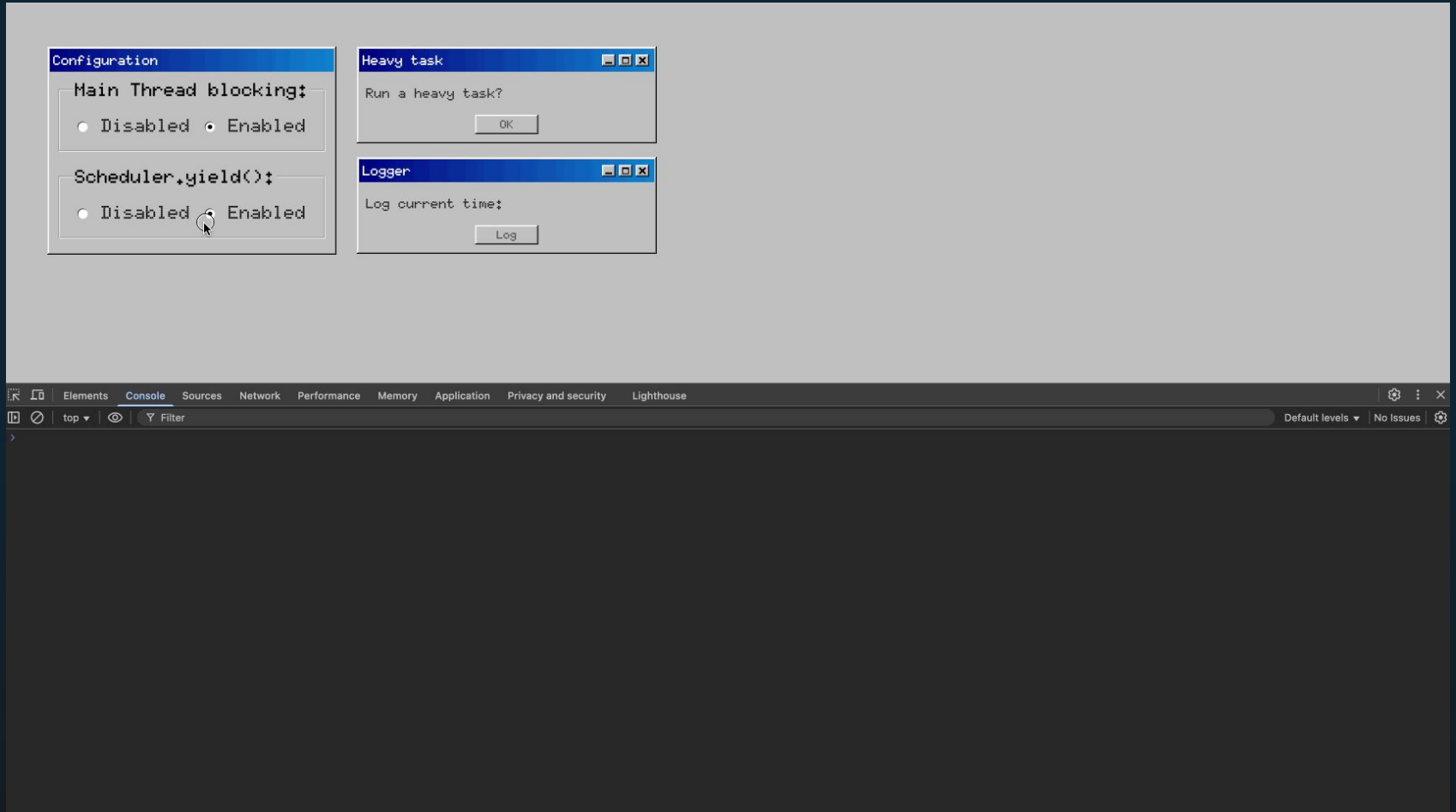
Tasks that can be safely postponed without affecting the current user experience, and are not visible to the user.

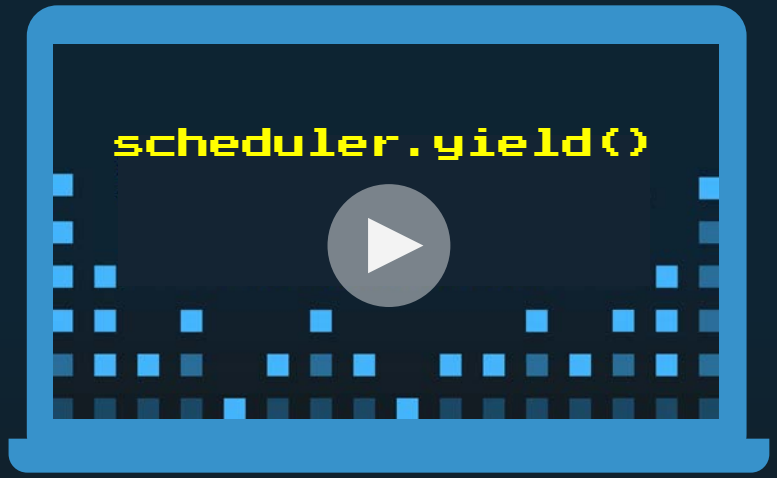
How to use Scheduler.yield() ?

heavyWork — scheduler.yield()



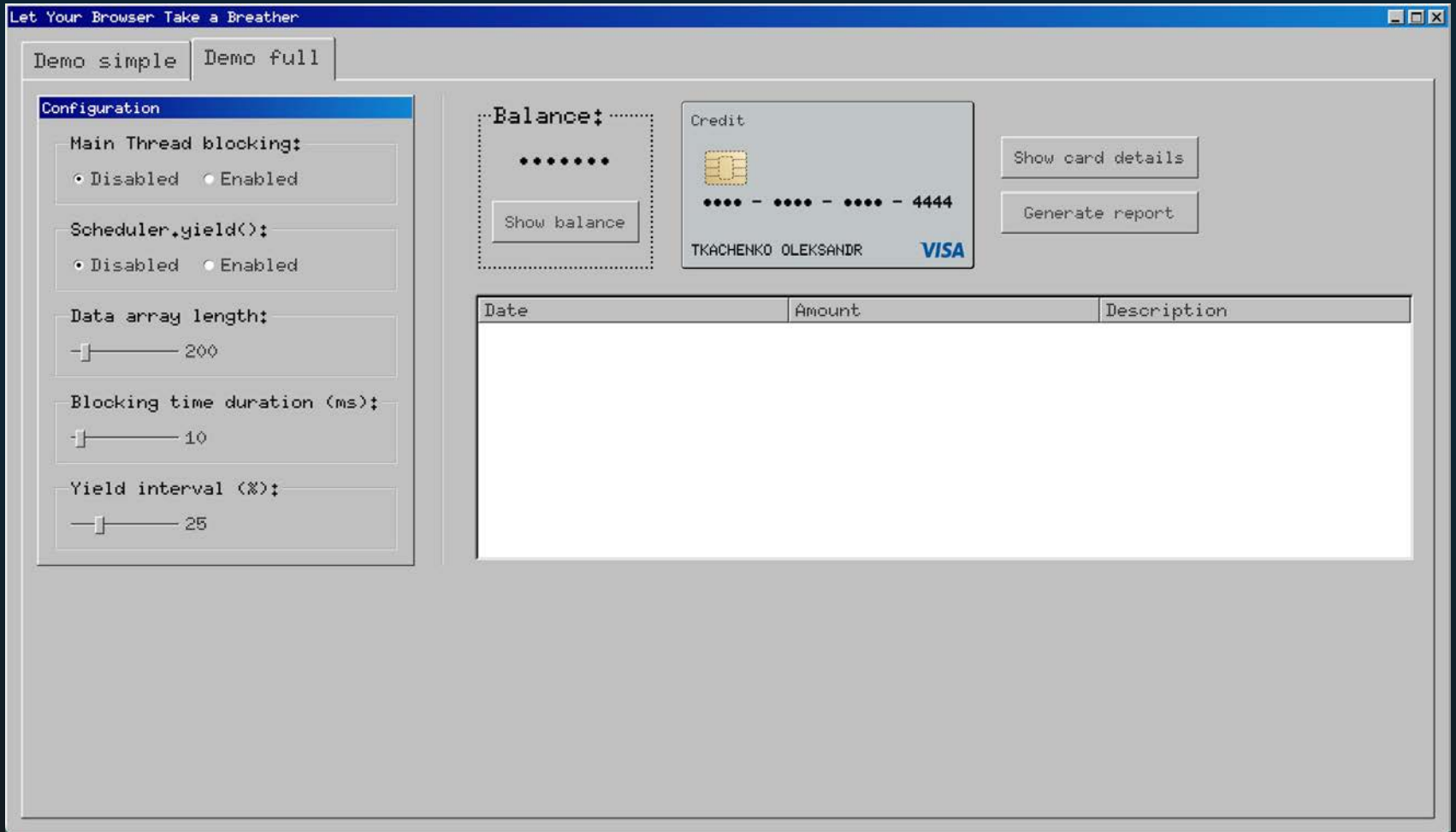
How to use Scheduler.yield() ?



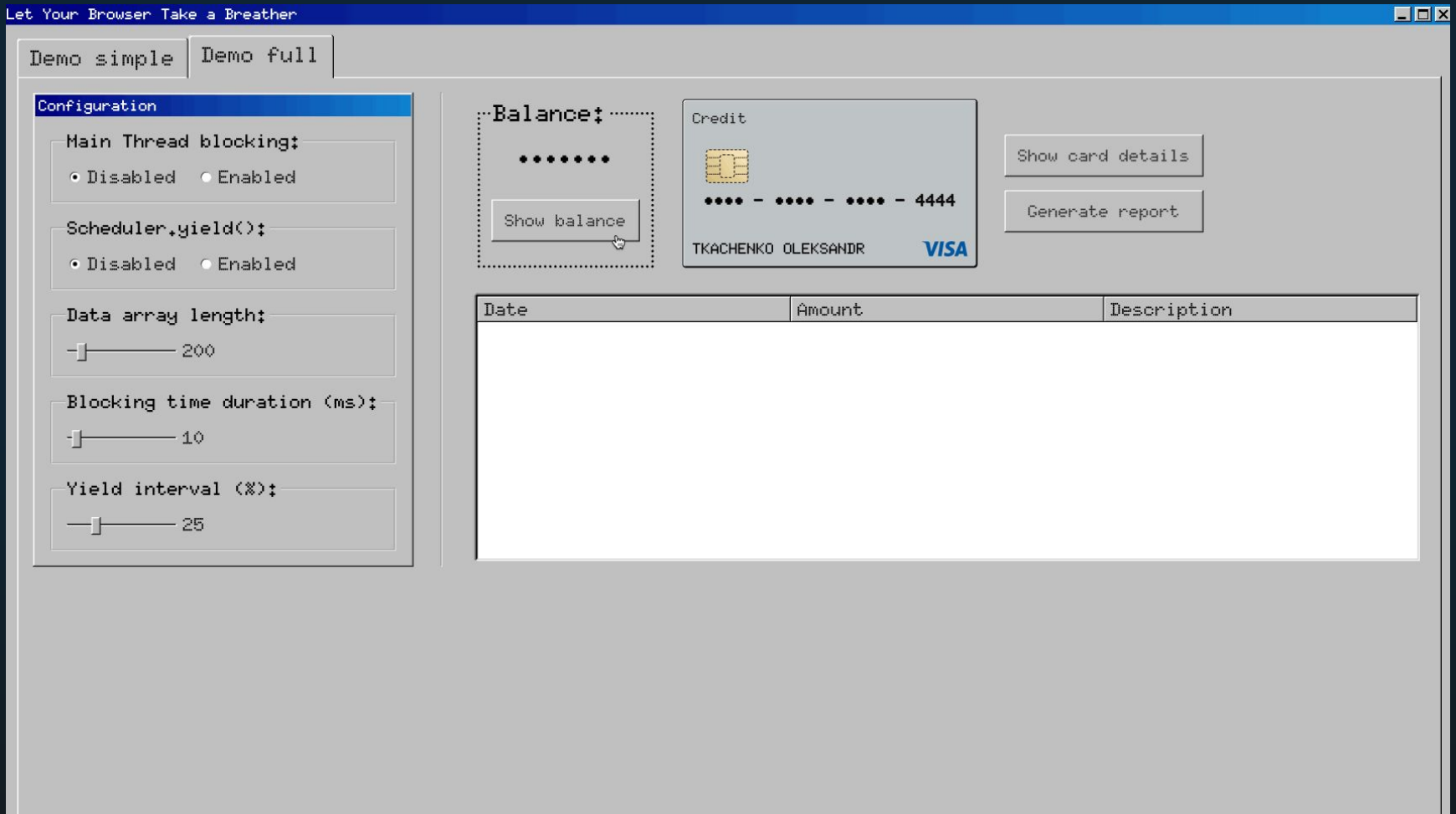


DEMO

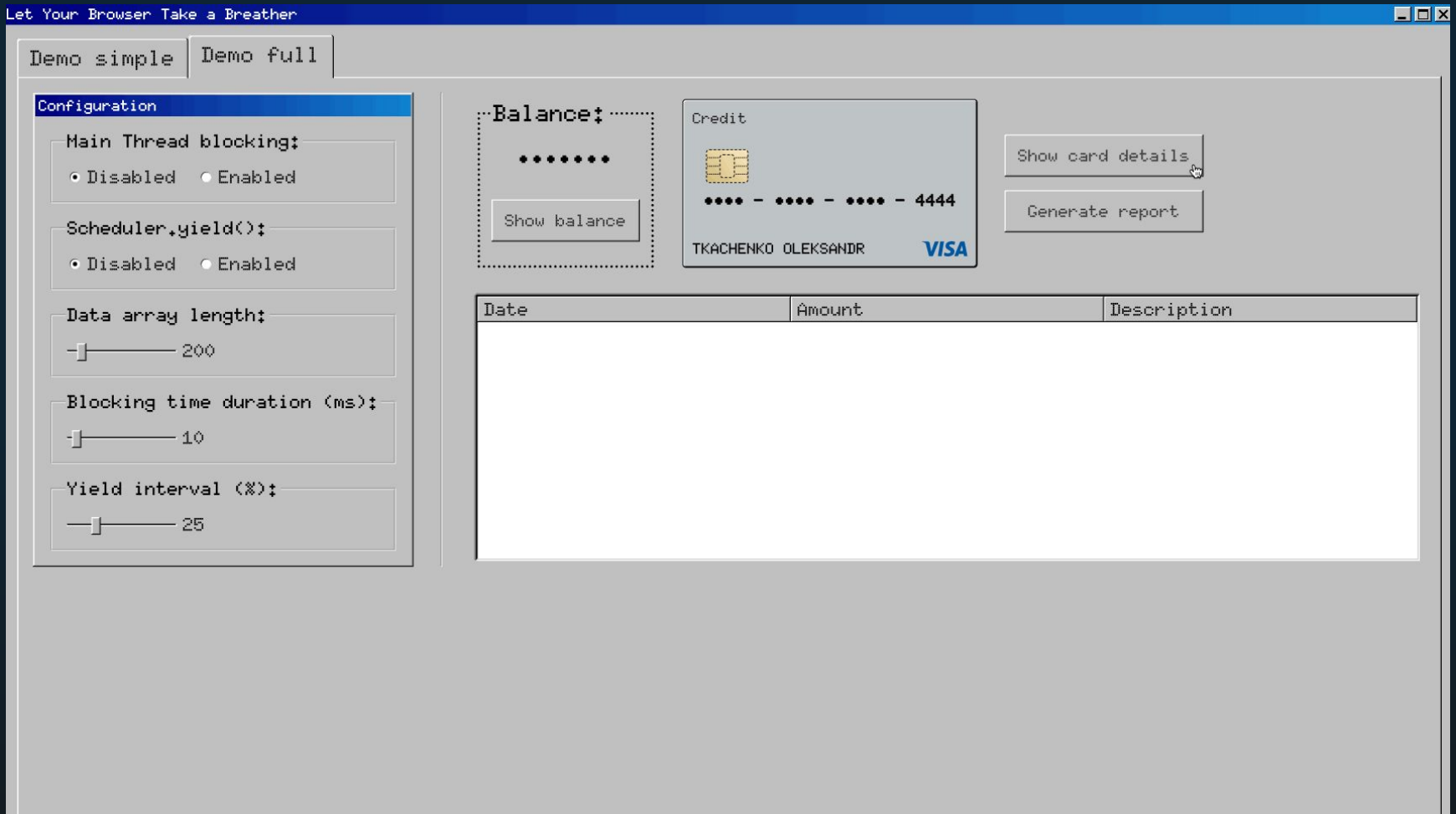
Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
200
Blocking time duration (ms):
10
Yield interval (%):
25
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

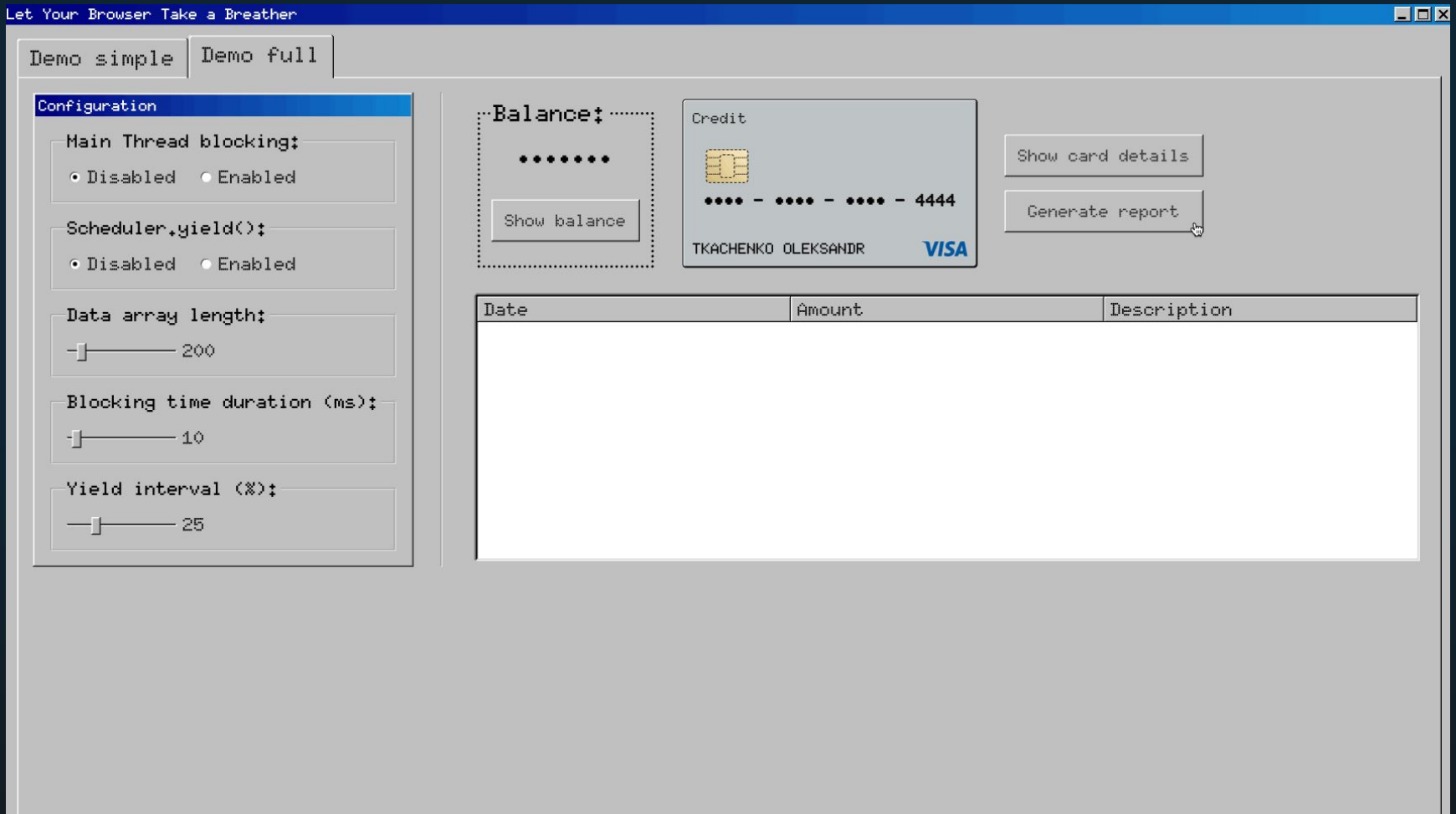
Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
200
Blocking time duration (ms):
10
Yield interval (%):
25
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

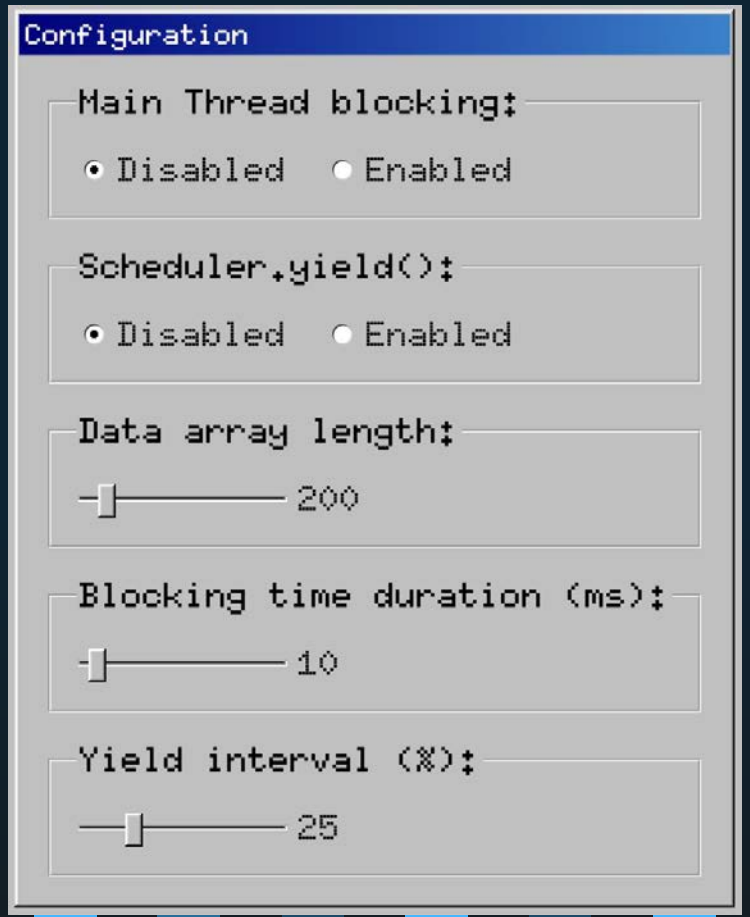
Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
200
Blocking time duration (ms):
10
Yield interval (%):
25
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

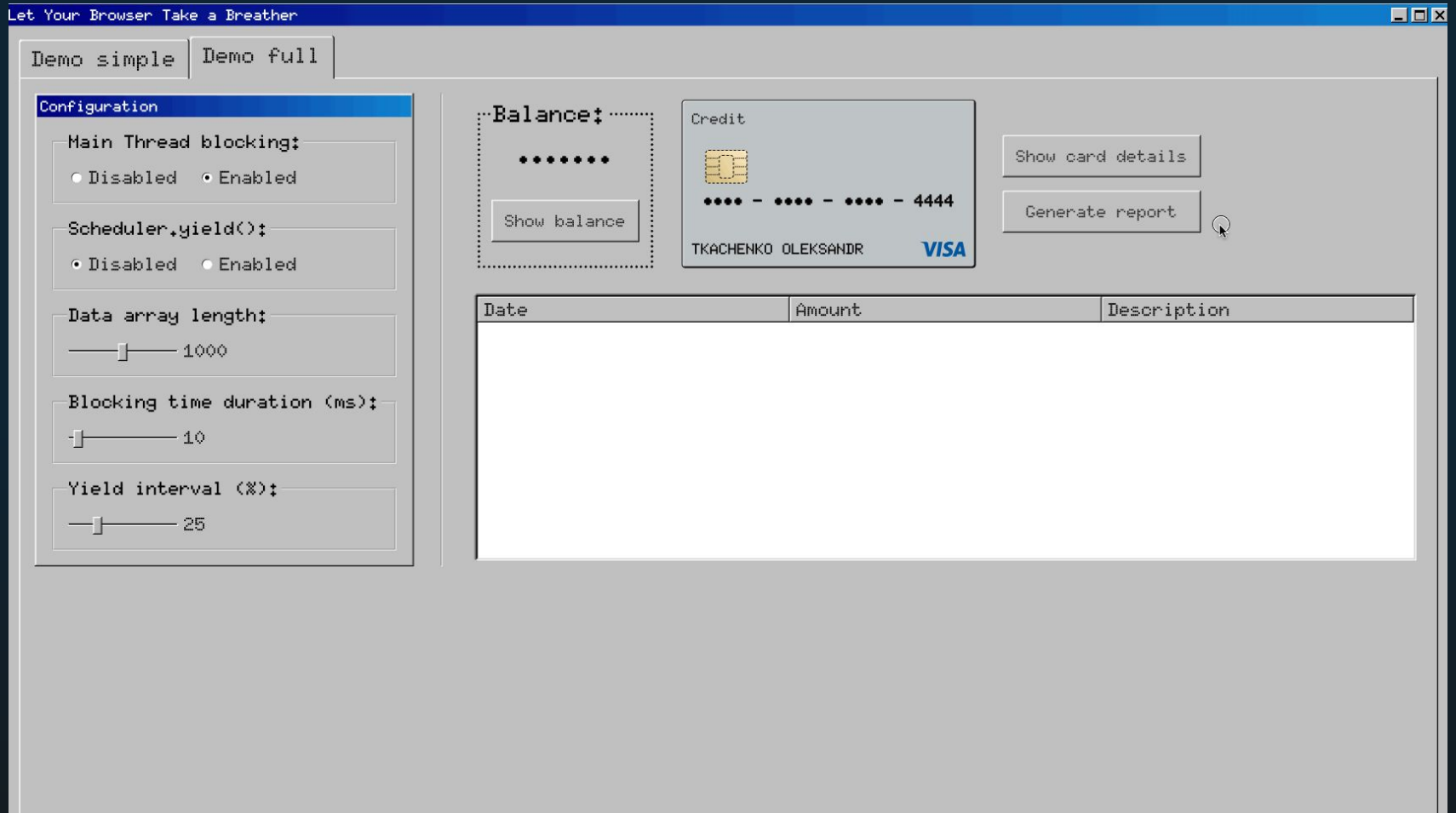
Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
200
Blocking time duration (ms):
10
Yield interval (%):
25
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

Demo


Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
200
Blocking time duration (ms):
10
Yield interval (%):
25

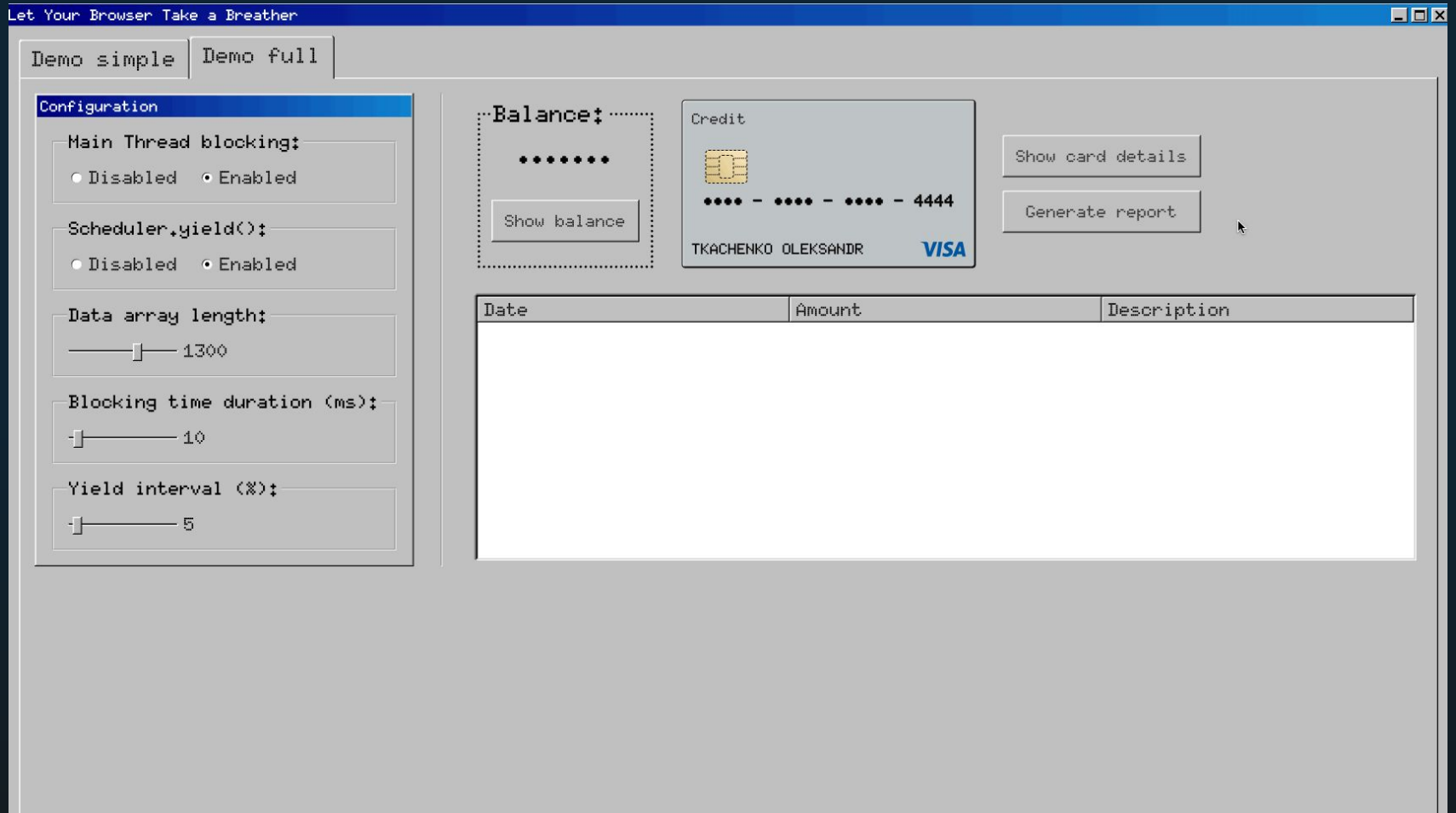
Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
1000
Blocking time duration (ms):
10
Yield interval (%):
25
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

Demo


Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
1300
Blocking time duration (ms):
10
Yield interval (%):
5

Demo


Let Your Browser Take a Breather
Demo simple
Demo full
Configuration
Main Thread blocking:
Disabled
Enabled
Scheduler.yield():
Disabled
Enabled
Data array length:
1300
Blocking time duration (ms):
10
Yield interval (%):
5
Balance:
•••••••
Show balance
Credit
•••• - •••• - •••• - 4444
TKACHENKO OLEKSANDR
VISA
Show card details
Generate report
Date
Amount
Description

THANKS

Scan for useful links