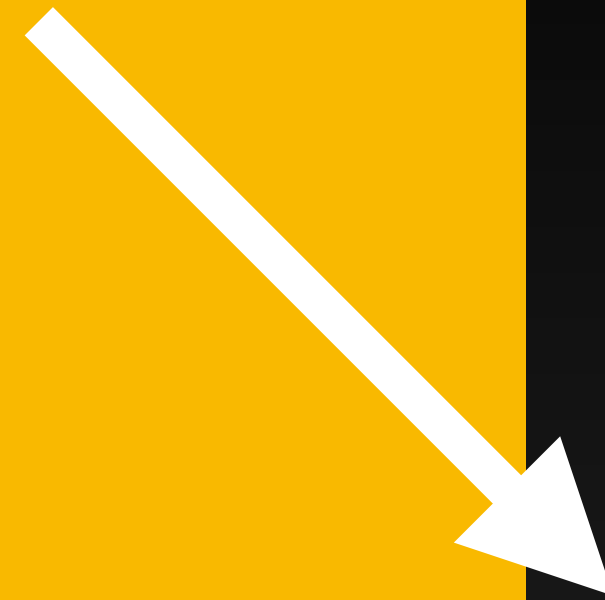```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

執行 line 1，放入 Call Stack

console.log(1)

main()

Call Stack

Web APIs

Event Loop ↻

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

console.log(1) 完成，從
Call Stack pop，
Terminal 印出 1

main()

Call Stack

Web APIs

Event Loop ↻

1

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

執行 line 2 ，放入 Call Stack

setTimeout(cb, 0)

main()

Call Stack

Web APIs

Event Loop

1

Terminal

Task Queue

```javascript
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

發出要求，要瀏覽器設置 0 ms 的倒數計時器，時間到了，就呼叫第一個參數的 arrow function

main()

Call Stack

setTimout(cb, 0)

Web APIs

Event Loop ↻

1

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

倒數完成，將第一個參數的 arrow function 放入 Task Queue

main()

Call Stack

Web APIs

Event Loop ↻

() => {console.log(2)}

1

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說
Event Loop 的機制是等到 Call Stack 空了之後，才會讓 Task Queue 的東西放入 Call Stack。
因為 Call Stack 還有東西，所以 Task Queue 裡面的東西要繼續等待

Call Stack

main()

Web APIs

Event Loop ↻

Terminal

1

Task Queue

() => {console.log(2)}

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

執行 line 5，放入 Call Stack

console.log(3)

main()

Call Stack

Web APIs

Event Loop ↻

1

Terminal

() => {console.log(2)}

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

console.log(3) 完成，從
Call Stack pop，
Terminal 印出 3

main()

Call Stack

Web APIs

Event Loop

1
3

() => {console.log(2)}

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

執行 line 6 ，放入 Call Stack

setTimeout(cb, 0)

main()

Call Stack

Web APIs

Event Loop ↻

1
3

Terminal

() => {console.log(2)}

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

發出要求，要瀏覽器設置 0 ms 的倒數計時器，時間到了，就呼叫第一個參數的 arrow function

main()

Call Stack

setTimout(cb, 0)

Web APIs

Event Loop ↻

1
3

Terminal

() => {console.log(2)}

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

倒數完成，將第一個參數的 arrow function 放入 Task Queue

Call Stack

main()

Web APIs

Event Loop ↻

```
1
3
```

Terminal

() => {console.log(2)}

() => {console.log(4)}

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

因為 Call Stack 還有東西，所以 Task Queue 裡面的東西要繼續等待

Call Stack

main()

Web APIs

Event Loop ↻

Terminal

1
3

Task Queue

() => {console.log(2)}

() => {console.log(4)}

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

執行 line 9，放入 Call Stack

console.log(5)

main()

Call Stack

Web APIs

Event Loop ↻

1
3

Terminal

() => {console.log(2)}

() => {console.log(4)}

Task Queue

```
console.log(1)
setTimeout(() => {
    console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
    console.log(4)
}, 0)
console.log(5)
```

步驟解說

console.log(5) 完成，從 Call Stack pop，Terminal 印出5

main()

Call Stack

Web APIs

Event Loop ↻

```
1
3
5
```

Terminal

() => {console.log(2)}

() => {console.log(4)}

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

main()，完成 pop

Call Stack

Web APIs

Event Loop ↻

1
3
5

() => {console.log(2)}

() => {console.log(4)}

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

() => {
console.log(2)
}

Call Stack

Web APIs

Event Loop ↻

```
1
3
5
```

() => {console.log(4)}

Terminal

Task Queue

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

印出 2，完成，從 Call Stack pop

Call Stack

Web APIs

Event Loop ↻

Terminal

1   2
3
5

Task Queue

() => {console.log(4)}

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

Call Stack 空了，所以 Event Loop 放行 Task Queue 的東西進來

() => {
console.log(4)
}

Call Stack

Web APIs

Event Loop ↻

1   2
3
5

Task Queue

Terminal

```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

步驟解說

印出 4，完成，從 Call Stack pop

Event Loop ↻

Call Stack

Web APIs

1	2
3	4
5	

Terminal

Task Queue