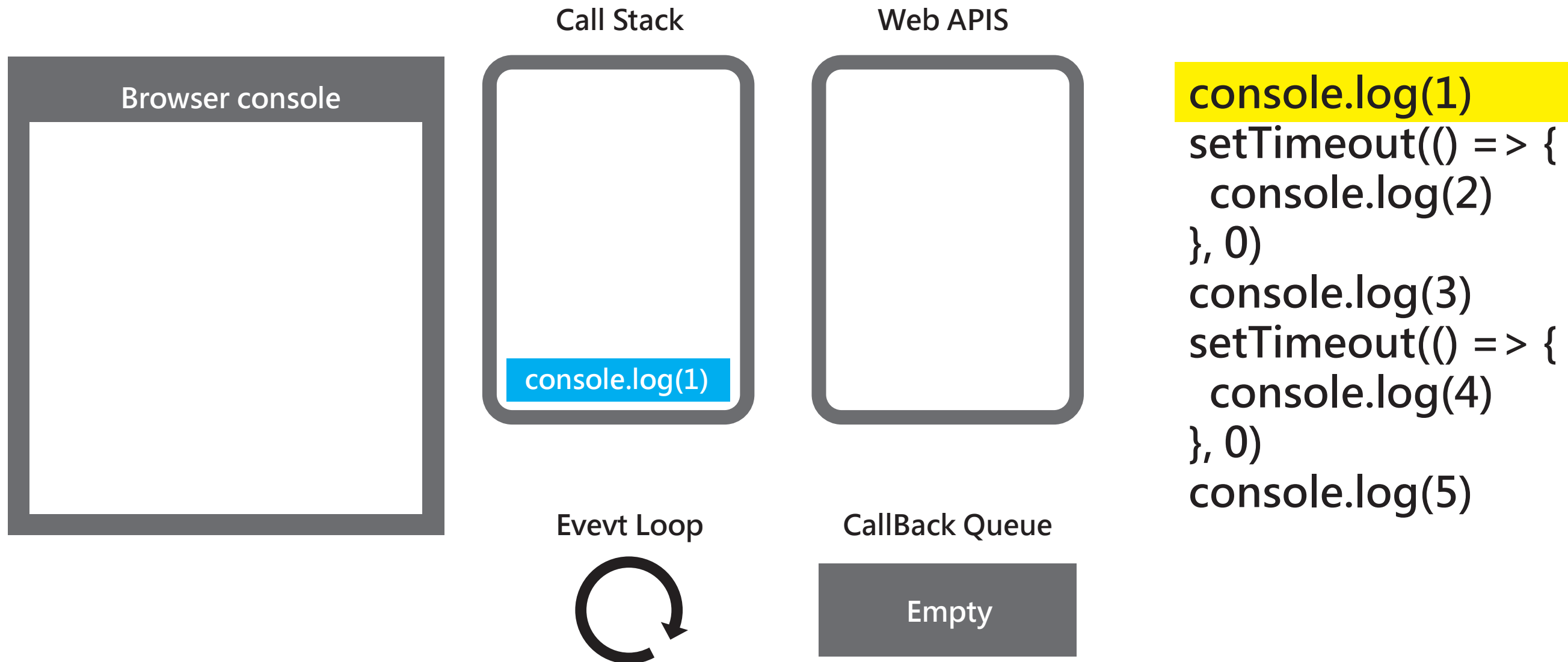
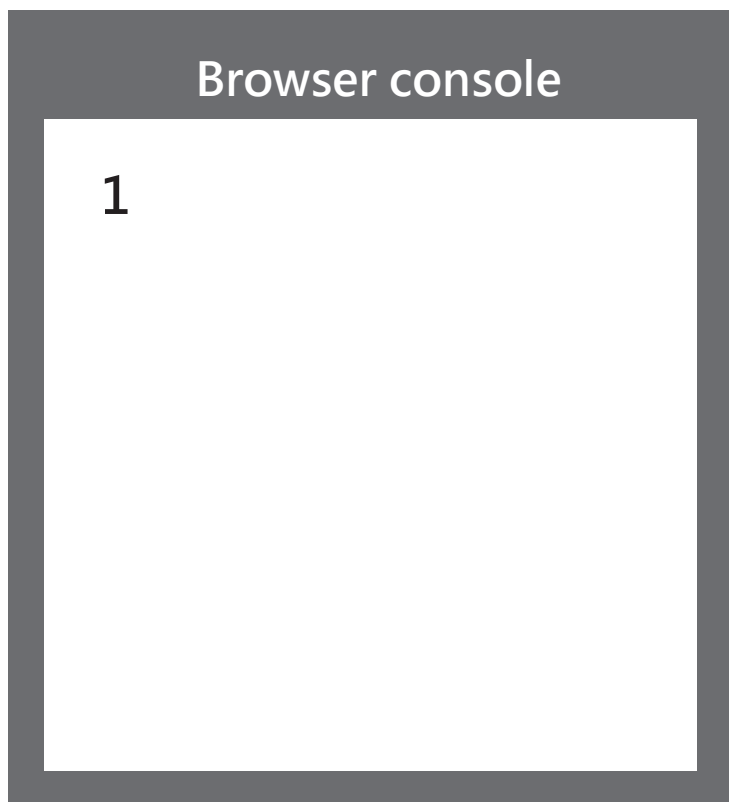


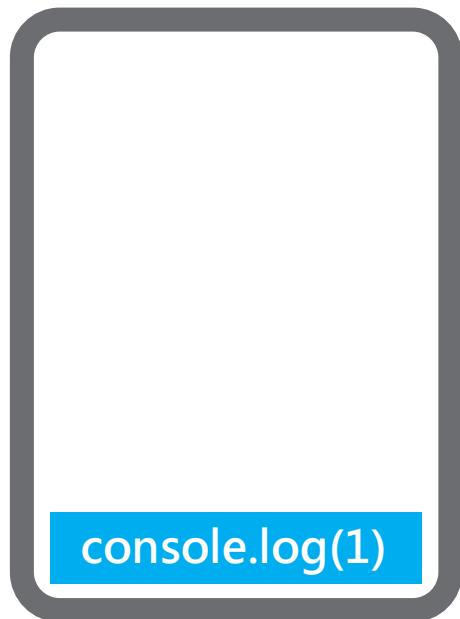
1. 新增 console.log(1) 至 Call Stack



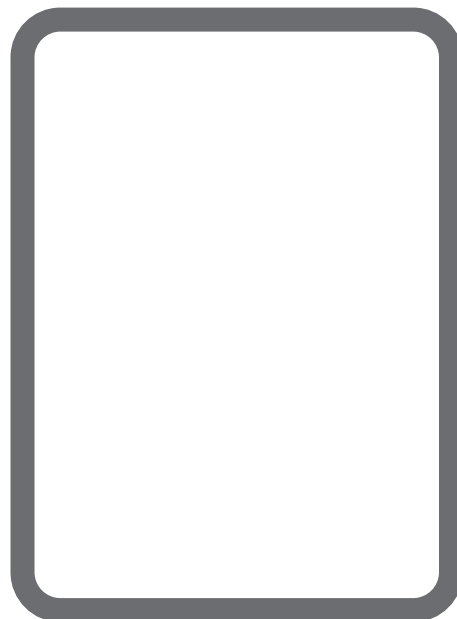
2. console.log(1)被執行



Call Stack



Web APIS



Event Loop

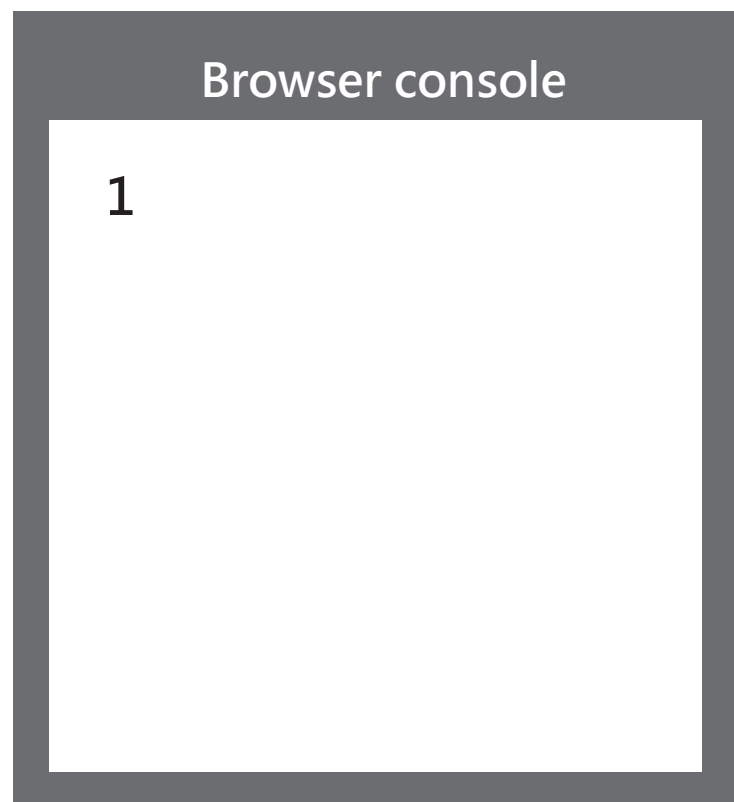


Callback Queue



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

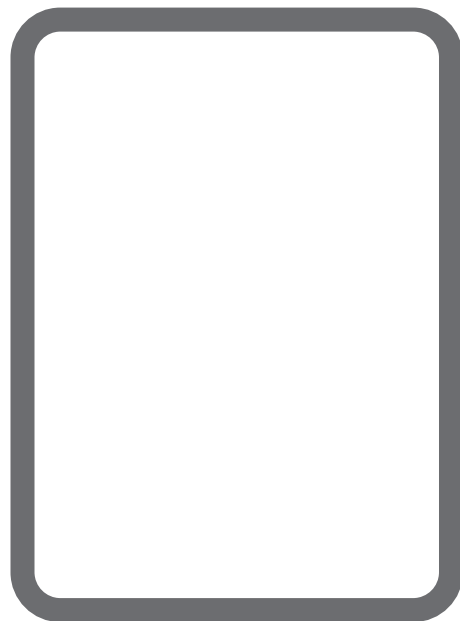
3. console.log(1) 執行完被移除出 Call Stack



Call Stack



Web APIS



Event Loop

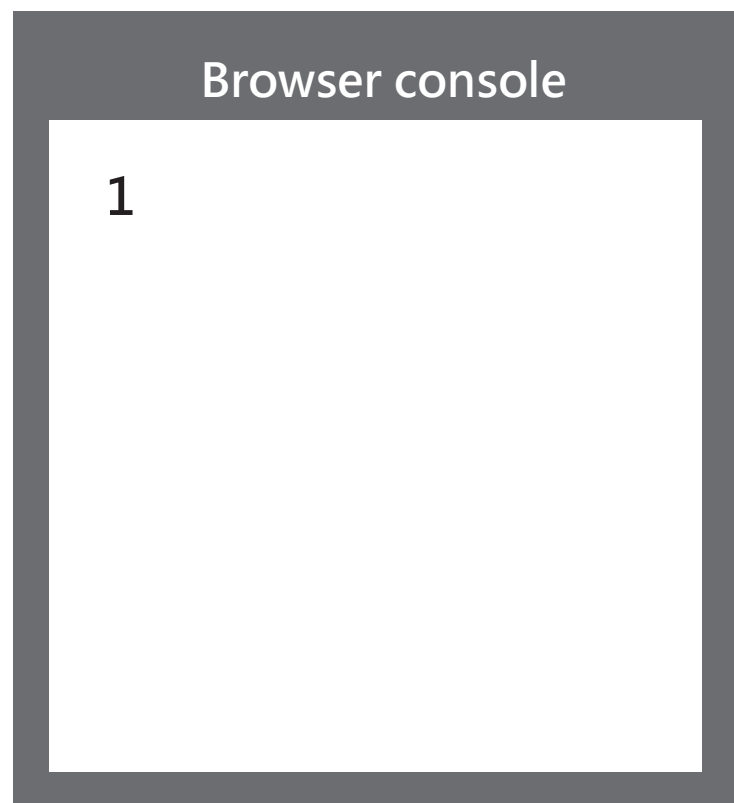


Callback Queue

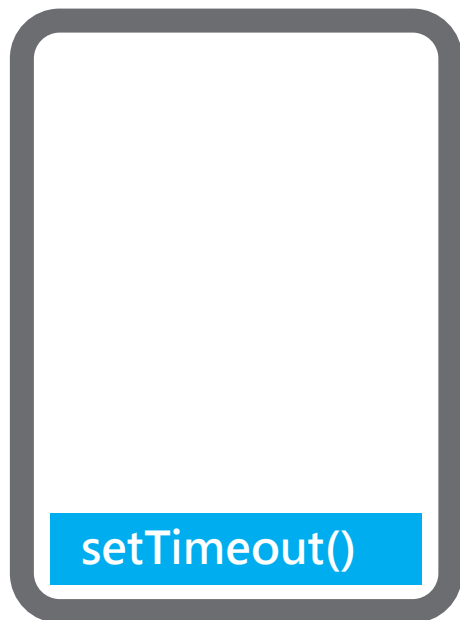


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

4. 新增 setTimeout(() { ... }) 至 Call Stack



Call Stack



Web APIS



Event Loop

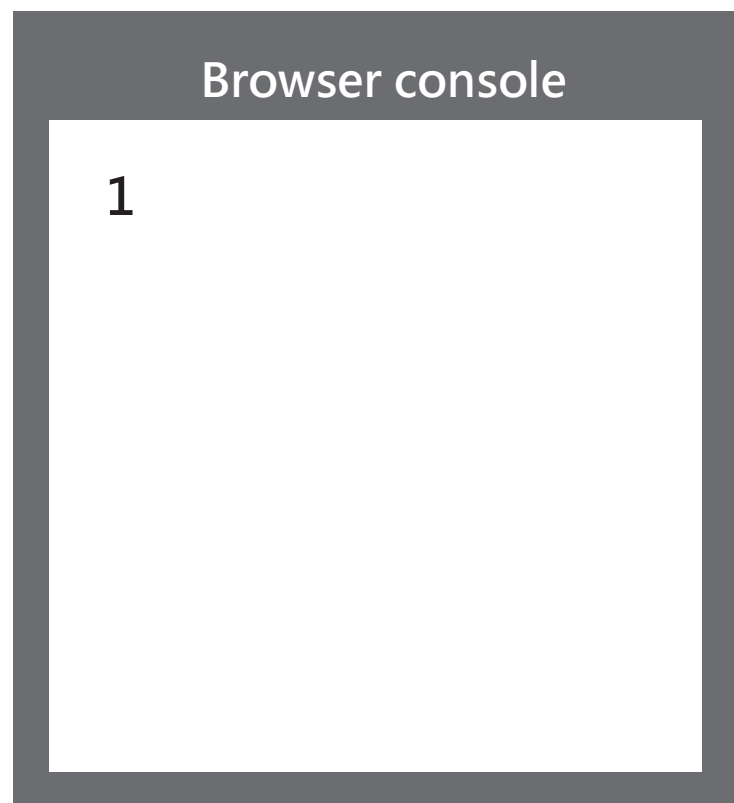


Callback Queue

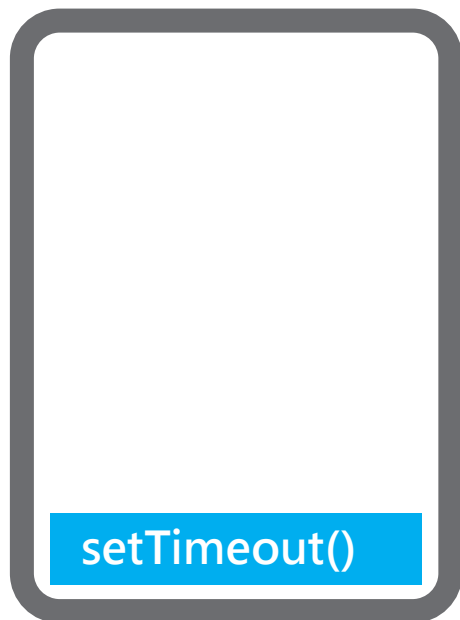


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

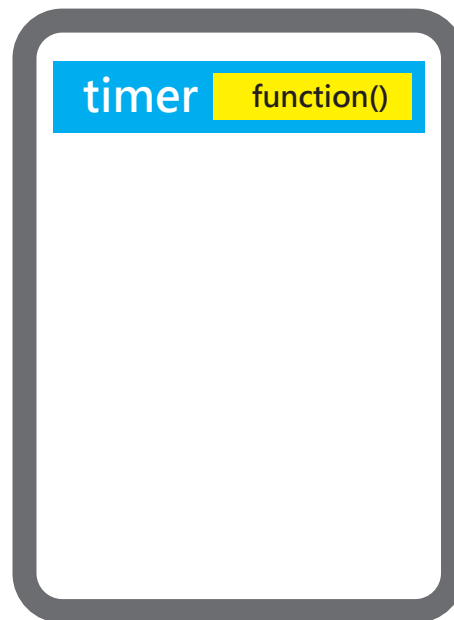
5. `setTimeout(() { ... })` 被執行，瀏覽器會根據 web API 建立一個定時器



Call Stack



Web APIS



Event Loop

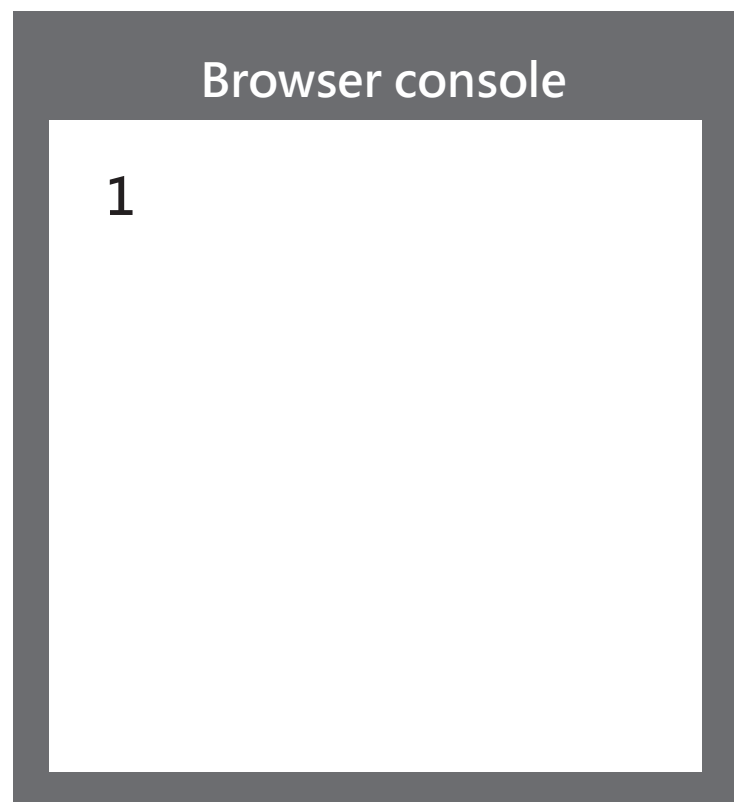


Callback Queue



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

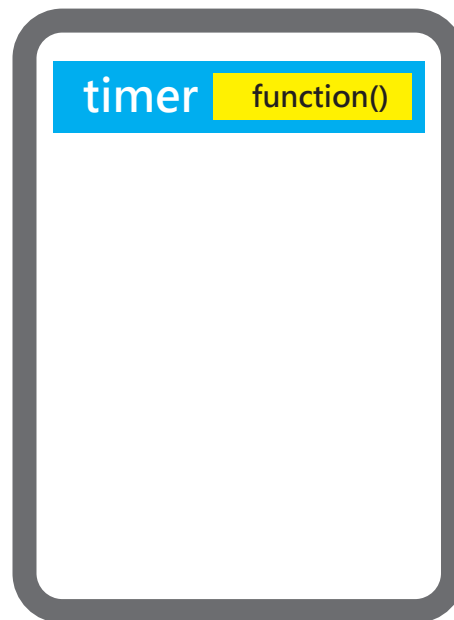
6. setTimeout(() { ... }) 執行完被移除出 Call Stack



Call Stack



Web APIS



Event Loop

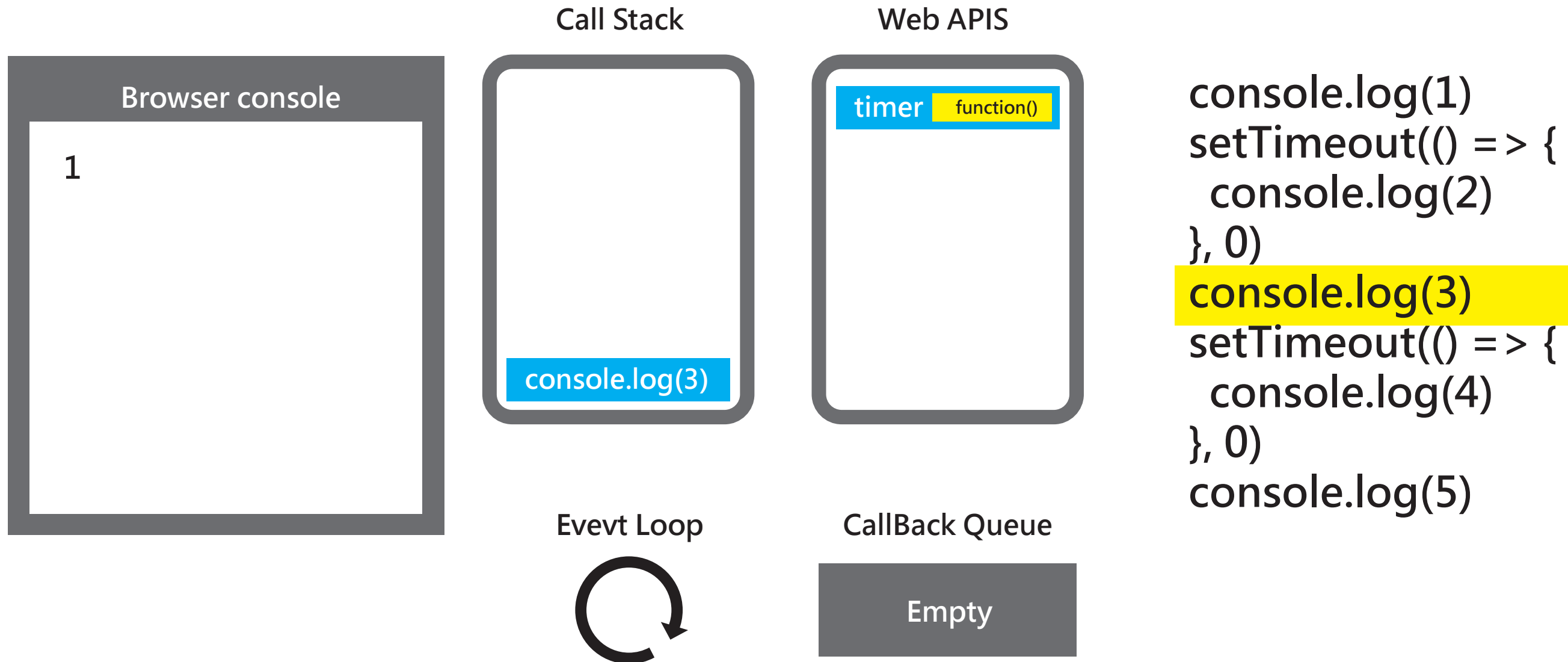


Callback Queue

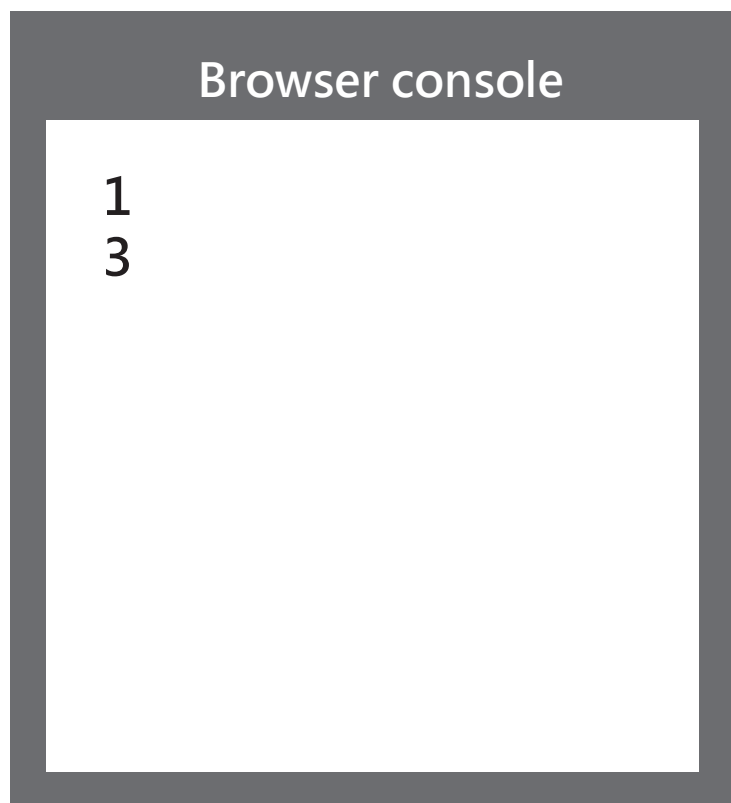


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

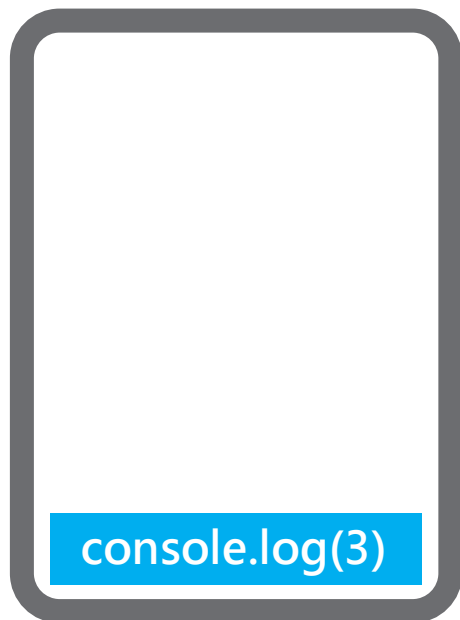
7. 新增 console.log(3) 至 Call Stack



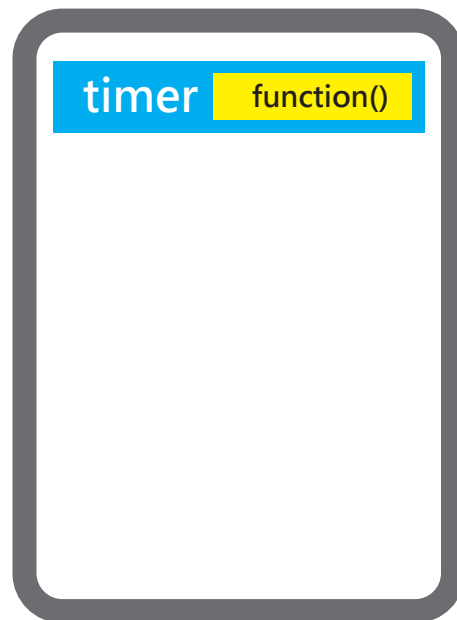
8. console.log(3)被執行



Call Stack



Web APIS



Event Loop

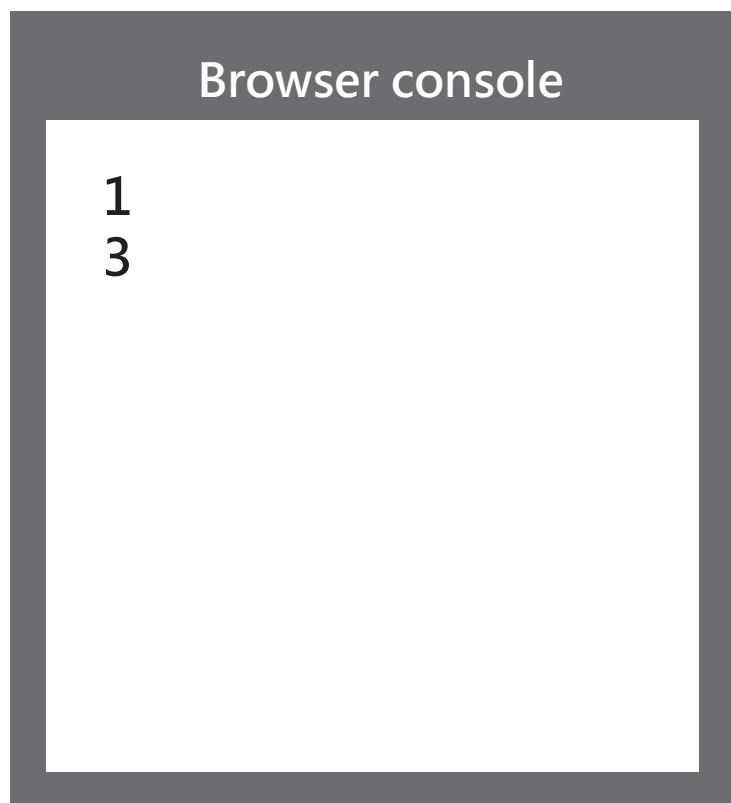


Callback Queue

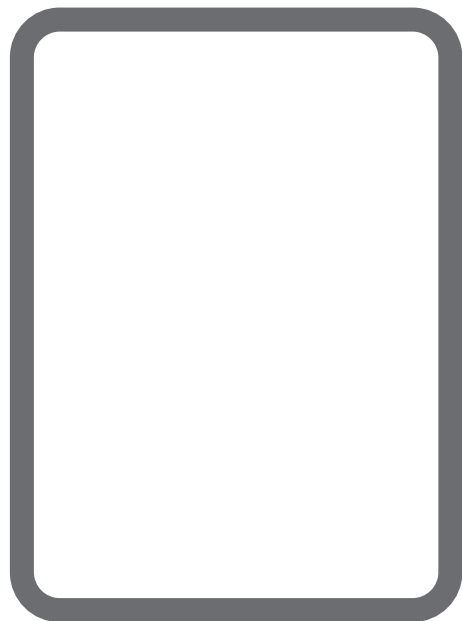


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

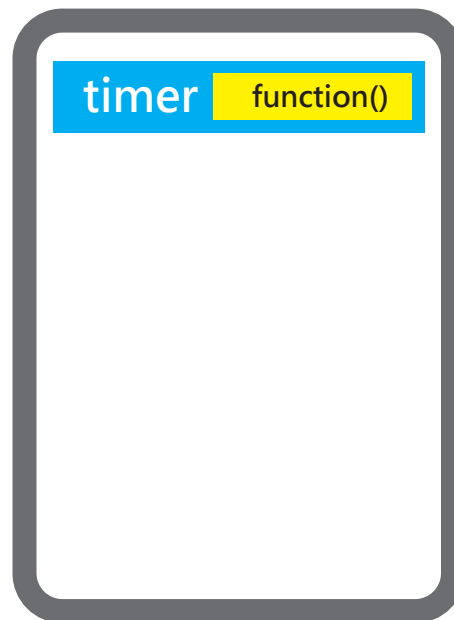

9. console.log(3) 執行完被移除出 Call Stack



Call Stack



Web APIS



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

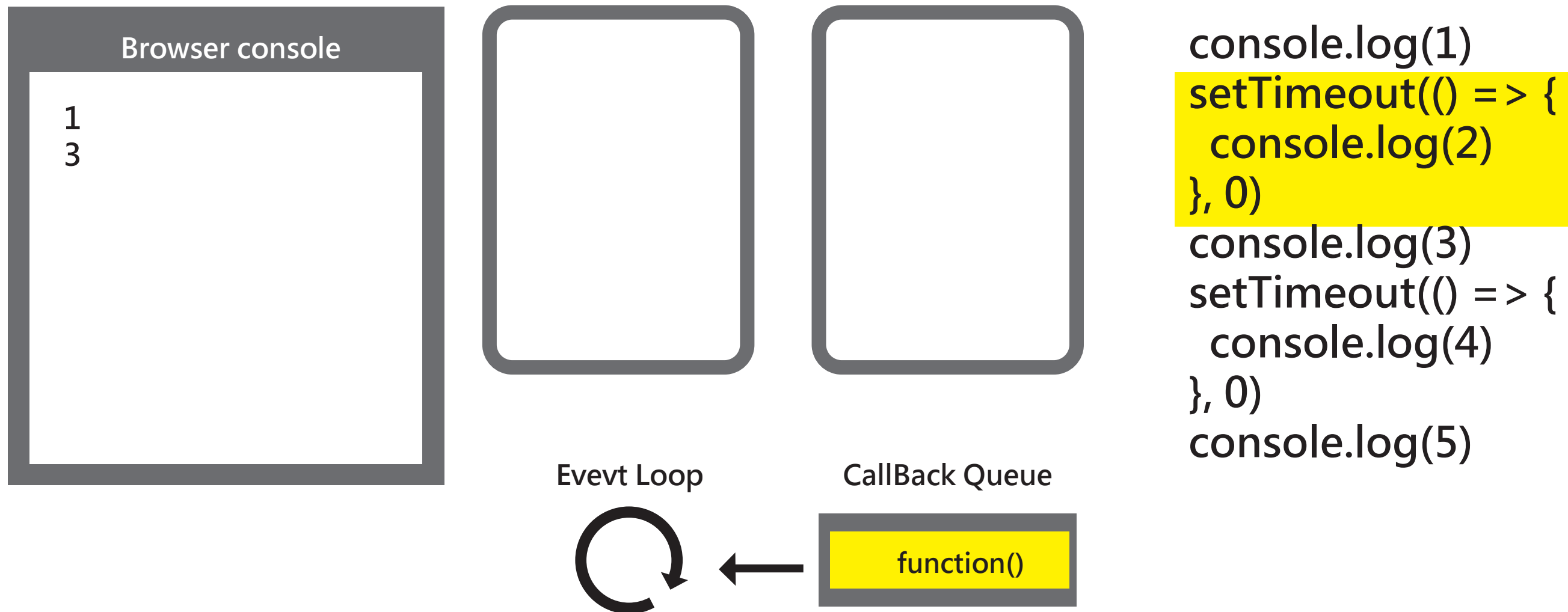
Event Loop



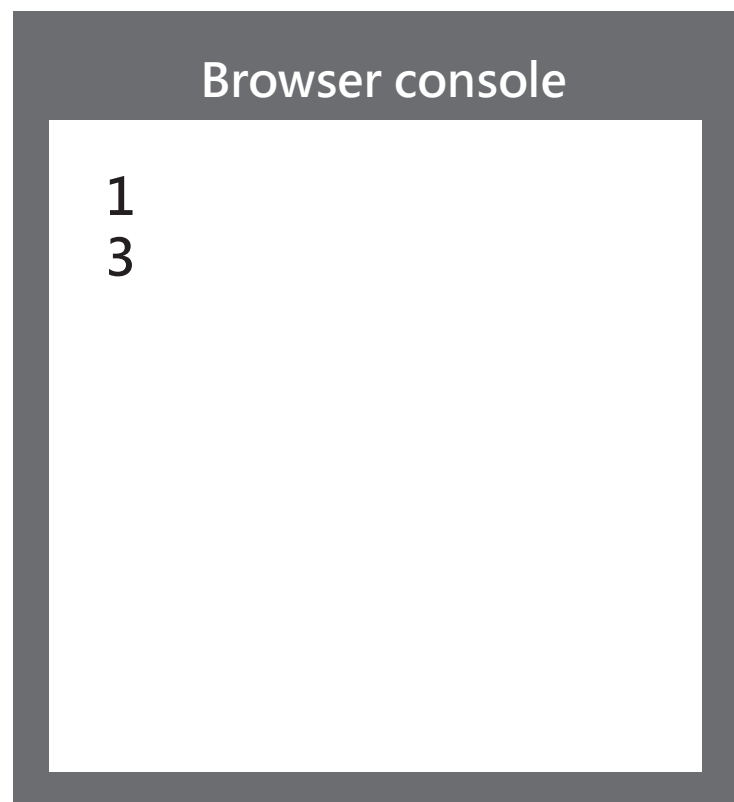
Callback Queue



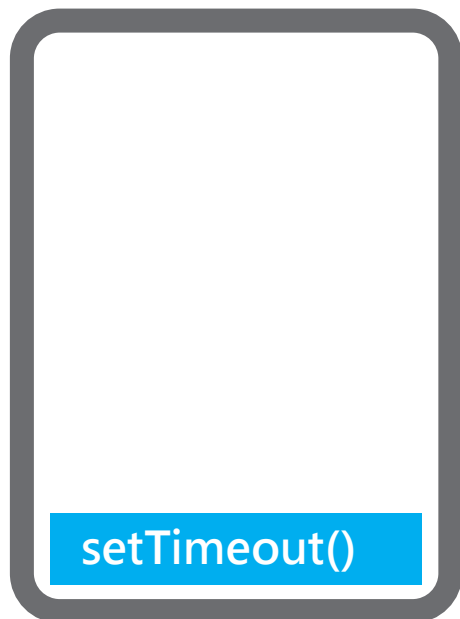
10. 至少 0ms後，定時器執行完成，此時它會將 function() 加入到 CallBack Queue



11. 新增 `setTimeout(() => { console.log(4)}, 0)` 至 Call Stack



Call Stack



Web APIs



Event Loop

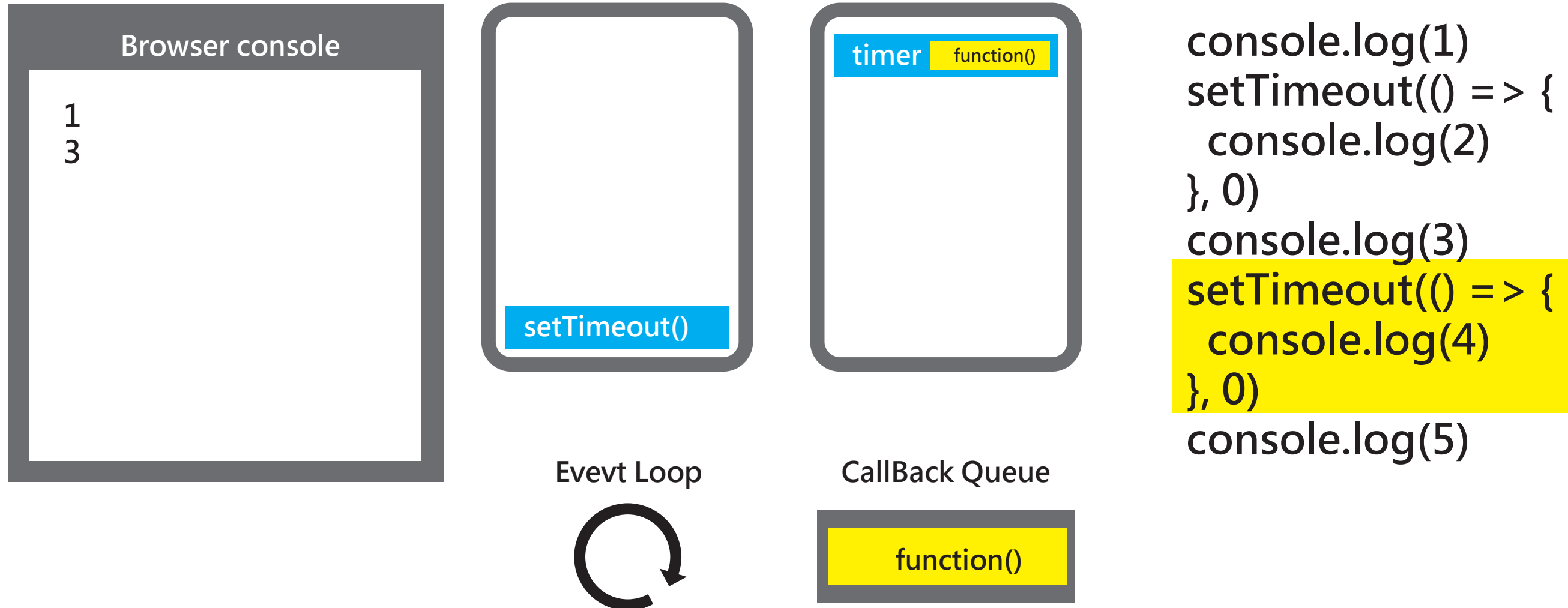


Callback Queue

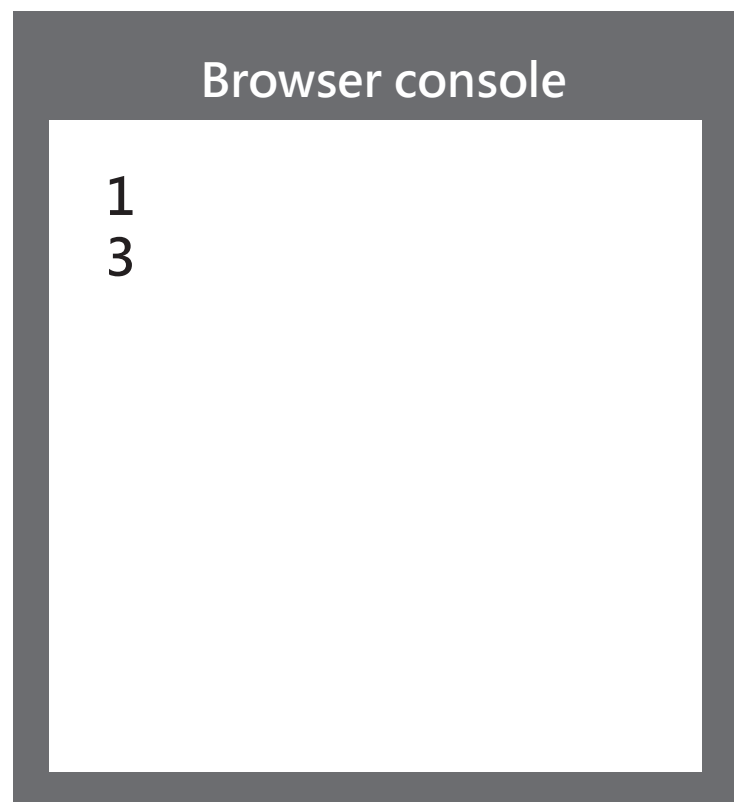


```
console.log(1)  
setTimeout(() => {  
  console.log(2)  
}, 0)  
console.log(3)  
setTimeout(() => {  
  console.log(4)  
}, 0)  
console.log(5)
```

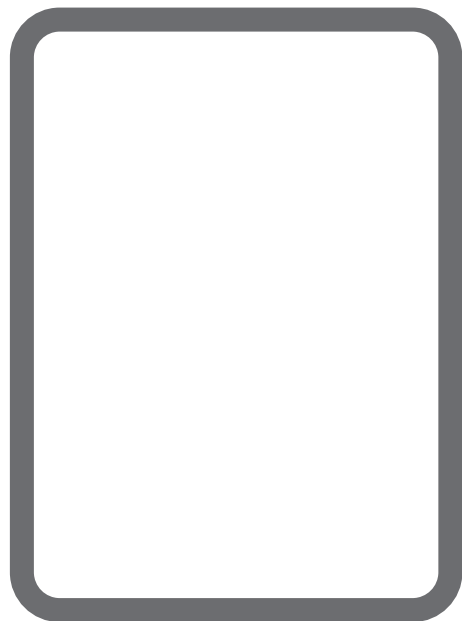
12. `setTimeout(() => { console.log(4)}, 0)`被執行，瀏覽器會根據 web API 建立一個定時器



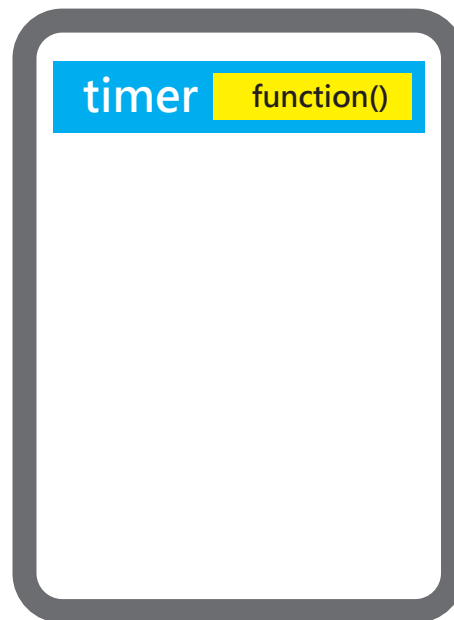
13. setTimeout(() { ... }) 執行完被移除出 Call Stack



Call Stack



Web APIS



Event Loop

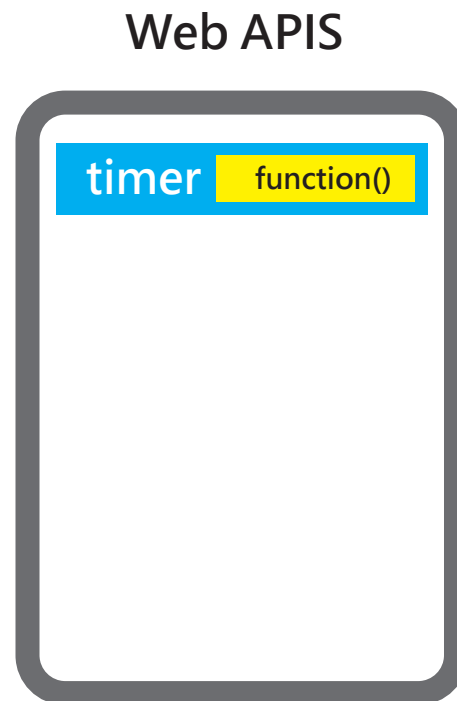
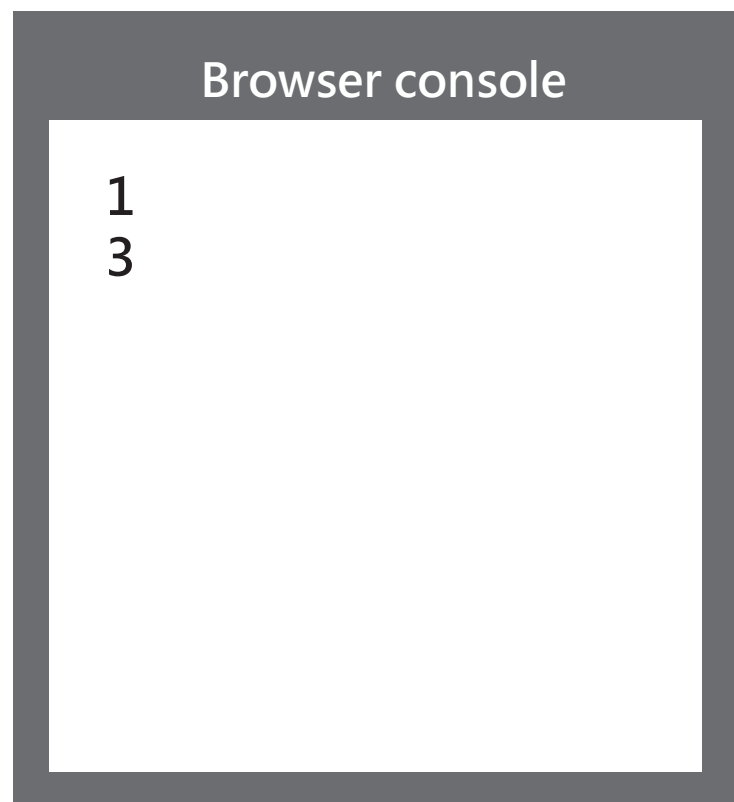


Callback Queue



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

14. 新增 console.log(5) 至 Call Stack



Event Loop

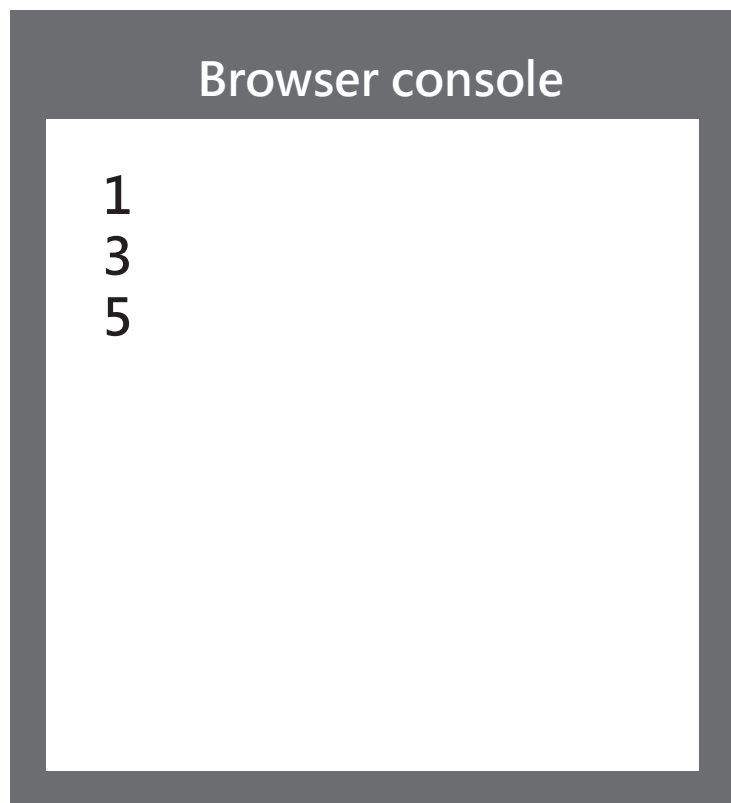


Callback Queue

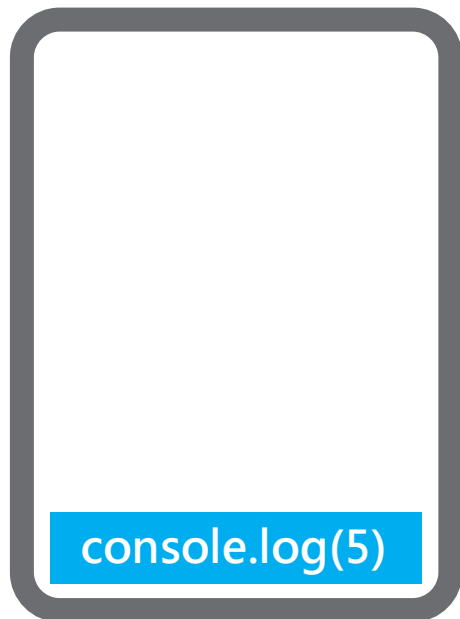


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

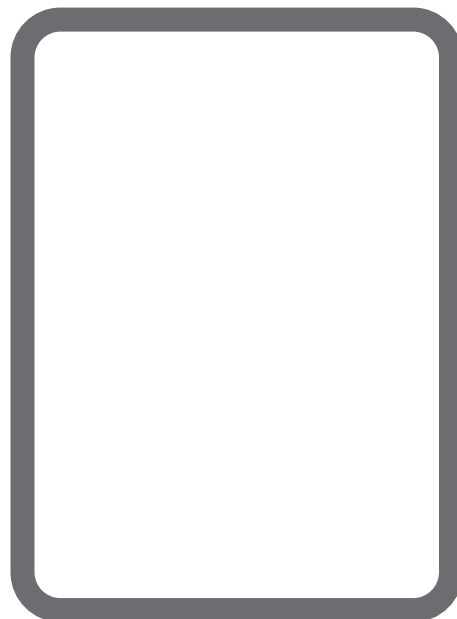
15. `console.log(5)`被執行，0秒後定時器執行完成，此時它會將將 `setTimeout ... console.log(4)` 移到 CallBack Queue



Call Stack



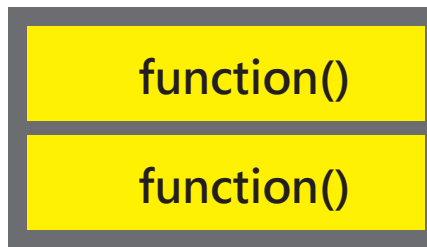
Web APIS



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

`console.log(5)`

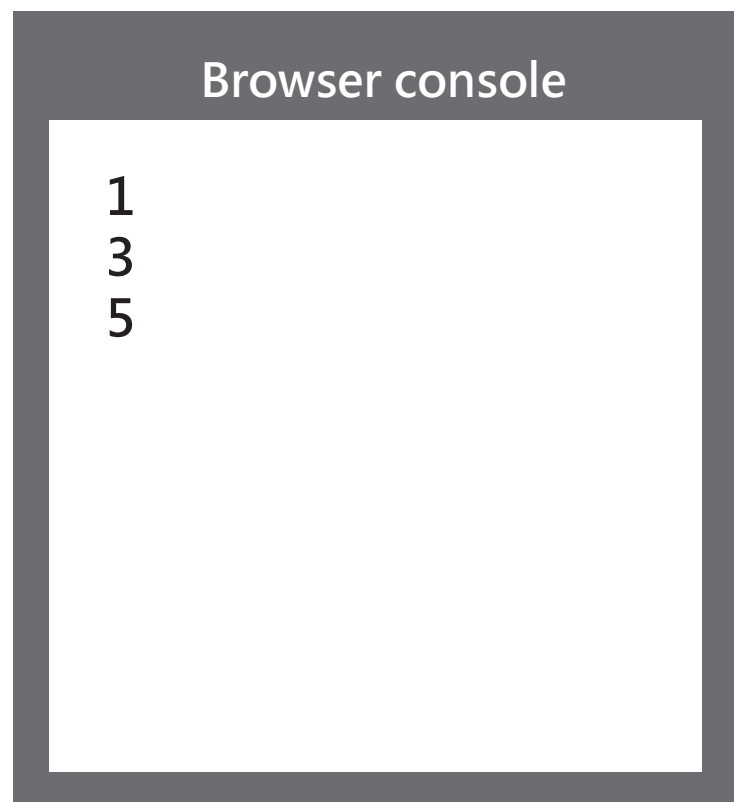
CallBack Queue



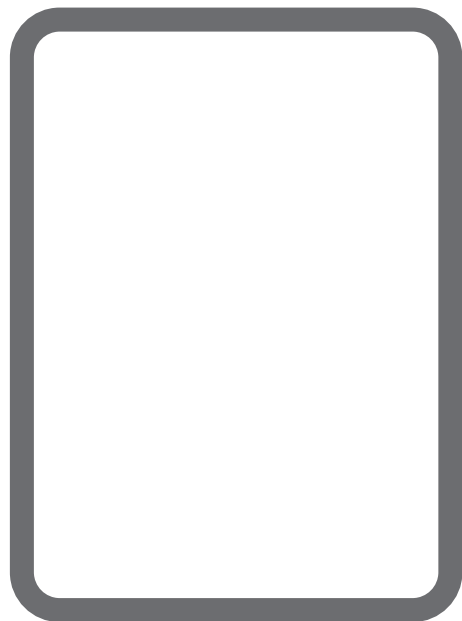
Event Loop



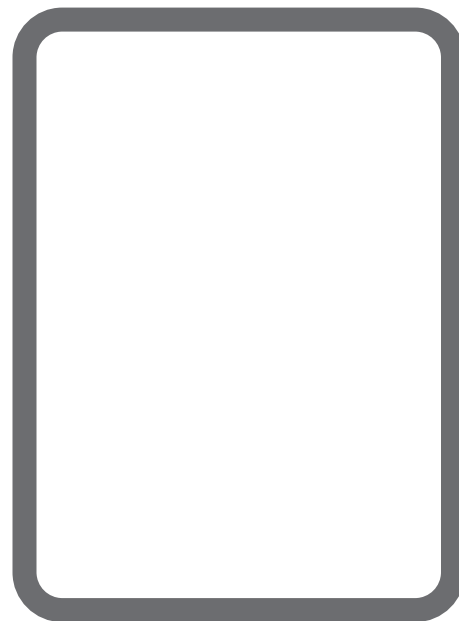
16. console.log(5) 執行完被移除出 Call Stack



Call Stack



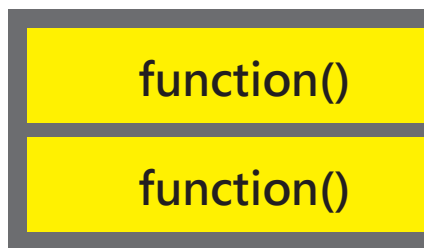
Web APIS



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

console.log(5)

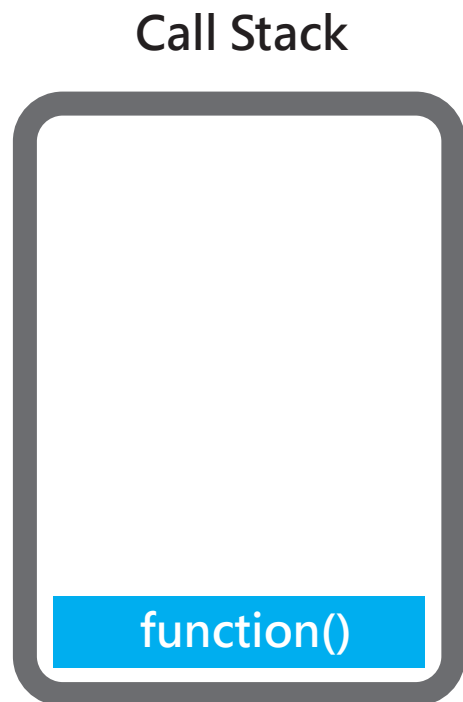
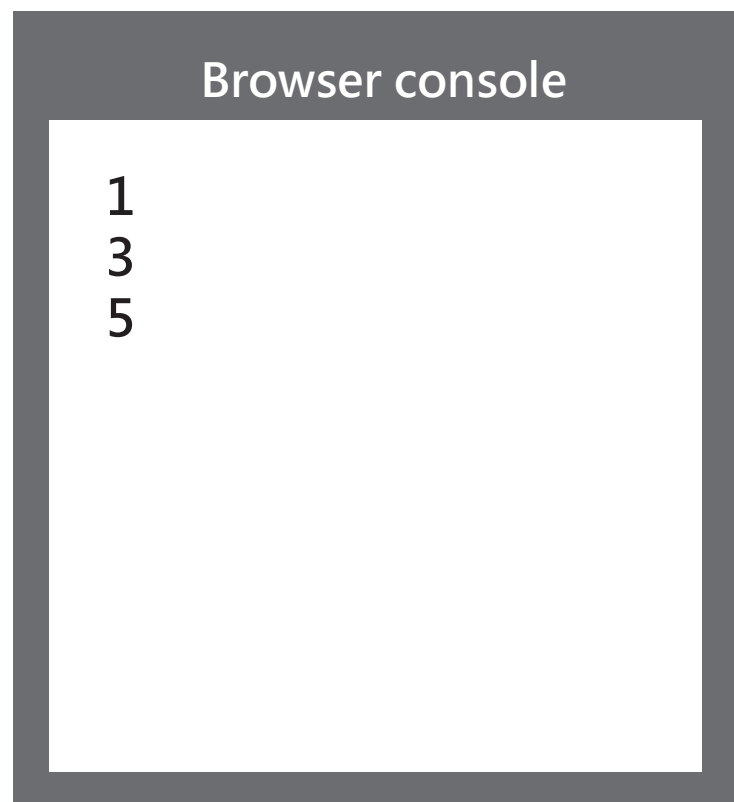
CallBack Queue



Event Loop



17. Event Loop 檢測到此時 Call Stack 為空，將 `setTimeout(() => { console.log(2)}, 0)` 取出壓入到 Call Stack 中



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

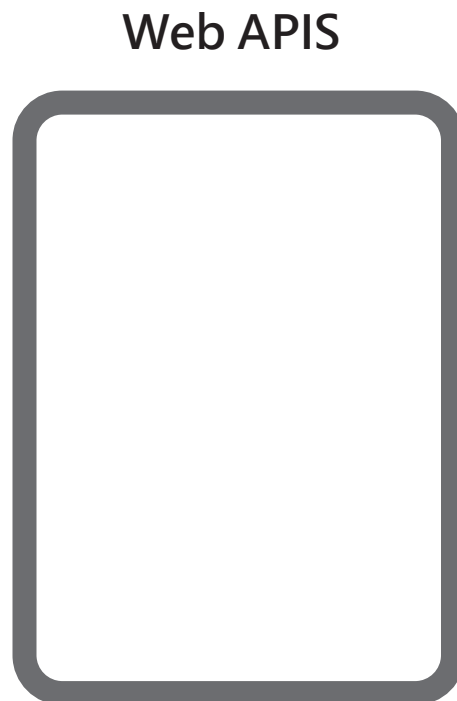
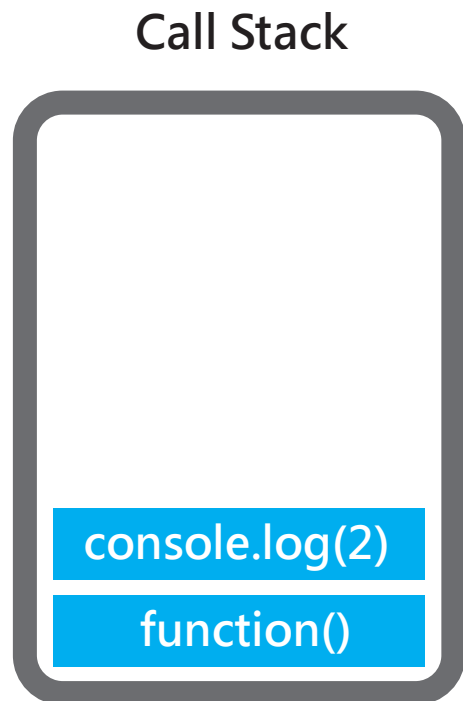
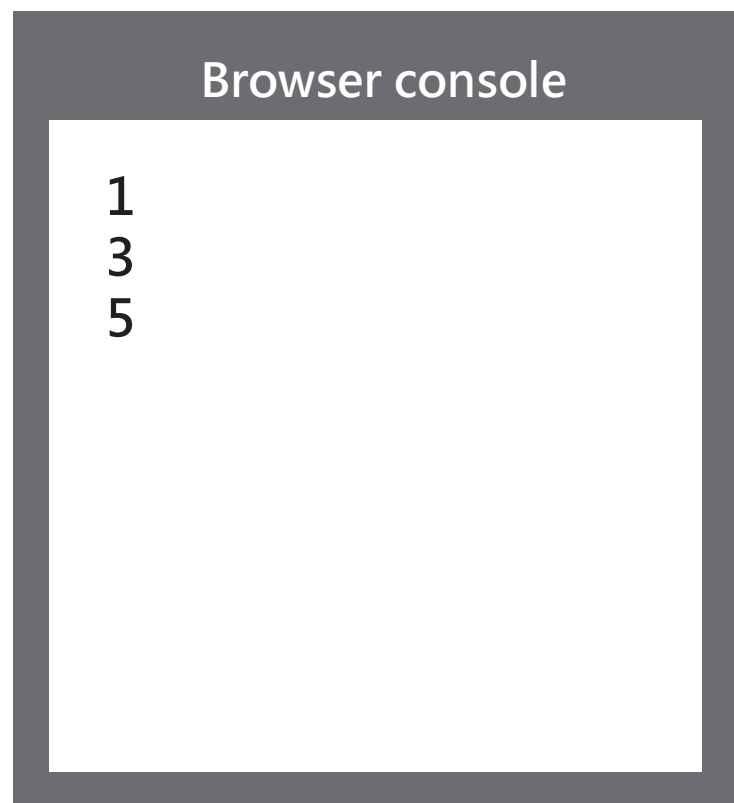
Event Loop



Callback Queue



18. function 被執行,console.log(2)被壓入Call Stack



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

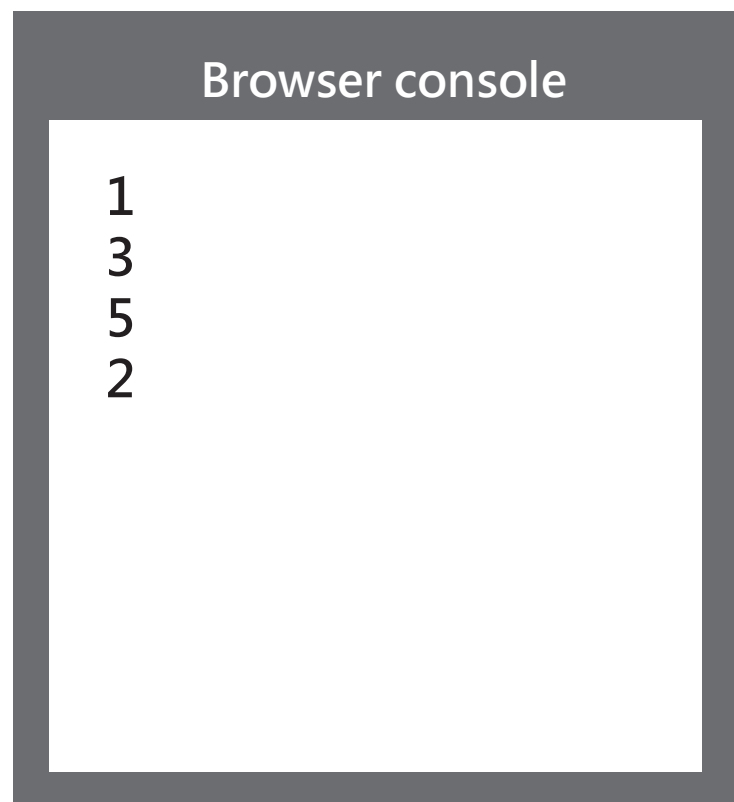
Event Loop



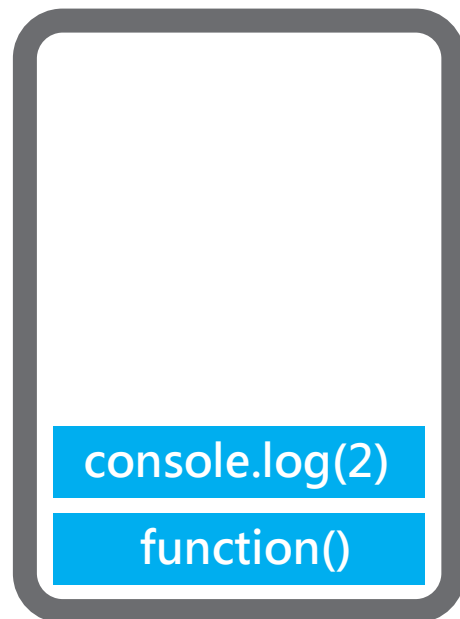
Callback Queue



19. console.log(2)被執行



Call Stack



Web APIS

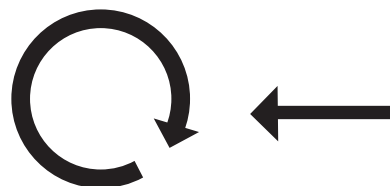


```
console.log(1)
setTimeout(() => {
  console.log(2)
}, 0)
console.log(3)
setTimeout(() => {
  console.log(4)
}, 0)
console.log(5)
```

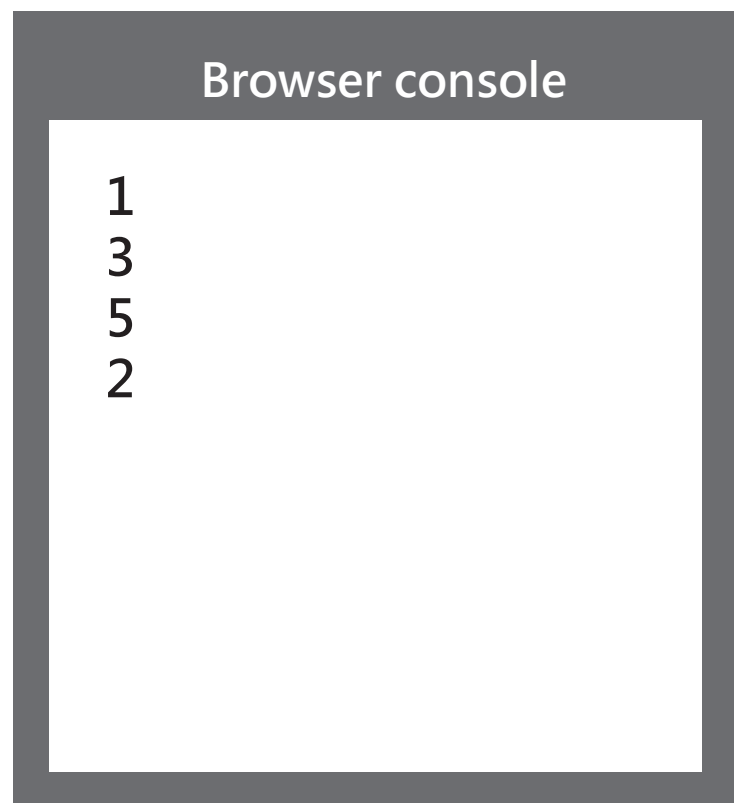
CallBack Queue



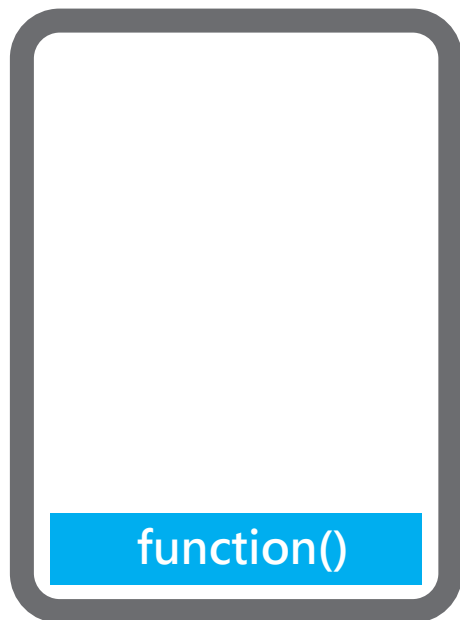
Event Loop



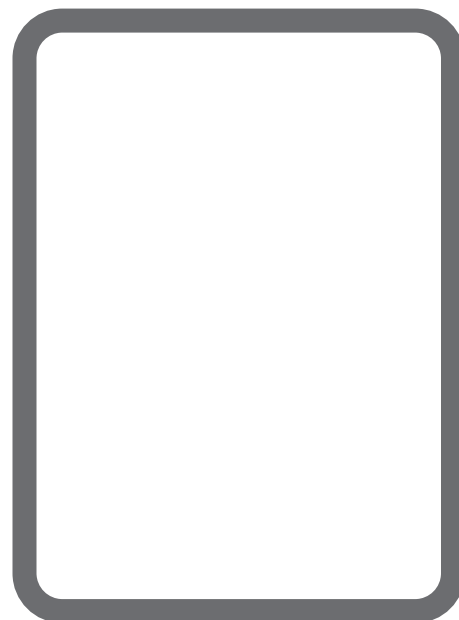
20. console.log(2) 執行完被移除出 Call Stack



Call Stack



Web APIS



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

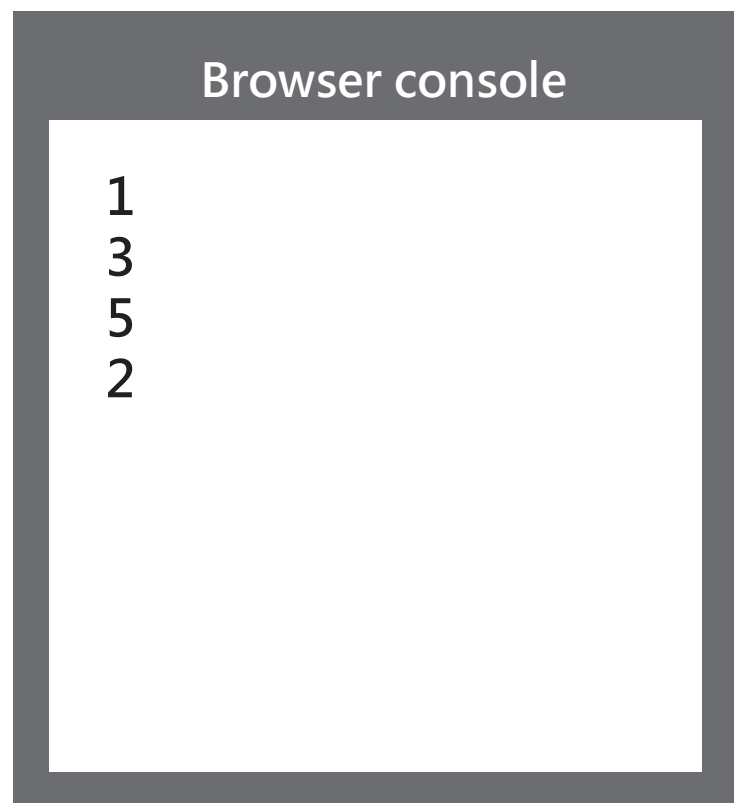
Event Loop



Callback Queue



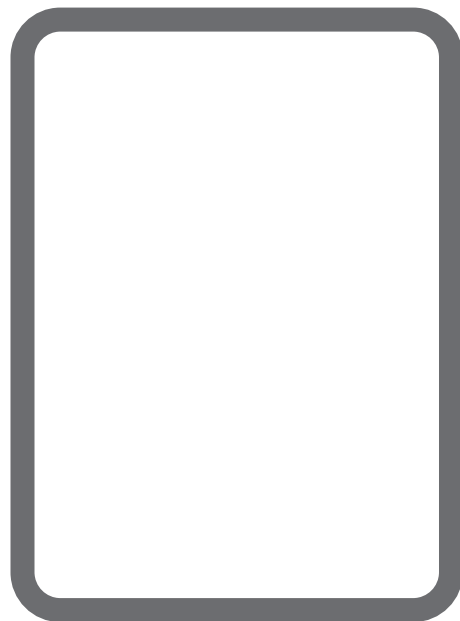
21. function() 執行完被移除出 Call Stack



Call Stack



Web APIS



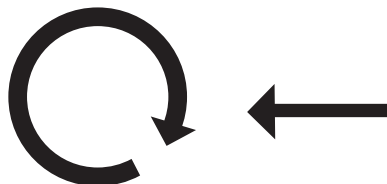
```
console.log(1)  
setTimeout(() => {  
  console.log(2)  
}, 0)
```

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

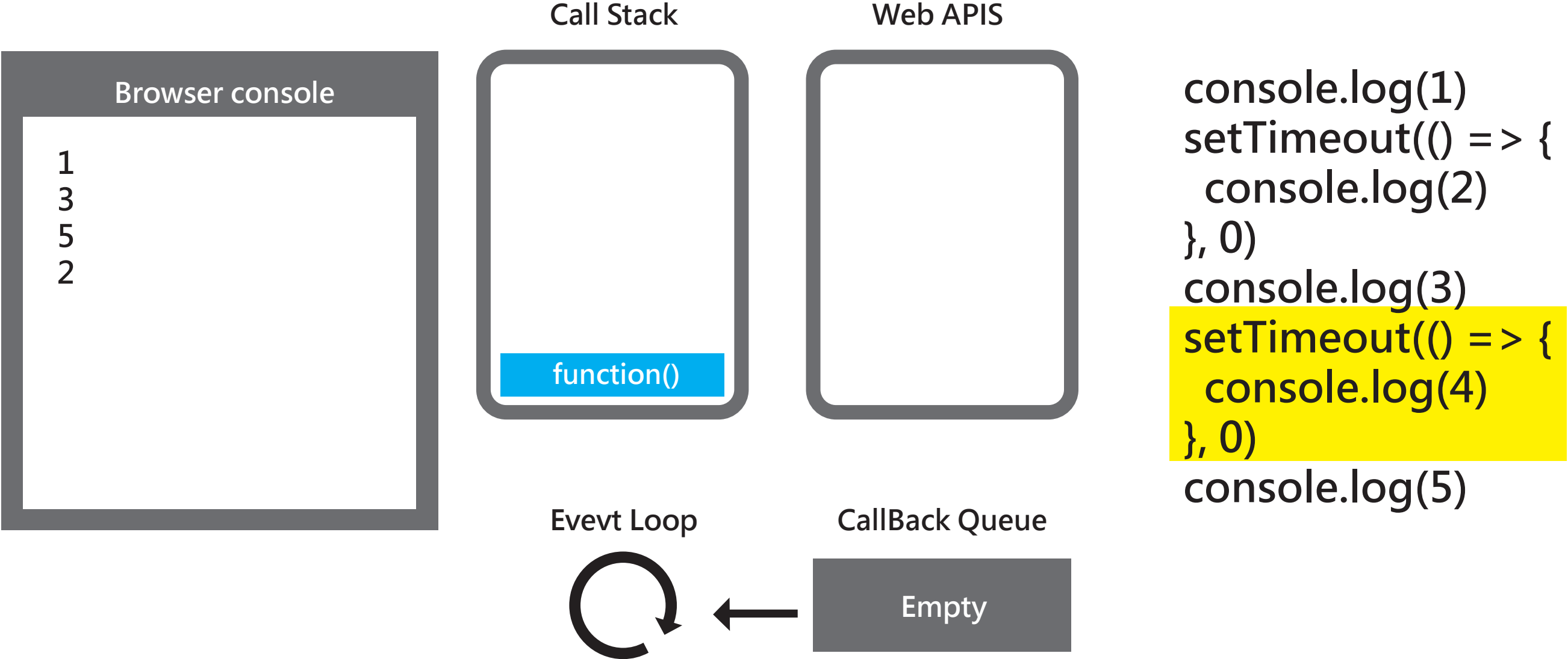
CallBack Queue



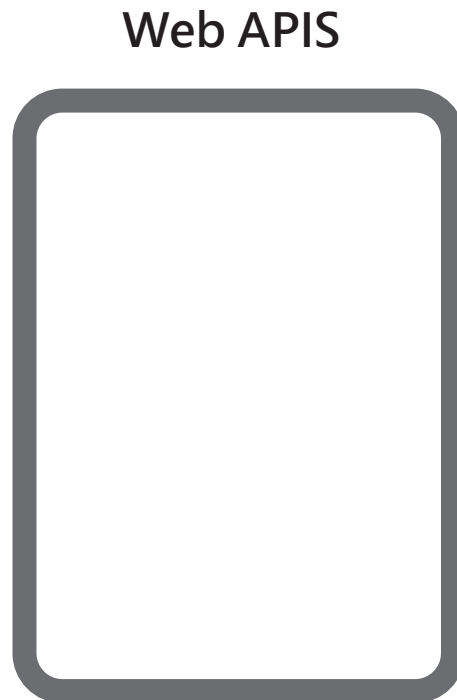
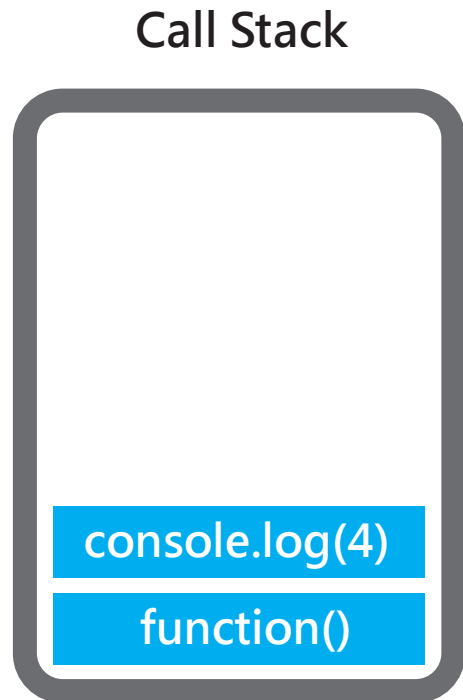
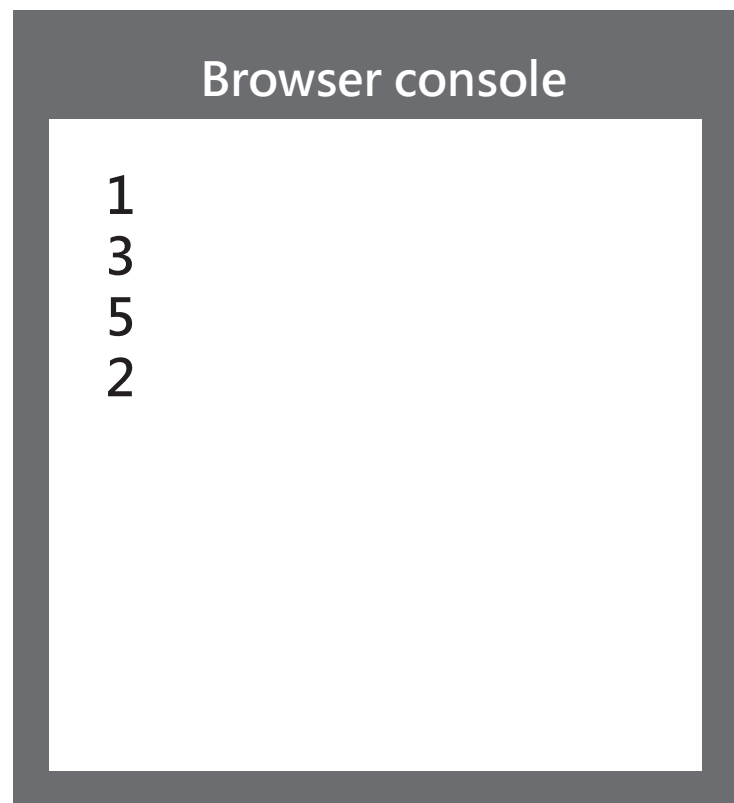
Event Loop



22. Event Loop 檢測到此時 Call Stack 為空，將 `setTimeout(() => { console.log(4)}, 0)` 取出壓入到 Call Stack 中



24. function 被執行,console.log(4)被壓入Call Stack



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

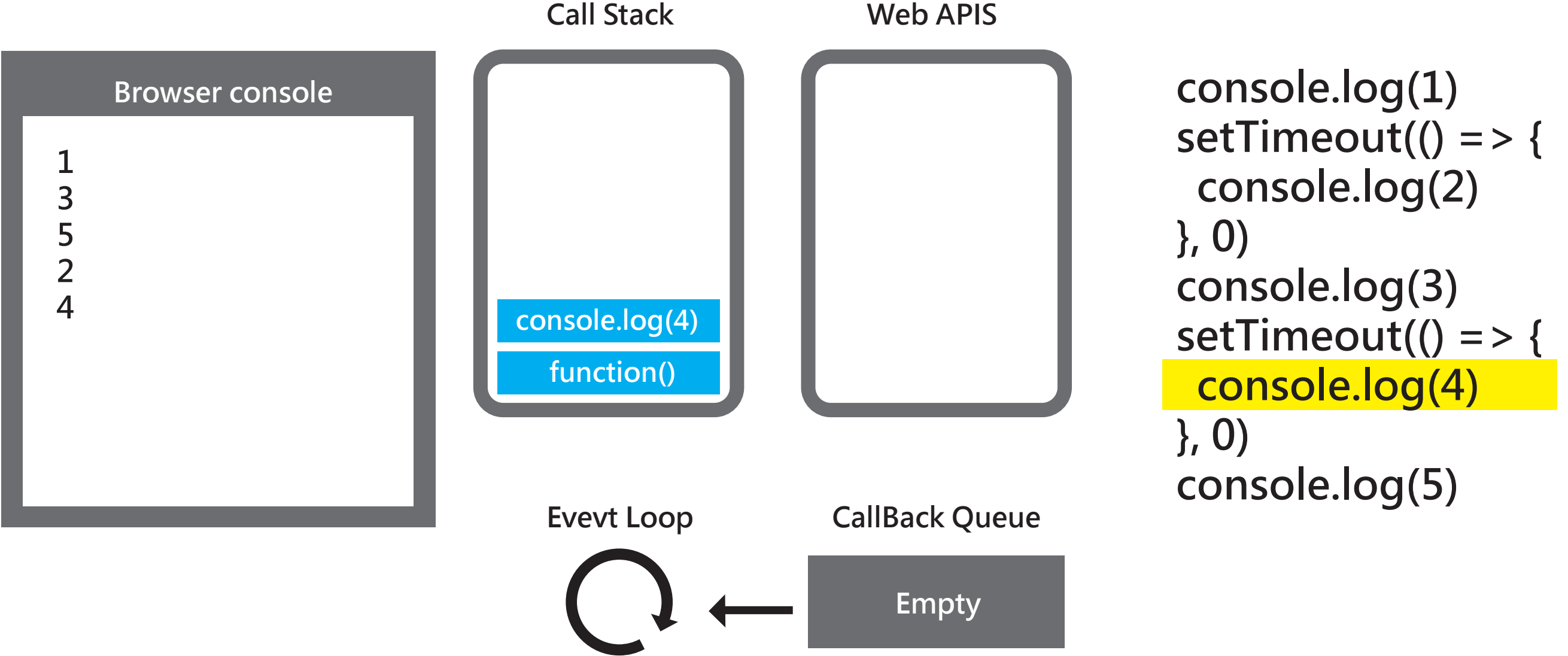
Event Loop



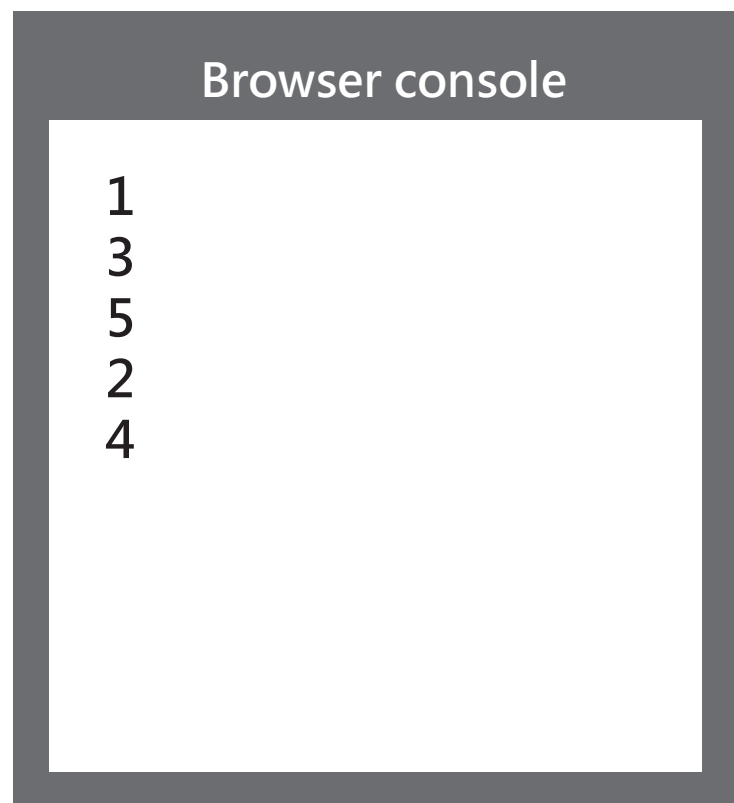
Callback Queue



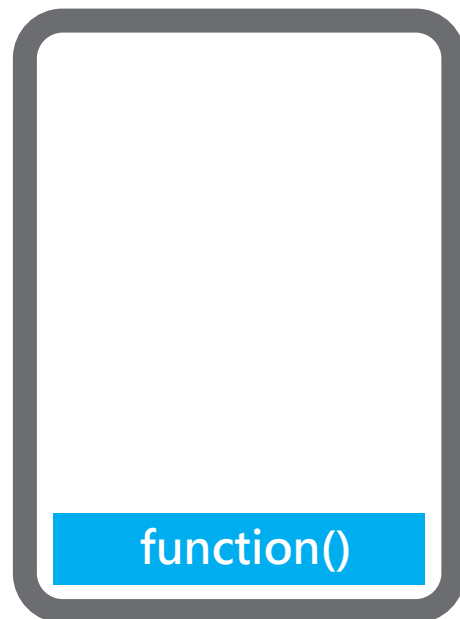
25. console.log(4)被執行



26. console.log(4) 執行完被移除出 Call Stack



Call Stack



Web APIS



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

Event Loop



Callback Queue



27. function() 執行完被移除出 Call Stack

