




```
function compareTimersLists(a, b) {  
  return (a.runAt - b.runAt) || (a.id - b.id);  
}
```

```
#executionQueue = new PriorityQueue(compareTimerLists, setPosition);
```

```
#createTimer(isInterval, callback, delay, ...args) {  
  const timerId = this.#currentTimer++;  
  this.#executionQueue.insert({  
    __proto__: null,  
    id: timerId,  
    callback,  
    runAt: this.#now + delay,  
  })  
}
```









@erickwendel_

```
#executionQueue = new PriorityQueue(compareTimersLists, setPosition)
```

```
function compareTimersLists(a, b) {  
  return (a.runAt - b.runAt) || (a.id - b.id);  
}
```

```
#createTimer(isInterval, callback, delay, ...args) {  
  const timerId = this.#currentTimer++;  
  this.#executionQueue.insert({  
    __proto__: null,  
    id: timerId,  
    callback,  
    runAt: this.#now + delay,  
  })  
}
```



@erickwendel

```
this.#now += time;
let timer = this.#executionQueue.peek();
while (timer) {
  if (timer.runAt > this.#now) break;
  FunctionPrototypeApply(timer.callback, undefined, timer.args);

  this.#executionQueue.shift();

  if (timer.interval) {
    timer.runAt += timer.interval;
    this.#executionQueue.insert(timer);
    return;
  }

  timer = this.#executionQueue.peek();
}
```