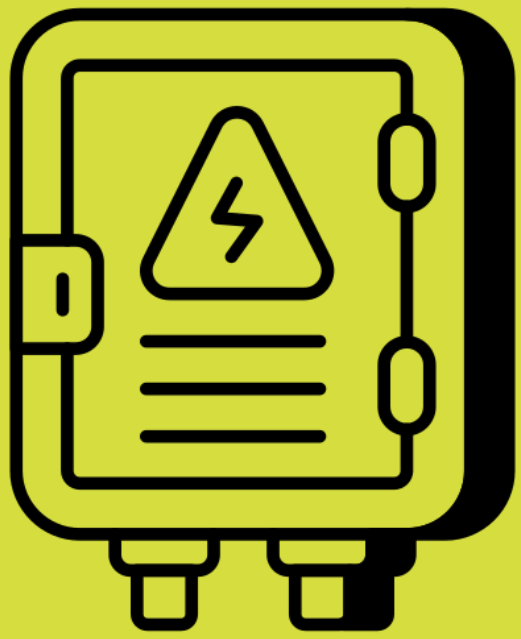
Sridhar Maskeri

Software Engineer



The Circuit Breaker Pattern

Swipe for more

Sridhar Maskeri Software Engineer

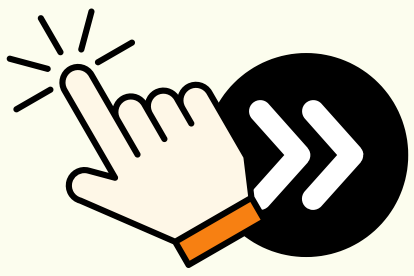
🧯 Circuit Breaker Class

Constructor

```
class CircuitBreaker {
  constructor(action, failLimit = 3, resetTime = 10000) {
    this.action = action
    this.failLimit = failLimit
    this.resetTime = resetTime

    this.failCount = 0
    this.lastFail = null
    this.state = 'CLOSED'
  }
```

Swipe for more



Sridhar Maskeri Software Engineer

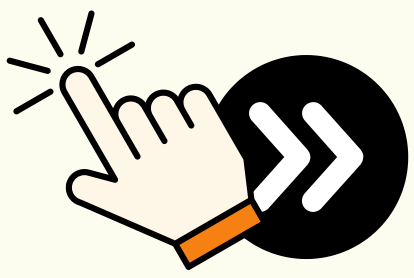
🧯 Circuit Breaker Class

call function

```
async call(...args) {
  if (this.state === 'OPEN') {
    if (Date.now() - this.lastFail > this.resetTime) {
      this.state = 'HALF_OPEN'
    } else {
      throw new Error('Circuit is open. Try later.')
    }
  }

  try {
    const result = await this.action(...args)
    this._reset()
    return result
  } catch (err) {
    this._trip()
    throw err
  }
}
```

Swipe for more



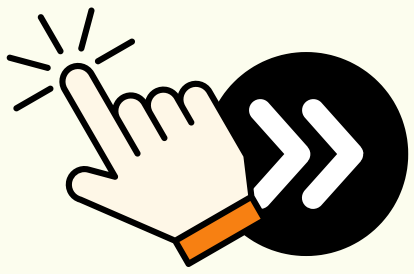
Sridhar Maskeri Software Engineer

🧯 Circuit Breaker Class

reset function

```
  _reset() {
    this.failCount = 0
    this.state = 'CLOSED'
  }
```

Swipe for more



Sridhar Maskeri Software Engineer

🧯 Circuit Breaker Class

trip function

```
_trip() {
    this.failCount++
    this.lastFail = Date.now()
    if (this.failCount >= this.failLimit) {
      this.state = 'OPEN'
    }
  }
}
```

Swipe for more

Sridhar Maskeri

Software Engineer

If you found this helpful, follow me for more dev content.

@sridhar-maskeri