



Microservice

Resilience4J

Resilience4J

Resilience4j is a lightweight, standalone library designed for implementing resilience patterns in Java applications, particularly those using microservices architecture. It provides mechanisms to handle failures gracefully and ensures that services remain responsive under failure conditions. It's Key features are:

- Retry
- Rate Limiter
- Circuit Breaker
- Integration with Spring Boot:

Retry

Resilience4J can help write logic for retry when a service call fails. This retries failed operations a certain number of times before giving up.

```
@Retry(name = "inventoryRetry", fallbackMethod = "fallBackMethodReduceOrder")
```

```
resilience4j:  
  retry:  
    configs:  
      default:  
        maxRetryAttempts: 3  
        waitDuration: 5s
```

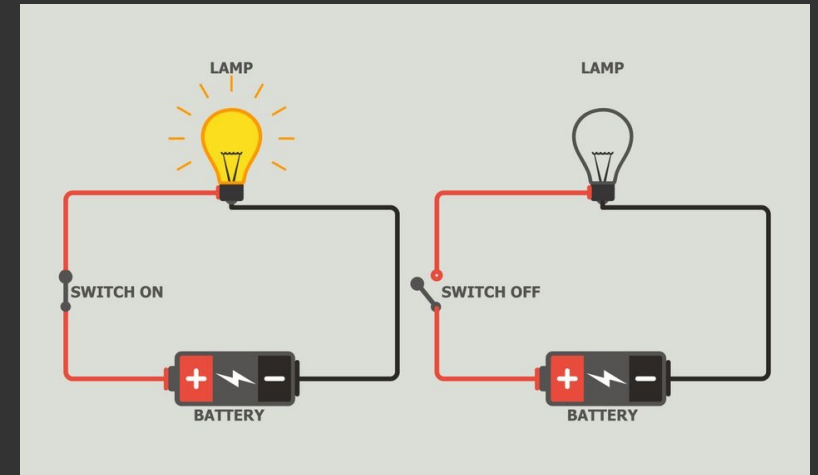
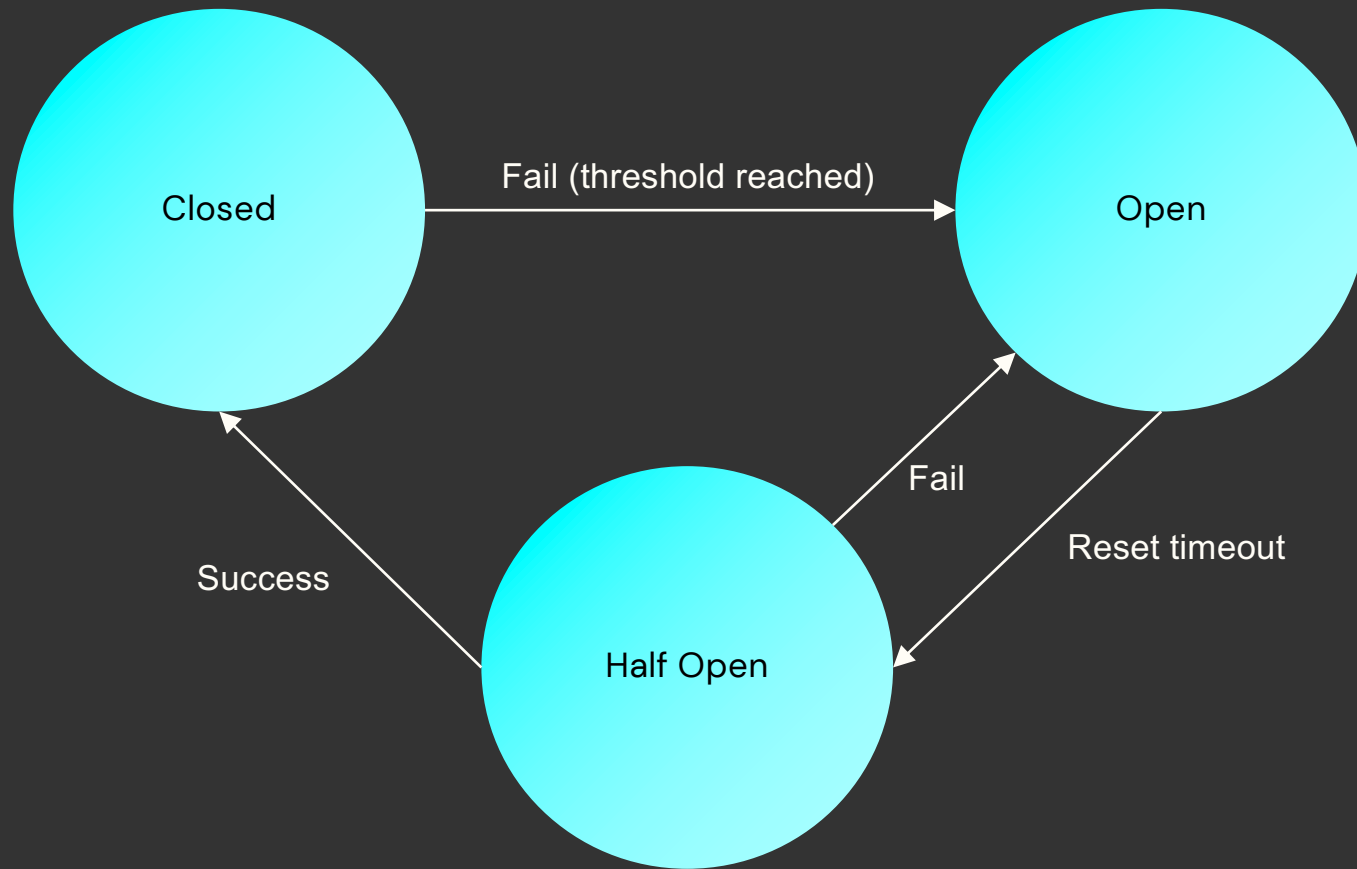
Rate Limiter

Controls the rate of requests to a service by limiting the number of calls allowed during a specific time period.

```
@RateLimiter(name = "inventoryRateLimiter", fallbackMethod =  
"fallBackMethodReduceOrder")
```

```
resilience4j:  
  ratelimiter:  
    configs:  
      default:  
        limitForPeriod: 1          # Max 10 calls in a refresh period  
        limitRefreshPeriod: 5s    # Refresh the limit every second  
        timeoutDuration: 1s       # Time to wait for permission before a request fails
```

Circuit Breaker



Circuit Breaker

The Circuit Breaker pattern helps stop an app from constantly trying something that's likely to fail. It prevents the app from waiting around for the issue to be fixed and saves computer power.

This pattern also lets the app figure out if the problem is gone. If it seems like the problem is fixed, the app can give it another shot.

Circuit Breaker using Resilience4J

```
@CircuitBreaker(name = "inventoryFeignClient",  
                fallbackMethod = "fallBackMethodReduceOrder")
```

```
resilience4j:  
  circuitbreaker:  
    instances:  
      inventoryFeignClient:  
        register-health-indicator: true  
        sliding-window-size: 10  
        failure-rate-threshold: 50  
        wait-duration-in-open-state: 10000  
        permitted-number-of-calls-in-half-open-state: 5  
        sliding-window-type: COUNT_BASED  
        minimum-number-of-calls: 5  
        automatic-transition-from-open-to-half-open-enabled: true
```

