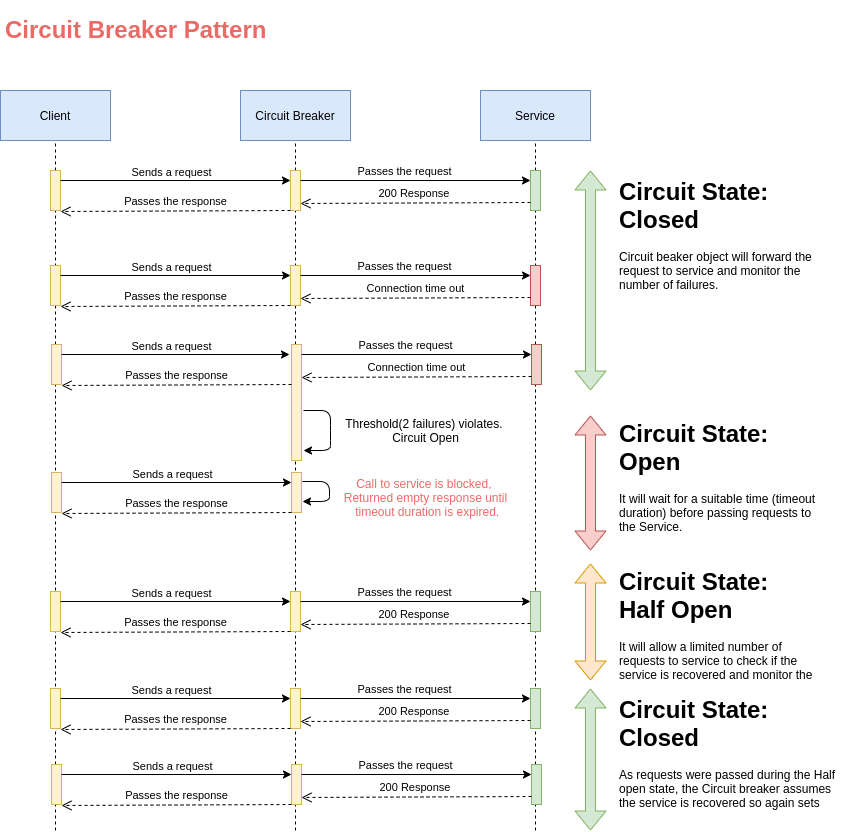
# Circuit Breaker Pattern

Prevents a network or service failure from cascading to other services.

* Prevents an application from repeatedly trying to execute an operation which is likely to fail.
* Function execution call is wrapped in a circuit breaker object, which monitors the failures.
* If the failure rate (average number of failed requests) is above a specified threshold, it will block subsequent requests to pass to actual service. So will return empty object immediately.
* After a suitable time (timeout period), it allows limited number of requests to service to check whether the service is recovered from the transient failure.
* If those requests succeed the circuit breaker resumes normal operation. Otherwise, if there is a failure the timeout period begins again.

Following diagram explain above steps.



# Retry Pattern

Detects, handles failures efficiently with the following strategies:

## Cancel

If the error will reproduce on retry, we prefer to cancel the retry. For example, when response is about Invalid credentials, it will fail every time.

## Retry

If the fault is rare, it could be due to unusual circumstances like network package becomes corrupted. So, we can retry immediately.

## Retry after delay

If the fault is transient or its due to busy failures, the network or service might need a short period while the connectivity issues are corrected or the backlog of work is cleared. In this case, we wait for some time and retry.

