

# Application Integration: SQS

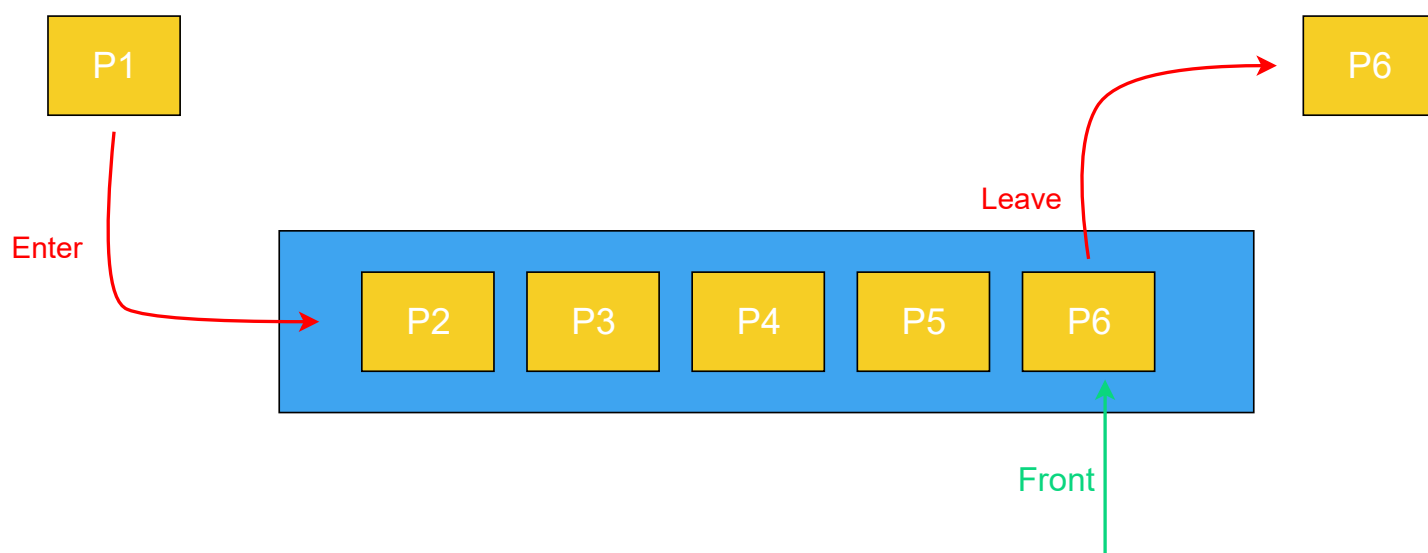
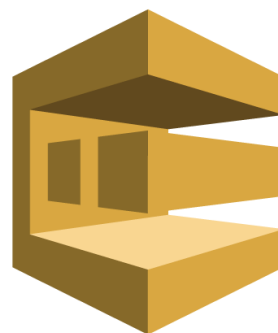
SQS: A queue in the cloud and its features are discussed in this lesson.

## We'll cover the following

- Zero capacity management

## A queue in the cloud

SQS is a highly-durable queue in the cloud. You put messages on one end, and a consumer takes them out from the other side. The messages are consumed in almost first-in-first-out order, but the ordering is not strict. The lack of strict ordering happens because your SQS queue is actually a bunch of queues behind the scenes. When you enqueue a message, it goes to a random queue, and when you poll, you also poll a random queue. In addition, duplicate messages can emerge within SQS, so your consumers should be prepared to handle this situation.



Queue of different processes

# Zero capacity management #

Like S3, SQS is one of the few AWS services that requires zero capacity management.

- There is no limit on the rate of messages enqueued or consumed, and you don't have to worry about any throttling limits.
- The number of messages stored in SQS (the backlog size) is also unlimited.

As long as you can tolerate the lack of strict ordering and the possibility of duplicates, this property makes SQS a great default choice for dispatching asynchronous work.

If you really need strict ordering and exactly-once delivery (no duplicates):

- SQS has an option to enable this property by marking your queue as FIFO.
- But these FIFO queues come with a throughput limit of 300 messages per second, so they're only viable if you're certain that your message rate will remain well clear of that limit.



In SQS, there is no strict ordering because SQS queue is actually a bunch of queues behind the scenes and duplicate messages can emerge within SQS too.

[Retake Quiz](#)

---

In the next lesson, we will take a look at Kinesis and it's different features.