

**NAME**

"Future::Queue" – a FIFO queue of values that uses Futures

**SYNOPSIS**

```
use Future::Queue;

my $queue = Future::Queue->new;

my $f = repeat {
    $queue->shift->then(sub {
        my ( $thing ) = @_;
        ...
    });
};

$queue->push( "a thing" );
```

**DESCRIPTION**

Objects in this class provide a simple FIFO queue the stores arbitrary perl values. Values may be added into the queue using the “push” method, and retrieved from it using the “shift” method.

Values may be stored within the queue object for `shift` to retrieve later, or if the queue is empty then the future that `shift` returns will be completed once an item becomes available.

**CONSTRUCTOR****new**

```
$queue = Future::Queue->new
```

Returns a new `Future::Queue` instance.

**push**

```
$queue->push( $item )
```

Adds a new item into the queue. If the queue was previously empty and there is at least one `shift` future waiting, then the next one will be completed by this method.

**shift**

```
$item = $queue->shift->get
```

Returns a `Future` that will yield the next item from the queue. If there is already an item then this will be taken and the returned future will be immediate. If not, then the returned future will be pending, and the next `push` method will complete it.

**AUTHOR**

Paul Evans <leonerd@leonerd.org.uk>