

Test::Mojo

```
use Test::Mojo;
```

```
use Test::More;
```

```
my $t = Test::Mojo->new;
```

```
$t->get_ok( 'http://mojolicious.org' )
```

```
    ->status_is(200)
```

```
    ->text_like( 'div#wrapperlicious h1' => qr/Perl/ );
```

```
$t->get_ok( 'https://api.twilio.com/2010-04-01.json' )
```

```
    ->status_is(200)
```

```
    ->json_is( '/subresource_uris/accounts' => "/2010-04-01/Accounts.json" );
```

```
done_testing();
```

```
get_ok('http://mojolicious.org')
```

```
get_ok('https://api.twilio.com/2010-04-01.json')
```

status_is(200)

status_is(200)

```
text_like('div#wrapperlicious h1' => qr/Perl/);
```



```
json_is('/source_unis/accounts'=>'/2010-04-01/Accounts.json');
```

Test::Mojo

```
use Test::Mojo;  
use Test::More;
```

```
my $t = Test::Mojo->new;
```

```
$t->get_ok( 'http://mojolicious.org' )  
    ->status_is(200)  
    ->text_like( 'div#wrapperlicious h1' => qr/Perl/ );
```

```
$t->get_ok( 'https://api.twilio.com/2010-04-01.json' )  
    ->status_is(200)  
    ->json_is( '/subresource_uris/accounts' => "/2010-04-01/Accounts.json" );
```

```
done_testing();
```

Plugins

```
package Mojolicious::Plugin::TypeCast;
use Mojo::Base 'Mojolicious::Plugin';

use Scalar::Util 'looks_like_number';

sub register {
    my ($self, $app, $conf) = @_;

    $app->helper(
        to_integer => sub {
            my ($self, $hash, @keys) = @_;

            for my $key (@keys) {
                next unless exists $hash->{$key};
                $hash->{$key} ||= 0; ## empty, undef are ok
                next unless looks_like_number($hash->{$key});
                $hash->{$key} += 0; ## convert to number
            }
        }
    );
}

1;
```

```
#!/usr/bin/env perl
use Mojolicious::Lite;
use lib 'lib';

plugin 'TypeCast';

post '/' => sub {
    my $c = shift;
    my $json = $c->req->json;

    $c->to_integer($json, qw/foo bar/);

    $c->render(json => $json);
};

app->start;
```