


```
use Mojolicious::Lite;
use Mojolicious::Plugin::OpenAPI;
use Time::Piece ();

get '/time/:tz' => sub {
    my $c = shift->openapi->valid_input or return;

    local $ENV{TZ} = $c->param('tz');
    my $time = Time::Piece->new;
    $c->render(openapi => { datetime => $time->datetime, epoch => $time->epoch });
}, "time-of-day";

get '/temperature/:lat/:lon' => sub {
    my $c = shift->openapi->valid_input or return;

    $c->render(openapi => { fahrenheit => "80°F", celsius => "26.667°C", kelvin => "299.82°K" });
}, "current-temperature";

post '/temperature/:lat/:lon' => sub {
    my $c = shift->openapi->valid_input or return;

    $c->render(openapi => { message => "ok" });
}, "set-current-temperature";

app->plugin(OpenAPI => { url => app->home->rel_file('schema.yaml') });

app->start;
```

use Mojo::Inc::Pug::OpenAPI;

get : /time/ : tz :

```
get '/temperature/:lat/:lon'
```

post '//temperature//:lat//:lon'

```
app->plugin(openAPI => {url => app->home->url_file('schema.yaml')});
```

```
my $c = shift->openapi->valid_input or return;
```

```
local $ENV{TZ} = $c->param( 'tz' );
```

```
my $time = Time::Piece->new;
```

```
$c->render(openapi => { datetime => $time->datetime, epoch => $time->epoch } );
```



```
my$c => shift -> openapi -> valid_input or return;
```

```
my$c => shift -> openapi -> valid_input or return;
```

```
my$c => shift -> openapi -> valid_input or return;
```

```
$c-render(openapi => { datime => $time->datetime, epoch => $time->epoch });
```

```
$c->render(openapi=>{fahrenheit=>'80°F', celsius=>'26.667°C', kelvin=>'299.82°K'})
```

```
$c->render(openapi => { message => "ok" });
```

```
use Mojolicious::Lite;
use Mojolicious::Plugin::OpenAPI;
use Time::Piece ();

get '/time/:tz' => sub {
    my $c = shift->openapi->valid_input or return;

    local $ENV{TZ} = $c->param('tz');
    my $time = Time::Piece->new;
    $c->render(openapi => { datetime => $time->datetime, epoch => $time->epoch });
}, "time-of-day";

get '/temperature/:lat/:lon' => sub {
    my $c = shift->openapi->valid_input or return;

    $c->render(openapi => { fahrenheit => "80°F", celsius => "26.667°C", kelvin => "299.82°K" });
}, "current-temperature";

post '/temperature/:lat/:lon' => sub {
    my $c = shift->openapi->valid_input or return;

    $c->render(openapi => { message => "ok" });
}, "set-current-temperature";

app->plugin(OpenAPI => { url => app->home->rel_file('schema.yaml') });

app->start;
```

```
paths:
  /time/{tz}:
    get:
      description: "Current time"
      x-mojo-name: "time-of-day"
      parameters:
        - name: tz
          in: path
          required: true
          description: time zone
          type: string
          enum:
            - "PST8PDT"
            - "MST7MDT"
            - "CST6CDT"
            - "EST5EDT"
      responses:
        200:
          description: "successful response"
          schema:
            type: object
            properties:
              datetime:
                type: string
              epoch:
                type: integer
        400:
          description: "Invalid timezone"
```