```perl
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;
```

"time-of-day"

```
"current-temperature"
```

```
"set-current-temperature"
```

```perl
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;
```

JSON Schema References

- Allows us to define schemas and inputs by name
- Less duplication, less room for error