

# Mojo VS Catalyst

Errietta Kostala

Blame mst

# Routing - Catalyst

```
sub hello : Local { }
```

```
sub hi : Path('hello') { }
```

# Routing - Catalyst

```
sub base :Chained("/") :PathPart("account") :CaptureArgs(0) {}
```

```
#/account/*
```

```
sub id :Chained("base") :PathPart("") :CaptureArgs(1) {
```

```
    my ($self, $c, $id) = @_;
```

```
}
```

```
#/account/*/edit
```

```
sub edit :Chained("id") :PathPart("edit") :Args(0) { }
```

# Routing - Mojo

```
my $r = $app->routes;
```

```
$r->any( '/account/:id/edit' )->to('account#edit');
```

```
sub edit {
```

```
    my $self = shift;
```

```
    my $id = $self->param('id');
```

```
}
```

# Request handling - Catalyst

```
my $foo = $c->req->query_params->{foo};
```

```
my $foo = $c->req->body_params->{foo};
```

```
my $foo = $c->req->params->{foo};
```

# Request handling - Mojo

```
my $foo = $c->req->query_params->param('foo');
```

```
my $foo = $c->req->body_params->param('foo');
```

```
my $foo = $c->req->param('foo');
```

# Templating - Catalyst

```
[% FOREACH user IN users - %]
```

```
    <li>[% user.name %]</li>
```

```
[% END %]
```



# Templating - Mojo

```
% foreach my $user (@users) {  
    <li>  
        %= user->name  
    </li>  
% }
```

# Databases - Catalyst

```
package MyApp::Model::DB;

use strict;

use base 'Catalyst::Model::DBIC::Schema;

__PACKAGE__->config(schema_class => 'MyApp::Schema',
    connect_info => { dsn => 'dbi:sqlite:hello', });

$c->model('DB')->resultset('Users')->find(1);
```

# Databases - Mojo

```
use My::Schema;
```

```
helper db => sub {
```

```
    return My::Schema->  
    connect('dbi:SQLite:mydb');
```

```
};
```

```
..
```

```
my $user = $self->db->resultset('Users')->find(1);
```

# JSON - Catalyst

```
package MyApp::View::JSON;

use base qw(Catalyst::View::JSON);

1;

..

$c->stash->{message} = "Hello world";

$c->forward('View::JSON');
```

# JSON - Catalyst

```
'application/json' => sub {  
    Class::Load::load_first_existing_class('JSON::MaybeXS',  
    'JSON')  
    ->can('decode_json')->(do { local $/; $_->getline });  
}  
  
...  
  
my $parsed_json = $c->req->body_data;
```

# JSON - Mojo

```
my $hash = $c->req->json;
```

```
$c->render(json => $hash);
```

# Testing - Catalyst

```
my $ua = Test::WWW::Mechanize::Catalyst;
```

```
$ua->get_ok('/foo');
```

```
is_ok $ua->response->code, 200;
```

```
$ua->content_is('bar');
```

# Testing - Catalyst

```
$ua->get_ok('/userids');
```

```
my @ids = decode_json($ua->content);
```

```
is_deeply \@ids, [1, 2, 3];
```



# Testing - Mojo

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

```
$t->json_is('/userids' => [1, 2, 3]);
```

```
$t->get_ok('/foo')->status_is(200)->content_is('bar');
```

# Thank you

<http://www.errietta.me/>