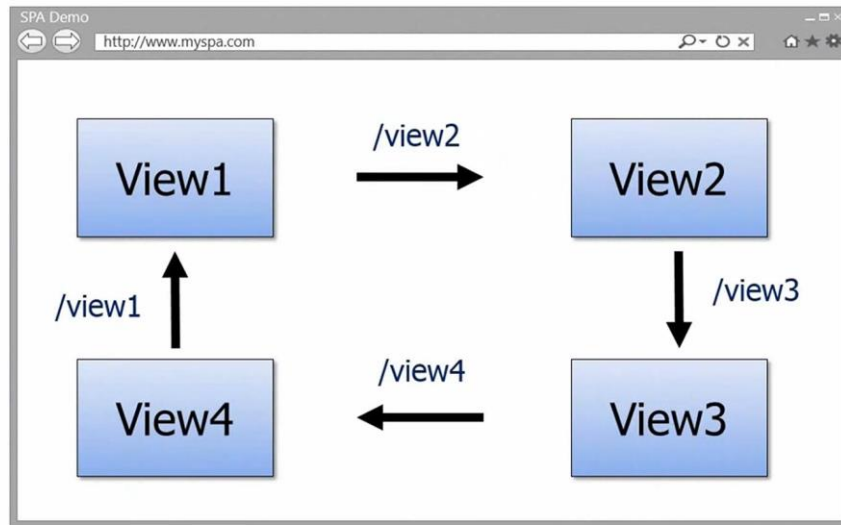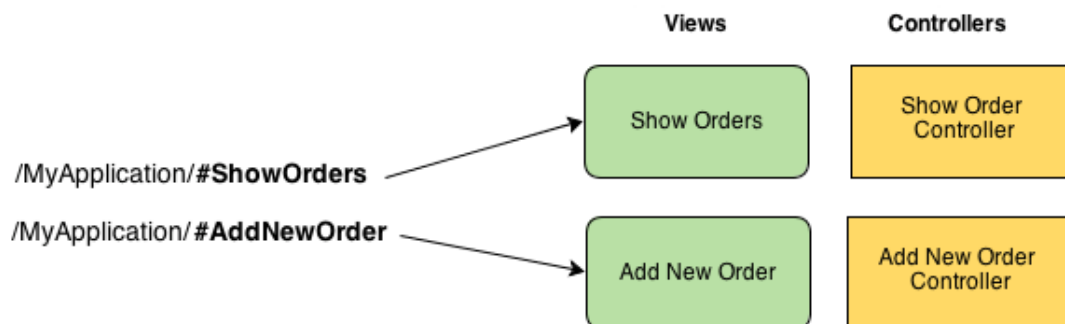# Routing

The Role of Routes

Routing helps you in dividing your application in logical views and bind different views to Controllers.

Introduction to $routeProvider
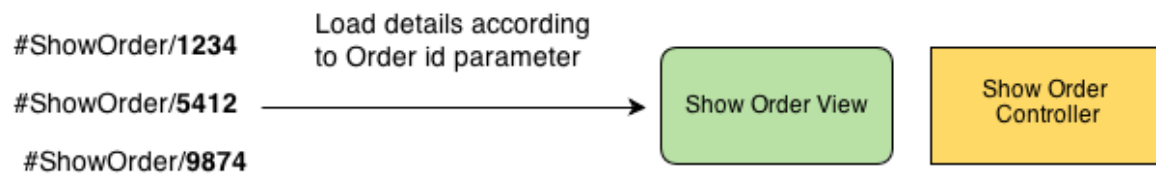
Application routes in Angular are declared via the `$routeProvider`, which is the provider of the $route service. This service makes it easy to wire together controllers, view templates, and the current URL location in the browser. Using this feature we can

implement deep linking, which lets us utilize the browser's history (back and forward navigation) and bookmarks.

Syntax to add Routing

Below is the syntax to add routing and views information to an angular application. We defined an angular app "sampleApp" using angular.module method. Once we have our app, we can use `config()`method to configure $routeProvider. $routeProvider provides method `.when()` and `.otherwise()` which we can use to define the routing for our app.

How to pass Parameters in Route Urls

How to Load local views (Views within <script> tag)

It is not always that you want to load view templates from different files. Sometimes the view templates are small enough that you might want them ship with main html instead of keeping them in separate html files.

ng-template directive

You can use ng-template to define small templates in your html file. For example:

```html
<script type="text/ng-template" id="add_order.html">

    <h2> Add Order </h2>

    {{message}}

</script>
```

Here we defined a template "add_order.html" inside <script> tag. Angular will automatically load this template in ng-view whenever add_order.html is referred in route.

Add Custom Data to RouteProvider

The $routeProvider provides methods when() and otherwise() which we used to define url routes. Sometime we might want to pass custom data based on certain route. For example you might use same Controller in different routes and use some custom data. For example:

```
when('/AddNewOrder', {

    templateUrl: 'templates/add_order.html',

    controller: 'CommonController',

    foodata: 'addorder'

}).

when('/ShowOrders', {

    templateUrl: 'templates/show_orders.html',

    controller: 'CommonController',

    foodata: 'showorders'

});


sampleApp.controller('CommonController', function($scope, $route) {

    //access the foodata property using $route.current

    var foo = $route.current.foodata;



    alert(foo);



});
```

In above code we defined a new property "foodata" while defining route. This property is then accessed in controller using `$route.current.foodata`.