**Controllers**

Controller is a JavaScript **constructor function** that is used to augment the [Angular Scope](https://docs.angularjs.org/guide/scope). Controllers are the place where we define our application behaviors by defining properties and functions.

$controller service is responsible for instantiating controllers.

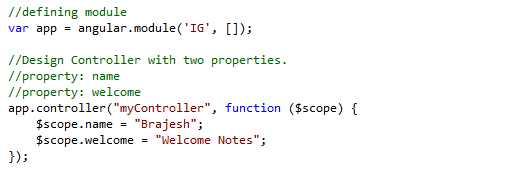
Use controllers to:

* Set up the initial state of the $scope object.
* Add behavior to the $scope object.

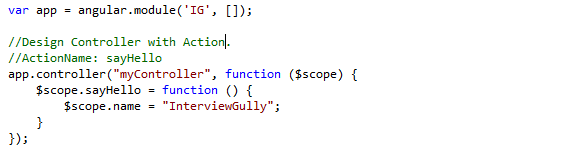
Do not use controllers to:

* Manipulate DOM — Controllers should contain only Application logic. Angular has [databinding](https://docs.angularjs.org/guide/databinding) for most cases and [directives](https://docs.angularjs.org/guide/directive) to encapsulate manual DOM manipulation.
* Format input — Use [angular form controls](https://docs.angularjs.org/guide/forms) instead.
* Filter output — Use [angular filters](https://docs.angularjs.org/guide/filter) instead.
* Share code or state across controllers — Use [angular services](https://docs.angularjs.org/guide/services) instead.
* Manage the life-cycle of other components (for example, to create service instances).

Property Initialization in Controller



Action declaration in Controller



**NOTE**: It is considered a best-practice to name our controllers as [Name]Controller, rather than [Name]Ctrl.

Example

external.js

//defining module

var app = angular.module('IG', []);

//Action Method: increase

//Action Method: decrease

app.controller('FirstController', function ($scope) {

$scope.counter = 0;

$scope.add = function (amount) { $scope.counter += amount; };

$scope.subtract = function (amount) { $scope.counter -= amount; };

});

index.html

<!DOCTYPE html>

<html ng-app="IG">

<head>

<title>AngularJS rootScope and scope :: InterviewGully.com</title>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.2.6/angular.js"></script>

<script src="Scripts/external.js"></script>

</head>

<body>

<div ng-controller="FirstController">

<h4>The simplest adding machine ever</h4>

<button ng-click="add(1)" class="button">Increase</button>

<button ng-click="subtract(1)" class="button alert">Decrease</button>

<h4>Current count: {{ counter }}</h4>

</div>

</body>

</html>

**Controller Hierarchy (Scopes within Scopes)**

By default, for any property that AngularJS cannot find on a local scope, AngularJS will crawl up to the containing (parent) scope and look for the property or method there. If AngularJS can’t find the property there, it will walk to that scope’s parent and so on and so forth until it reaches the Controllers $rootScope.

If it doesn’t find it on the $rootScope, then it moves on and is unable to update the view.

//defining module

var app = angular.module('IG', []);

//Property: person

app.controller('ParentController', function ($scope) {

$scope.person = { greeted: false };

});

//Action: sayHello

app.controller('ChildController', function ($scope) {

$scope.sayHello = function () {

$scope.person.name = "Ari Lerner";

$scope.person.greeted = true;

}

});

To see this behavior in action, let’s create a ParentController that contains the user object and a

ChildController that wants to reference that object.

If we bind the ChildController under the ParentController in our view, then the parent of the ChildController’s $scope object will be the ParentController’s $scope object. Due to the prototypal behavior, we can then reference data on the ParentController’s containing $scope on the child scope.

<!DOCTYPE html>

<html ng-app="IG">

<head>

<title>AngularJS rootScope and scope :: InterviewGully.com</title>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.2.6/angular.js"></script>

<script src="Scripts/external.js"></script>

</head>

<body>

<div ng-controller="ParentController">

<div ng-controller="ChildController">

<button ng-click="sayHello()">Say hello</button>

</div>

{{ person }}

</div>

</body>

</html>

**Sharing Data between Controller**

//defining module

var app = angular.module('IG', []);

//Property: person

app.controller('FirstController', function ($scope,data) {

$scope.person.name = data;

});

//Property: person

app.controller('SecondController', function ($scope,data) {

$scope.person.name = data;

});

//factory services

app.factory('data', function () {

return {

Message: 'hey I am ur service'

};

});

<!DOCTYPE html>

<html ng-app="IG">

<head>

<title>AngularJS rootScope and scope :: InterviewGully.com</title>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.2.6/angular.js"></script>

<script src="Scripts/external.js"></script>

</head>

<body>

<div ng-controller="FirstController">

{{ person }}

</div>

<div ng-controller="SecondController">

{{ person }}

</div>

</body>

</html>