**Understanding AngularJS $rootScope and $scope**

Scope is an object that refers to the application model. it acts as glue between controller and view. Scopes are hierarchical in nature and follow the DOM (Document Object Model) structure of your angular app.

AngularJS has two scope objects: $rootScope and $scope.

**$scope**

A $scope is a JavaScript object which is used for communication between controller and view. Basically, $scope binds a view (DOM element) to the view model and functions defined in a controller.

**$rootScope**

The $rootScope is the top-most scope. An app can have only one $rootScope which will be shared among all the components of an app. Hence it acts like a global variable. All other $scopes are children of the $rootScope.

**AngularJS: $rootScope and $scope with example**

external.js

//defining module

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

//the run function acts as a main method for the angular app.

app.run(function ($rootScope) {

$rootScope.site = "interviewgully.com";

$rootScope.name = "Brajesh Kumar"

});

app.controller("myController", function ($scope, $rootScope) {

$scope.name = "Sujeet Srivastava";

$scope.welcome = "Welcome to " + $rootScope.site;

});

app.controller("testController", function ($scope, $rootScope) {

$scope.welcome = "Welcome to " + $rootScope.site;

});

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="myController">

<h1>rootScopeController</h1>

<p>rootScope: {{site}}</p>

<!--changes in site will be done on controller level(myController level) rootScope not global rootScope object-->

<p>rootScope(ng-model): <input type="text" ng-model="site" /></p>

<p>welcome from scope: {{welcome}}</p>

<p>welcome from scope(ng-model): <input type="text" ng-model="welcome" /></p>

<p>scope: {{name}}</p>

</div>

<div ng-controller="testController">

<h1>ScopeController</h1>

<p>rootScope: {{site}}</p>

<!--changes in site will be done on controller level(testController level) rootScope not global rootScope object-->

<p>rootScope(ng-model): <input type="text" ng-model="site" /></p>

<p>welcome from scope: {{welcome}}</p>

<p>name from rootScope: {{name}}</p>

</div>

<div>

<h1>Outside from both Controller</h1>

<p>name from rootScope: {{name}}</p>

<p>site from rootScope: {{site}}</p>

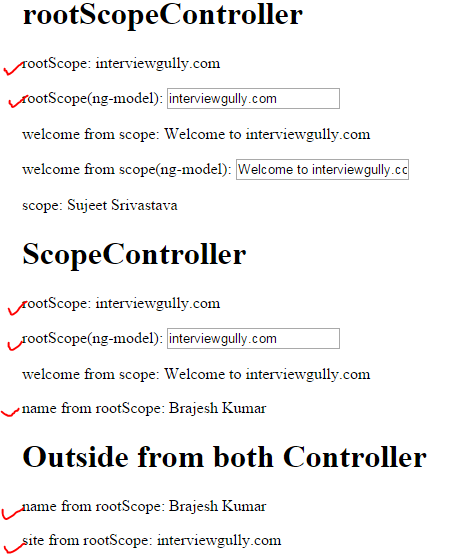
</div>

</body>

</html>

**How it works**

1. When you use ng-model with $rootScope objects then AngularJS updates those objects under a specific $scope of a controller but not at global level $rootScope.



2. Create a private $scope for each controller to bind it to the view.

