

```

app.controller('DivisionController', ['$scope', '$rootScope', '$location', '$http',
'$routeParams', '$window', function ($scope, $rootScope, $location, $http, $routeParams,
$window) {
    $scope.changeField = false;
    $scope.changeDuration = false;
    $scope.changeFavField = false;
    $scope.changeName = false;
    $scope.newPool = false;
    $scope.newPoolName = "";
    $scope.allLetters = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    $scope.division = "";
    $scope.division.letters = [];
    $scope.orderByField = 'Number';
    $scope.reverseSort = false;
    $scope.finalsTypes = [{ nr: 0, name: "Round Robin"}, { nr: 1, name: "Knockout"} ];
    $scope.finalsType = "";

    //Getting all the data for the divisions, by using a get request to the backend, which
    will send back the data from the database.
    $scope.getDivisionData = function() {
        $http.get($rootScope.apiUrl + "/Division/Details?id=" + $routeParams.divisionId)
        .success(function(data)
        {
            if(data.status === "success"){
                $scope.division = data;
                $scope.finalsType = $scope.finalsTypes[data.FinalsStage];
                for(var i = 0; i < $scope.division.Matches.length; i++)
                {
                    $scope.division.Matches[i].StartTime = new
                    Date(parseInt($scope.division.Matches[i].StartTime.substr(6)));
                }

                $scope.division.letters = [];
                for (var i = 0; i < $scope.division.FinalsLinks.length; i++)
                {
                    $scope.division.letters.push($scope.allLetters[i]);
                }
            }
            else {
                $scope.error = "Kunne ikke læse række";
            }
        }).error(function(err)
        {
            $scope.error = err;
        })
    }

    $scope.getDivisionData();

    $scope.newPoolFunc = function() {
        $scope.newPool = !$scope.newPool;
    }

    //Adding a new pool to the division. Making a post request to the backend to call the
    Create function in PoolController.
    $scope.addPool = function(name) {
        $scope.buttonDisabled = true;

```

```

    $http.post($rootScope.apiUrl + "/Pool/Create", { name: name, divisionId:
$routeParams.divisionId })
    .success(function(data) {
        if(data.status === "success"){
            $scope.newPoolFunc();
            $scope.getDivisionData();
        } else {
            $scope.error = "Kunne ikke tilføje pulje";
            $scope.buttonDisabled = false;
        }
        $scope.buttonDisabled = false;
    }).error(function(err){
        $scope.deleteErr = err;
    })
    $scope.getDivisionData();
}
$scope.buttonDisabled = false;

//Deleting the specific division. Making a post request to the backend to call the
Delete function DivisionController.
$scope.remove = function() {
    var deleteDivision = $window.confirm('Er du sikker på du vil slette rækken? \nHvis
kampprogrammet har været planlagt, vil det nu blive slettet og kan tage lidt tid');

    if(deleteDivision){
        $http.post($rootScope.apiUrl + "/Division/Delete", { id: $routeParams.divisionId })
        .success(function(data) {
            if(data.status === "success"){
                $location.path("/tournament/" + $routeParams.tournamentId);
            }
            else {
                $scope.error = "Kunne ikke slætte række";
            }
        }).error(function(data) {
            $scope.deleteErr = data;
        })
    }
}

$scope.changeFieldFunc = function() {
    $scope.changeField = !$scope.changeField;
}

//This function is used the change the fieldsize of the division. Done by making a post
request ot the backend which calls the Edit function
//in DivisionController, and sends the required parameters with it.
$scope.changeNewFieldFunc = function(editField) {
    $http.post($rootScope.apiUrl + "/Division/Edit", { name: $scope.division.Name,
fieldSizeInt: editField, tournamentId: $routeParams.tournamentId, id:
$routeParams.divisionId, matchDuration: $scope.division.MatchDuration })
    .success(function(data){
        if(data.status === "success"){
            $scope.division.FieldSize = data.editField;
            $scope.changeFieldFunc();
            $scope.getDivisionData();
        } else {
            $scope.error = "Kunne ikke ændre banestørrelse";
        }
    })
}

```

```

    }).error(function(err){
        $scope.editErr = err;
    })
}

$scope.changeDurationFunc = function() {
    $scope.changeDuration = !$scope.changeDuration;
}

//This function is used the change the matchduration of the division. Done by making a
post request ot the backend which calls the Edit function
//in DivisionController, and sends the required parameters with it.
$scope.changeNewDurationFunc = function(editMatchDuration) {
    $http.post($rootScope.apiUrl + "/Division/Edit", { name: $scope.division.Name,
    fieldSizeInt: $scope.division.FieldSize, tournamentId: $routeParams.tournamentId, id:
$routeParams.divisionId, matchDuration: editMatchDuration })
    .success(function(data){
        if(data.status === "success"){
            $scope.division.MatchDuration = data.editMatchDuration;
            $scope.changeDurationFunc();
            $scope.getDivisionData();
        } else {
            $scope.error = "Kunne ikke ændre kamplængde";
        }
    }).error(function(err){
        $scope.editErr = err;
    })
}

//Function used to to go a specific pool
$scope.gotoPool = function(currPool) {
    $location.url($location.url() + "/pool/" + currPool.Id);
}

//Function used to to go a specific team
$scope.gotoTeam = function(currTeam, currPool) {
    $location.url($location.url() + "/pool/" + currPool.Id + "/team/" + currTeam.Id);
}

//Function used to to go the tournament overview page
$scope.gotoTournament = function() {
    $location.url("/tournament/" + $routeParams.tournamentId);
}

$scope.changeDivNameFunc = function() {
    $scope.changeName = !$scope.changeName;
}

//This function is used the change the name of the division. Done by making a post
request ot the backend which calls the Edit function
//in DivisionController, and sends the required parameters with it.
$scope.changeNewDivNameFunc = function(newName) {
    $http.post($rootScope.apiUrl + "/Division/Edit", { name: newName, fieldSizeInt:
$scope.division.FieldSize, tournamentId: $routeParams.tournamentId, id:
$routeParams.divisionId, matchDuration: $scope.division.MatchDuration })
    .success(function(data){
        if(data.status === "success"){
            $scope.division.Name = data.newName;

```

```

        $scope.changeDivNameFunc();
        $scope.getDivisionData();
    } else {
        $scope.error = "Kunne ikke ændre række-navn";
    }
}).error(function(err){
    $scope.editErr = err;
});
}

$scope.finalLinkChanged = function(finalLink, letter)
{
    $http.post($rootScope.apiUrl + "/FinalsLink/Edit", { id: finalLink.Id, finalStage:
($scope.allLetters.indexOf(letter) + 1), poolPlacement: finalLink.PoolPlacement })
        .success(function(data) {
        })
        .error(function(err) {
            $scope.uflErr = err;
        })
    }

    $scope.finalTypeChanged = function(finaltype)
    {
        $http.post($rootScope.apiUrl + "/Division/ChangeStructure", { divisionId:
$scope.division.Id, typeId: finaltype.nr })
            .success(function(data) {
            })
            .error(function(err) {
                $scope.uflErr = err;
            })
        }

        $scope.isScheduled = false;

        $scope.schedule = function () {
            $scope.isScheduled = !$scope.isScheduled;
        }
    }]);

```