

AngularJS

Cours et notions

J'aurais commencé ce TP avec les étudiants (normalement). Il faut donc le terminer.

Data binding

Exemple 1 : Cet exemple très basique remplit une table avec des données issues d'un tableau

```
<!DOCTYPE html>
<html>
<head>
  <title>Remplit un tableau avec des données au format JSON</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.9/angular.min.js">
  </script>
  <style type='text/css'>
    table {
      width: 100%;
    }
    table, tr, td {
      border: 1px solid #DDDDDD;
      border-collapse: collapse;
    }
  </style>
</head>
<body>
  <h1>Recette des crêpes</h1>
  <div ng-app>
    <div ng-init="articles = {nom:'Farine', prix:0.58},
    {nom:'Sucre', prix:1.20}, {nom:'Oeufs', prix:0.15},
    {nom:'Lait', prix:0.79}, {nom:'Beurre', prix:7.9}]">
      <table>
        <caption>J'ai {{articles.length}} articles à acheter</caption>
        <tr>
          <th>Article</th>
          <th>Prix au kg, au litre, ou à l'unité</th>
        </tr>
        <tr ng-repeat="article in articles">
          <td>{{article.nom}}</td>
          <td>{{article.prix | currency:"€ "}}</td>
        </tr>
      </table>
    </div>
  </div>
</body>
</html>
```

Exercice 1 : Ajouter une colonne avec les quantités suivante : Farine 375g, Sucre : 50g, Œufs : 4, Lait : 50cl, Beurre : 20g. Ajouter une colonne qui calcule le prix à payer par article (pour gérer les différentes unités utilisez la directive `ng-if`, ex : ` `).

Contrôler

Exemple 2 : Autre façon (beaucoup plus standard) de faire en utilisant un module et un contrôleur :

```
<script>
angular.module('initExample', [])
.controller('contrôleur', ['$scope', function($scope) {
  $scope.list = [
    { nom: 'Farine', prix: 2.3 },
    { nom: 'Sucre', prix: 1.35 },
    { nom: 'Gâteaux', prix: 3.9 },
    { nom: 'Huile', prix: 5.25 }
  ];
}]);
```

```

</script>
<div ng-app="initExample" ng-controller="controleur" >
  <table>
    <caption>J'ai {{list.length}} articles à acheter</caption>
    <tr>
      <th>Article</th>
      <th>Prix</th>
    </tr>
    <tr ng-repeat="article in list">
      <td>{{article.nom}}</td>
      <td>{{article.prix | currency:"€ "}}</td>
    </tr>
  </table>
</div>

```

Exercice 2 : Adaptez votre page de l'exercice 1 en utilisant un contrôleur AngularJS. Permettre de modifier les prix en utilisant des champs `<input>` et la directive `ng-model`. Ajoutez un champs d'entrée de type number permettant de paramétrer le nombre de personnes. Pensez à filtrer les résultats avec `| number : fractionSize` afin de tronquer les nombres décimaux.

Exemple 3: Compter le nombre de mots saisis

```

<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8">
  <title>Contrôleurs 2 - jsFiddle demo by bestmomo</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.9/angular.min.js"></scri
pt>
</head>
<body>
  <div ng-app="Teste" ng-controller="controleur">
    <textarea ng-model="texte" ng-change="changement()"></textarea>
    Nombre de mots : {{nombre}}
  </div>
  <script>
    angular.module('Teste', [])
      .controller('controleur', [
        '$scope', function($scope) {
          $scope.nombre = 0;
          $scope.changement = function() {
            this.nombre = this.texte.split(/\b\w+\b/).length - 1;
          };
        }
      ]);
  </script>
</body></html>

```

Exercice 3 : Compléter l'exemple en comptant le nombre de phrases.

Exercice 4 : Complétez votre recette en ajoutant le prix total qui se mettra à jour automatiquement (voir du côté de `ng-bind`).

Exemple 4 : Construire un tableau en saisissant des données

```

<!DOCTYPE html>
<html>
<head>
  <title>Saisie dans un tableau</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.9/angular.min.js"></scri
pt>
  <style type='text/css'>
    table {
      width: 100%;
    }
  </style>

```

```

        table, tr, td {
            border: 1px solid #DDDDDD;
            border-collapse: collapse;
        }
    </style>
</head>
<body>
    <div ng-app="Saisie" ng-controller="controleur">
        <form ng-submit="ajouter(nom, age)">
            Nom : <input type="text" name="nom" ng-model="nom">
            Age : <input type="number" name="age" ng-model="age">
            <button type="submit">Ajouter</button>
        </form>
        <table>
            <caption>Personnes</caption>
            <tr>
                <th>Nom</th>
                <th>Age</th>
            </tr>
            <tr ng-repeat="personne in personnes">
                <td>{{personne.nom}}</td>
                <td>{{personne.age}}</td>
            </tr>
        </table>
    </div>
    <script>
    angular.module('Saisie', [])
        .controller('controleur', ['$scope', function ($scope) {
            $scope.personnes = new Array();
            $scope.ajouter = function (nom, age) {
                $scope.personnes.push({ 'nom': nom, 'age': age });
            };
        }]);
    </script>
</body>
</html>

```

Exercice 5 : En vous inspirant de l'exemple, ajoutez des champs `<input>` permettant d'ajouter de nouveaux ingrédients à votre recette de pâte à crêpes (mmm les crêpes au rhum, *avec modération*).

Appel Ajax

Exemple 5 : Appel AJAX à un fichier de données au format JSON. Module, contrôleur, factory, injection de données.

```

<!DOCTYPE html>
<html><head>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.9/angular.min.js">
</script>
    <title>Appel Ajax</title>
    <style type='text/css'>
        table {
            width: 100%;
        }
        table, tr, td {
            border: 1px solid #DDDDDD; border-collapse: collapse;
        }
    </style>
</head>
<body >
    <script>
    angular.module('initExample', [])
        .controller('controleur', function ($scope, Donnees) {
            $scope.list = [
                { Nom_Musicien: 'Bach', Prenom_Musicien: 'Jean_Sébastien' }
            ];
        });
    </script>

```

```

    ];
    $scope.fetchData = function () {
        $scope.list = [];
        var data = new Donnees().getProfile().then(function(data) {
            data.data.forEach(function (elem) {
                $scope.list.push(elem);
            });
        });
    }
})

.factory('Donnees', function ($http) {
    var apiUrl = 'donnees.json';
    var Donnees = function () {
        this.profile = null;
    };
    Donnees.prototype.getProfile = function () {
        var self = this;
        return $http.get(apiUrl).then(function (response) {
            self.profile = response.data;
            return response;
        });
    };
    return Donnees;
});
</script>
<div ng-app="initExample" ng-controller="controleur">
    <button ng-click="fetchData()">Données</button>
    <table>
        <caption>{{list.length}} Musicien</caption>
        <tr>
            <th>Nom</th>
            <th>Prénom</th>
        </tr>
        <tr ng-repeat="article in list">
            <td>{{article.Nom_Musicien}}</td>
            <td>{{article.Prenom_Musicien}}</td>
        </tr>
    </table>
</div>
</body>
</html>

```

Exercice 6 : Enregistrez les ingrédients de votre recette dans un fichier json et chargez dynamiquement votre fichier via un appel Ajax AngularJS (certaines parties de votre page ne s'afficheront qu'après le chargement du fichier, pensez à utiliser **ng-hide** ou **ng-show** pour cela). Tester avec une autre recette de votre choix (n'oubliez pas d'inclure un titre et le nombre de personnes pour chaque recette). Pour une api plus haut niveau, on pourra utiliser **\$resource**.