

ng-bind-template DIRECTIVE IN ANGULARJS

- The ngBindTemplate directive specifies that the element text content should be replaced with the interpolation of the template in the ngBindTemplate attribute.
- Instead of ng-bind, using the ng-bind-template bind more than one expression (`{{...}}`).
- This directive is essential hence HTML elements cannot contain span element like TITLE and OPTION.

Syntax for ng-bind-template directive in AngularJS:

```
<element ng-bind-template= "{{...}} {{...}}" > </element>
```

Parameter Values:

Value	Description
expression	Denotes one or more expressions to evaluate, each surrounded by <code>{{}}</code> .

Sample coding for ng-bind-template directive in AngularJS:

```
<!DOCTYPE html>
<html>
<head>
  <title>Wikitechy AngularJS Tutorial</title>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/angular.min.js"> </sc
ript>
  <body>
    <div ng-app="myApp" ng-controller="templateCtrl">
      <h3>ng-bind-template Directive example in AngularJS Tutorial</h3>
      FirstName:<input type="text" ng-model="firstName"> <br/> <br/>
      LastName:<input type="text" ng-model="lastName">
      <p ng-bind-template="{{ firstName }}{{lastName }}!"> </p>
    </div>
    <script>
      var app = angular.module("myApp", []);
      app.controller("templateCtrl", function($scope) {
        $scope.firstName = "wiki";
        $scope.lastName = "techy";
      });
    </script>
  </body>
</html>
```



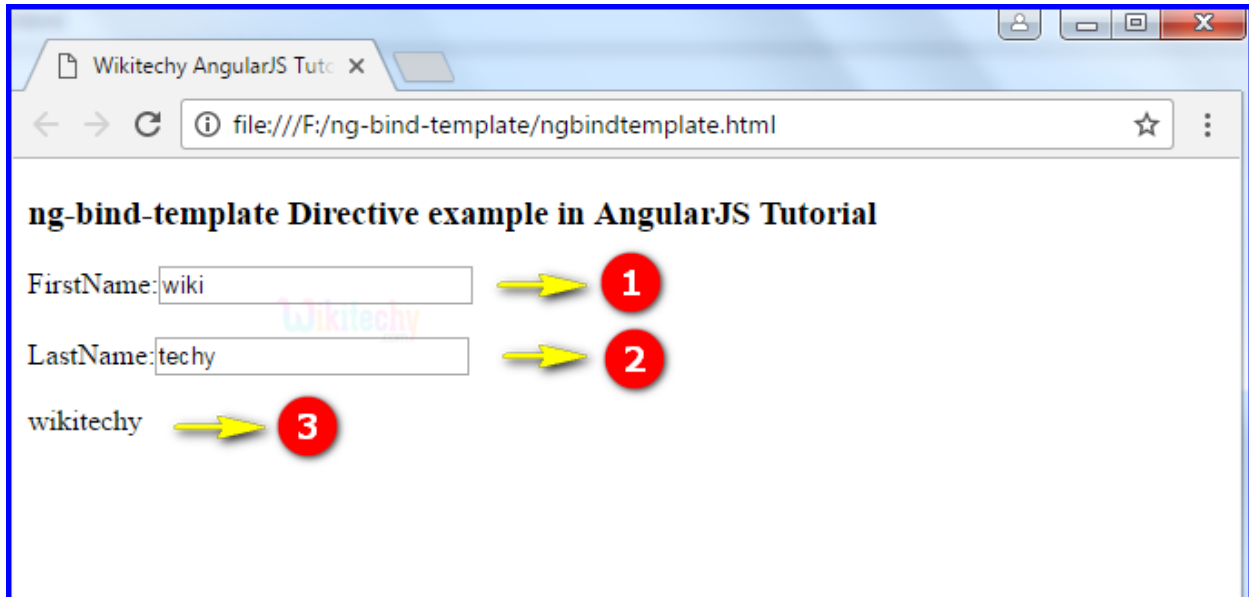
Code Explanation for ng-bind-template directive in AngularJS:

```
<!DOCTYPE html>
<html>
<head>
  <title>Wikitechy AngularJS Tutorial</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.5.6/
angular.min.js"></script>
<body>
  <div ng-app="myApp" ng-controller="templateCtrl">
    <h3>ng-bind-template Directive example in AngularJS Tutorial</h3>
    FirstName:<input type="text" ng-model="firstName"><br/><br/>
    LastName:<input type="text" ng-model="lastName">
    <p ng-bind-template="{{firstName}} {{lastName}}"></p>
  </div>
  <script>
    var app = angular.module("myApp", []);
    app.controller("templateCtrl", function($scope) {
      $scope.firstName = "wiki";
      $scope.lastName = "techy";
    });
  </script>
</body>
</html>
```

1. The **ng-app** specifies the root element (**myApp**) to define AngularJS application.
2. The **ng-controller** specifies the application controller in AngularJS the controller value is given as "**templateCtrl**".
3. The **ng-bind-template** is used to bind the AngularJS values in the format of multiple expression (**{{ firstName}} {{ lastName}}**) to the HTML Document and the output displayed in **<p>** tag. but the **ng-bind** directive is not support the multiple expression instead of it can be use the concatenate operator (+).
4. **\$scope.firstName** is used to declare the scope object as "**wiki**".
5. **\$scope.lastName** is used to declare the scope object as "**techy**".



Sample Output for ng-bind-template directive in AngularJS:



1. The output displays the firstName "wiki".
2. The output displays the lastName "techy".
3. The output displays the concatenation of firstname and lastname ("wikitechy!") using ng-bind-template.