# RESTRICTIONS OF DIRECTIVES IN ANGULARJS

- We can restrict our own directives for specific HTML element. These are the methods restrict the directives.
    - Element Name– restrict :”E”.
    - Attribute Name– restrict :”A”.
    - Class Name – restrict :”C”.
    - Comment – restrict :”M”.
- We can combine the restrict method like ‘AEC’ this will be allow us to invoke directives by Attribute(A) name or element(E) name or class name(C) method.
- We should define the directiveName in **camelCase** and case sensitive, we should invoke the directive in **dash-delimited**(directive-name) format.

## Syntax for Restrict Directives Using ‘E’ (Element):

```
<directive-name></directive-name>
.
.
app.directive("directiveName", function() {
    return {
        restrict : "E",
    };
};
```

- The Above Syntax only allows to invoke directive-name by using Element only.

## Syntax for Restrict Directives using ‘A’ (Attribute):

```
<element directive-name ></element>
.
.
app.directive("directiveName", function() {
    return {
        restrict : "A",
    };
};
```

- The Above Syntax only allows to invoke directive-name by using Attribute only.

## Syntax for Restrict Directives using ‘C’ (Class):

```
<element class="directive-name"></element>
.
.
app.directive("directiveName", function() {
    return {
        restrict : "C",
    };
};
```

- The Above Syntax only allows to invoke directive-name by using Class only.

## Syntax for Restrict Directives using ‘M’ (Comment):

```
<!-- directive: directive-name -->
.
.
app.directive("directiveName", function() {
    return {
        restrict : "M",
      replace : true,
    };
};
```

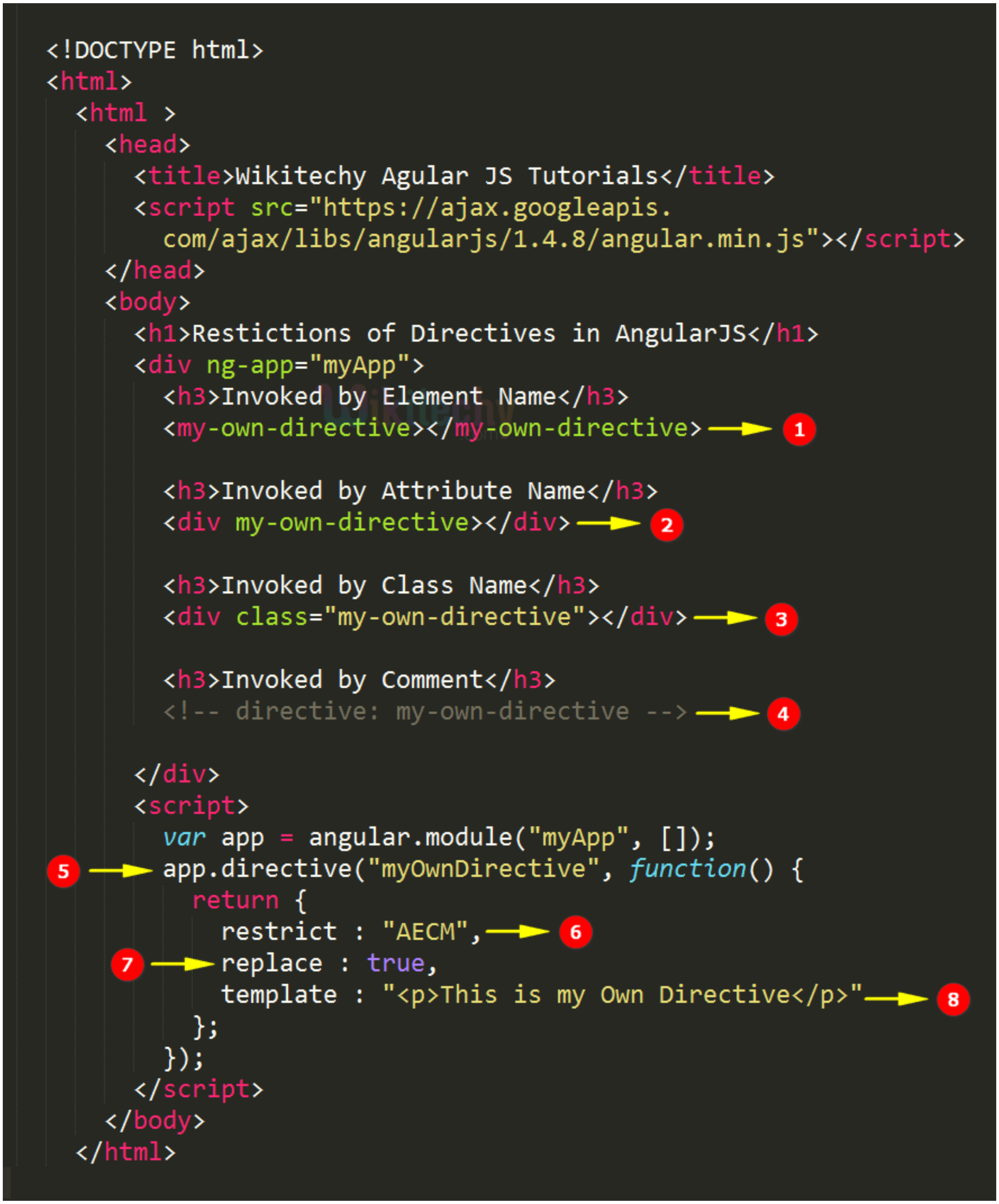
- The Above Syntax only allows to invoke directive-name by using Comment only.
- We have to include the **replace:true** property while using restrict for comments or else the comment would be invisible.

## Sample code for MySQL Insert with PHP in AngularJS:

- We are going to see a Sample code for restrict with **“AECM”** we can change the Values depends upon our requirements like **“AE”** or **“AEC”** or **“C”.**

```
<!DOCTYPE html>
<html>
    <head>
        <title>Wikitechy AngularJS Tutorials</title>
        <script src="https://ajax.googleapis.com/ajax/libs/angularjs/
1.4.8/angular.min.js"> </script>
    </head>
    <body>
        <h1> Restictions of Directives in AngularJS </h1>
        <div ng-app="myApp">
            <h3>Invoked by Element Name</h3>
            <my-own-directive></my-own-directive>
            <h3>Invoked by Attribute Name</h3>
            <div my-own-directive></div>
            <h3>Invoked by Class Name</h3>
            <div class="my-own-directive"></div>
            <h3>Invoked by Comment</h3>
            <!-- directive: my-own-directive -->
        </div>
        <script>
            var app = angular.module("myApp", []);
            app.directive("myOwnDirective", function() {
                return {
                    restrict : "AECM",
                    replace : true,
                    template : "<p>This is my Own Directive</p>"
                };
            });
        </script>
    </body>
</html>
```

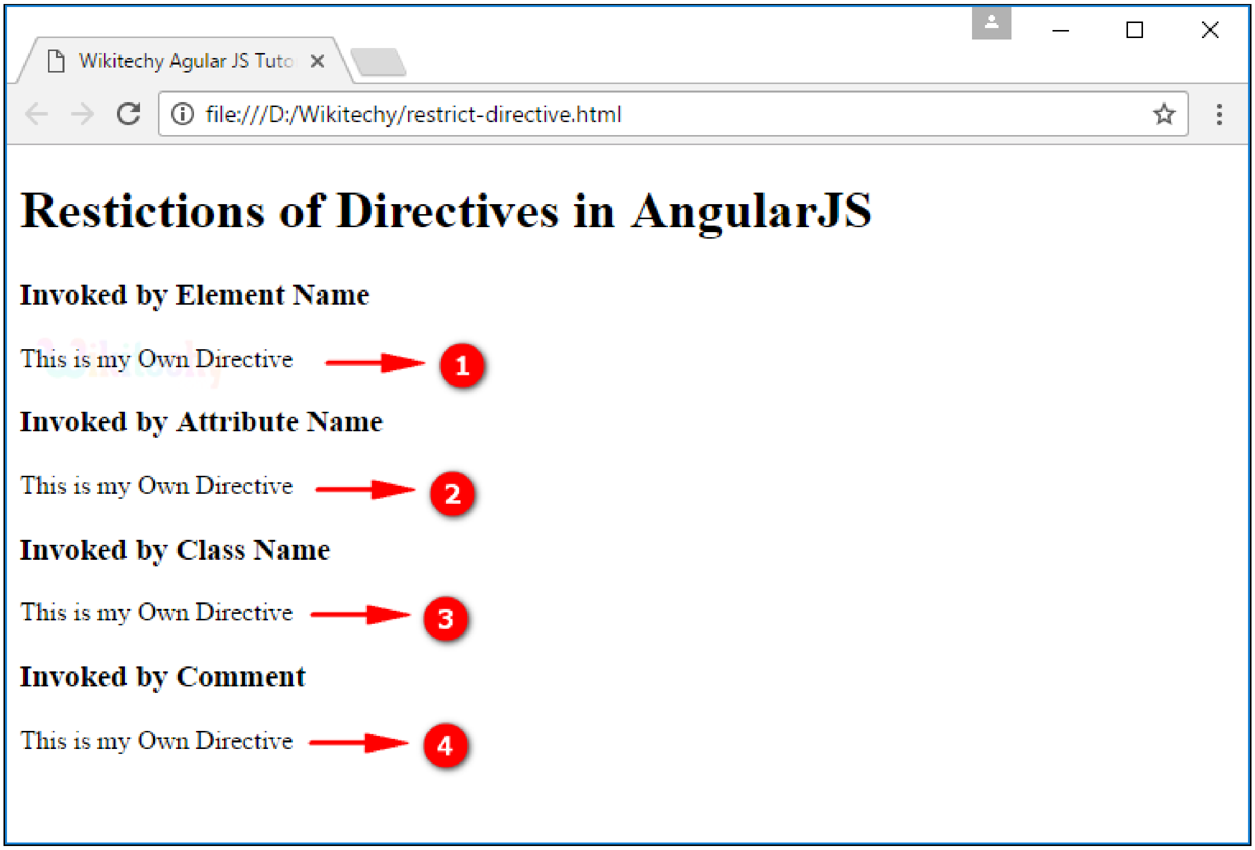
## Code Explanation for restrict Directives in AngularJS:



```
<!DOCTYPE html>
<html>
  <html >
    <head>
      <title>Wikitechy Agular JS Tutorials</title>
      <script src="https://ajax.googleapis.
        com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>
    </head>
    <body>
      <h1>Restictions of Directives in AngularJS</h1>
      <div ng-app="myApp">
        <h3>Invoked by Element Name</h3>
        <my-own-directive></my-own-directive>

        <h3>Invoked by Attribute Name</h3>
        <div my-own-directive></div>

        <h3>Invoked by Class Name</h3>
        <div class="my-own-directive"></div>

        <h3>Invoked by Comment</h3>
        <!-- directive: my-own-directive -->

      </div>
      <script>
        var app = angular.module("myApp", []);
        app.directive("myOwnDirective", function() {
          return {
            restrict : "AECM",
            replace : true,
            template : "<p>This is my Own Directive</p>"
          };
        });
      </script>
    </body>
  </html>
```

1. Invoke directive as Element Name.
2. Invoke directive as Attribute Name.
3. Invoke directive as Class Name.
4. Invoke directive as Comment.
5. To define own custom Directive named myOwnDirective.
6. The **restrict : "AECM"** will be allow us to invoke directives by **A**ttribute name or **E**lement name or **C**lass name or Co**M**ment method.
7. The **replace : true** is used to display the directive content in comments. (Example: this will replace the **<!-- directive: my-own-directive -->** with **"<p>This is my Own Directive</p>"** ).
8. The **template** to bind the content in the directive.

## Sample Output for restrict Directives in AngularJS:


Wikitechy Agular JS Tuto ×
file:///D:/Wikitechy/restrict-directive.html
1
2
3
4

# Restictions of Directives in AngularJS

### Invoked by Element Name

This is my Own Directive

### Invoked by Attribute Name

This is my Own Directive

### Invoked by Class Name

This is my Own Directive

### Invoked by Comment

This is my Own Directive

1. The output shows custom directive by invoke **Element Name** method.
2. The output shows custom directive by invoke **Attribute Name** method.
3. The output shows custom directive by invoke **Class Name** method.
4. The output shows custom directive by invoke **Comment** method.