

[← Back to Week 3](#)[X Lessons](#)[Prev](#)[Next](#)

Lesson 1 - Asynchronous Behavior, Promises and Ajax in Angular

Lesson 2 - Introduction to Custom Directives

Lesson 3 - Directive APIs, Manipulating the DOM, and transclude



Lecture 30, Part 1: Directive APIs and "&" 7 min



Lecture 30, Part 2: Directive APIs and "&" 12 min



Practice Quiz: Quiz 29 2 questions



Lecture 31, Part 1: Manipulating the DOM with link 4 min



Lecture 31, Part 2: Manipulating the DOM with link 14 min



Practice Quiz: Quiz 30 3 questions



Lecture 32, Part 1: Using Directive's transclude to 2 min

PRACTICE QUIZ

Quiz 29

2 questions

To Pass
Deadline

100% or higher
June 11, 11:59 PM PDT

[Start](#)



Quiz 29

Practice Quiz, 2 questions

2/2 points (100%)

**Congratulations! You passed!**[Next Item](#)1 / 1
points

1. Given the following snippet of directive's template code and assuming that myMethod is the property name on the isolate scope that's declared with '&' as the value:

```
1 <button ng-click="dirController.myMethod(dirController.arg);"
```

What must the 'dirController.arg' value be?

- ☐ Anything that makes sense for the business logic of the parent controller: string with a name, object with name property, etc.
- ☒ A map of key/value pairs, where the key is the argument name with which the parent controller declared this method call in the parent's template.

Correct1 / 1
points

2. In Javascript, functions are just objects. So, why can't you just pass in a function to execute into the directive using '=' just like any other object?

- ☐ Passing function values using '=' is not possible in Angular.
- ☒ You could do that, but it's generally not a good idea to do that because that function can refer to the 'this' variable (or might at some future version of it)



Quiz 29

Practice Quiz, 2 questions

2/2 points (100%)



A map of key/value pairs, where the key is the argument name with which the parent controller declared this method call in the parent's template.

Correct



2. In Javascript, functions are just objects. So, why can't you just pass in a function to execute into the directive using '=' just like any other object?

1 / 1
points

Passing function values using '=' is not possible in Angular.



You could do that, but it's generally not a good idea to do that because that function can refer to the 'this' variable (or might at some future version of it), and the 'this' reference will therefore not refer to what it was originally intended to. It will refer to the directive's controller instance instead of the parent controller's instance.

Correct

Correct!



Passing it as a '&' or reference is just new syntax. It's the same thing as passing it through '=' syntax.