










[← Back to Week 2](#)[X Lessons](#)[Prev](#)[Next](#)

## Filters, Digest Cycle and Data Binding

## Looping, Controller As Syntax

## Creating and Configuring Custom Services

 Lecture 20, Part 1: Custom Services 4 min Lecture 20, Part 2: Custom Services 11 min **Practice Quiz:** Quiz 18 2 questions Lecture 21, Part 1: Custom Services with .factory() 7 min Lecture 21, Part 2: Custom Services with .factory() 9 min **Practice Quiz:** Quiz 19 2 questions Lecture 22, Part 1: Custom Services with .provider() 5 min Lecture 22: Part 2: Custom Services with .provider() 8 min **Practice Quiz:** 2 questions

### PRACTICE QUIZ

## Quiz 19

2 questions

**To Pass  
Deadline**100% or higher  
June 4, 11:59 PM PDT[Start](#)



## Quiz 19

Practice Quiz, 2 questions

2/2 points (100%)

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

1. When creating a service with the `.factory()` method, it is ultimately the AngularJS framework that creates the function/object that is the service.

1 / 1  
points

True



False

**Correct**

Correct. It is YOU, the developer, who must instantiate either an object or create a function as part of the implementation of the factory.



2. The `.factory()` method is limited to producing only Singleton Services. It can't simply produce some object we want to use in our application.

1 / 1  
points

True



False

**Correct**

Correct. The whole point of a `.factory()` method is that it can produce *whatever* you need. It's a factory! Just order a "widget" and it'll be made for you. 😊