Multiple tackles!

Two battles down and one to go! This time your monster will be facing a monster that has an elemental advantage over it. You will not be able to rely on your elemental move to win.

We need to teach your monster how to use its Tackle move multiple times in a row. We are going to change its Tackle move to make it more powerful.

We will teach your monster to randomly attack between three and five times. It will be extremely powerful and will help you win this battle. Let's add in the code and then talk about what it is doing. Edit the `tackleButtonPressed` function in the following way:

```
func tackleButtonPressed(numberOfTackles: Int)
{
    for counter in 1...numberOfSwipes
    {
        performTackle()
    }
}
```

Functions with parameters

We are changing our `tackleButtonPressed` function to take in a parameter! This means that when the game calls `tackleButtonPressed`, it will also have to include the number of times your monster should perform swipe. The syntax for this is:

```
func functionName(parameterName: parameterType)
{
    // function code goes here
}
```

In our case, the parameter name is `numberOfTackles`. We can access `numberOfTackles` and do anything to it that we could do to a normal variable.

Remember how we talked about `Strings` and `Ints`? They are different **types**. In this case, our parameter type is `Int` -- again, it's just a fancy name for a whole number.

Remember, the call to this function is automatically generated for us -- it is a **Tackle Button Handler**. Just for a moment though, let's peer under the hood and see what that function call looks like. If we wanted to pretend that the Tackle button had been pressed and we randomly

were going to attack 4 times:

```
tackleButtonPressed(4)
```

We pass in the `4` parameter value inside the parentheses. We don't have to do this since the function will automatically get called for us, with a parameter between 3-5.

For-loops

The second thing we see in the changes is a for-loop. For-loops allow us to run code multiple times. This is good since our monster will be performing swipe a random number of times! The syntax for a for-loop is:

```
for counter in startingNumber...endingNumber
{
    // for-loop code goes here
}
```

The `counter` will count how many times we have run the code. A for-loop will execute the code inside the loop once for each number from `startingNumber` until `endingNumber` inclusive. Each execution, it will update `counter` to the current number. In our case, if `numberOfSwipes` is 3, `counter` will equal 1, then 2, then 3. Each time execution it will run `performTackle()` since it is inside the for-loop's curly braces.

Now that your monster knows how to tackle multiple times, run the code and see if you can win the third battle. Good luck!