

Solution Review: Variable Number of Arguments

This lesson discusses the solution to the challenge given in the previous lesson.

```
package main
import "fmt"

func main() {
    // function calls for multiple arg.
    fmt.Println(sumInts())
    fmt.Println(sumInts(2, 3))
    fmt.Println(sumInts(0, 1, 2, 3, 4, 5, 6, 7, 8, 9))
}

func sumInts(list ...int) (sum int){    // function to calculate sum of variable arg.
    for _, v := range list {
        sum = sum+v
    }
    return
}
```



Variable Arguments

In the code above, at **line 11** there is a function header: `sumInts(list ...int) (sum int)` which means that `sumInts` takes a *variable* number of parameters and returns an integer `sum`. Then, we have a *for* loop at **line 12**, where we iterate over the list of these parameters and start adding them together as: `sum=sum+v` and returning `sum` at **line 15**. Now, look at the `main`. At **line 6**, **line 7** and **line 8** we are calling `sumInts` for different values and printing the output to verify the results.

That's it about the solution. In the next lesson, you'll study the concept of deferring and tracing functions.