

Challenge: Filtering with Higher-Order Functions

This lesson brings you a challenge to solve.

WE'LL COVER THE FOLLOWING ^

- Problem statement
- Input
- Sample input
- Output
- Sample output

Problem statement

Make a function `Filter` which accepts `s` as the first parameter and a function `fn func(int) bool` as the second parameter. The `Filter` should return a *slice* of the elements of `s` which fulfill the function `fn` (make it true). Test this out with `fn` testing if the integer is **even**.

Input

A slice and the function name(use function that checks if the number is even or not)

Sample input

```
[0 1 2 3 4 5 6 7 8 9] // slice
even                  //function that checks if the number is even or not
```

Output

Slice containing even numbers only

Sample output

```
[0 2 4 6 8]
```

Try to implement the function below. Feel free to view the solution, after giving some shots. Good Luck!

```
package main
import "fmt"
import "strconv"
import "encoding/json"

func Filter(s []int, fn func(int) bool) []int {

    return nil
}

func even(n int) bool {
    if n%2 == 0 {
        return true
    }
    return false
}
```



Filtering with Higher Order Functions

We hope that you were able to solve the challenge. The next lesson brings you the solution to this challenge.