

```
1 #lang racket
2
3 ;; Defining filter
4
5 (define (my-filter f l)
6   (if (empty? l)
7       l
8       (if (f (first l))
9           (cons (first l) (my-filter f (rest l)))
10              (my-filter f (rest l)))))
11
12 (my-filter string? (list 1 2 #t "cat"))
13
14 (define numbers (list 5 6 7 8))
15
16 (define (greater-than-5 lst)
17   (filter (lambda (x) (> x 5)) numbers))
18
19 (greater-than-5 numbers)
20
21 (filter (lambda (x) null) numbers)
22
23 ;;Exercise: titlecase
24
25 (define (all-titlecase lst)
26   (filter (lambda (x) (equal? (string-titlecase x) x))
26   lst))
27
28 (all-titlecase (list "cat" "Cat" "CAT" "ALL Cats"))
29
```