

Generators and iterators

- Example: `range(a, b)`
- Generators and iterators are similar - basically, an interface to a function that generates terms of a sequence one after the other
- Can be accessed using for loops or `next(...)`
- Can be defined using `yield` instead of `return` - "returns" multiple times:

```
def myRange(a, b, step=1):  
    x = a  
    while x < b:  
        x += step  
        yield x
```

7. LeetCode problems