Solution Review: Return Even Numbers From 1 to n

Make your own iterator class to return a list of even numbers.

WE'LL COVER THE FOLLOWING

Solution: Use Iterator

Notice (in the code below) that the `next` method in lines 8-16 makes a list, then appends the even numbers in that range to the list using a `for` loop.

```
class MyRange:
  def __init__(self, n):
    self.n = n

  def __iter__(self):
    return self

  def next(self):
    evenArray = [] # next method returns this list
    for i in range(1, self.n+1):
      if i % 2 is 0: # checks if number is even
        value = i
        evenArray.append(i) # adds the even number to the list
      else: # number was odd
        i+=1
    return evenArray

myrange = MyRange(8)
print (myrange.next())
```

The figure below illustrates how this is done.

1 of 29

```
i = 1 % 2
```

2 of 29

```
i = 1 % 2 â□0
```

3 of 29

```
i = 2
```

4 of 29

```
i = 2 % 2
```

5 of 29

```
i = 2 % 2 = 0
```

6 of 29

```
i = 2 % 2 = 0
```

2

7 of 29

```
i = 3
```

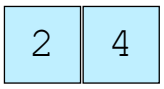

```
i = 3 % 2
```


```
i = 3 % 2 â□0
```

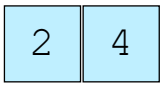

```
i = 4
```


```
i = 4 % 2
```

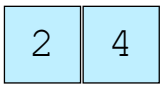

```
i = 4 % 2 = 0
```



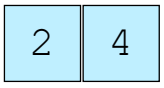
```
i = 5
```



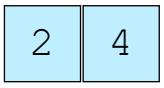

```
i = 5 % 2
```



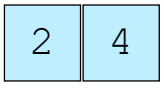

```
i = 5 % 2 â□0
```



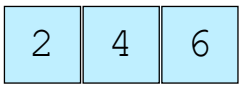

```
i = 6
```



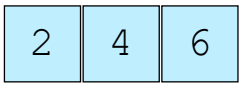

```
i = 6 % 2
```



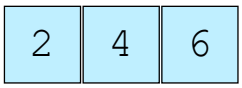

```
i = 6 % 2 = 0
```



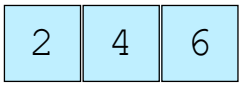
```
i = 7
```



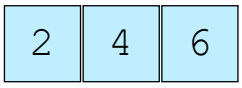

```
i = 7 % 2
```




```
i = 7 % 2 â□0
```

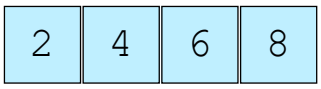



```
i = 8
```

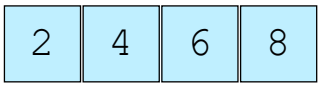



```
i = 8 % 2
```

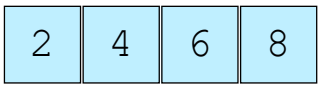

```
i = 8 % 2 = 0
```



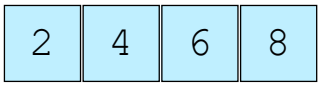
```
i = 9
```

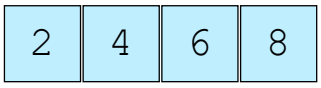



```
i = 9 = self.n + 1
```




```
i = 9 = self.n + 1
  Loop ends
```





Now, let's try to output numbers from n down to 0 in the next lesson.