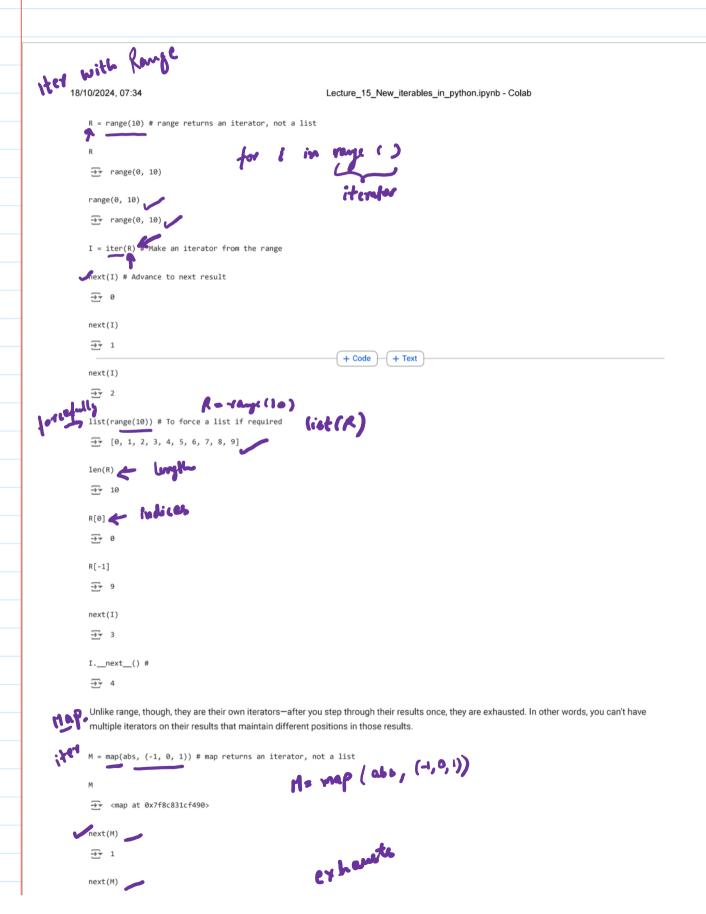
Lecture 15_New_iterables_in_python

10 September 2024 13:13



Lecture_15_New_iterables_in_python.ipynb - Colab

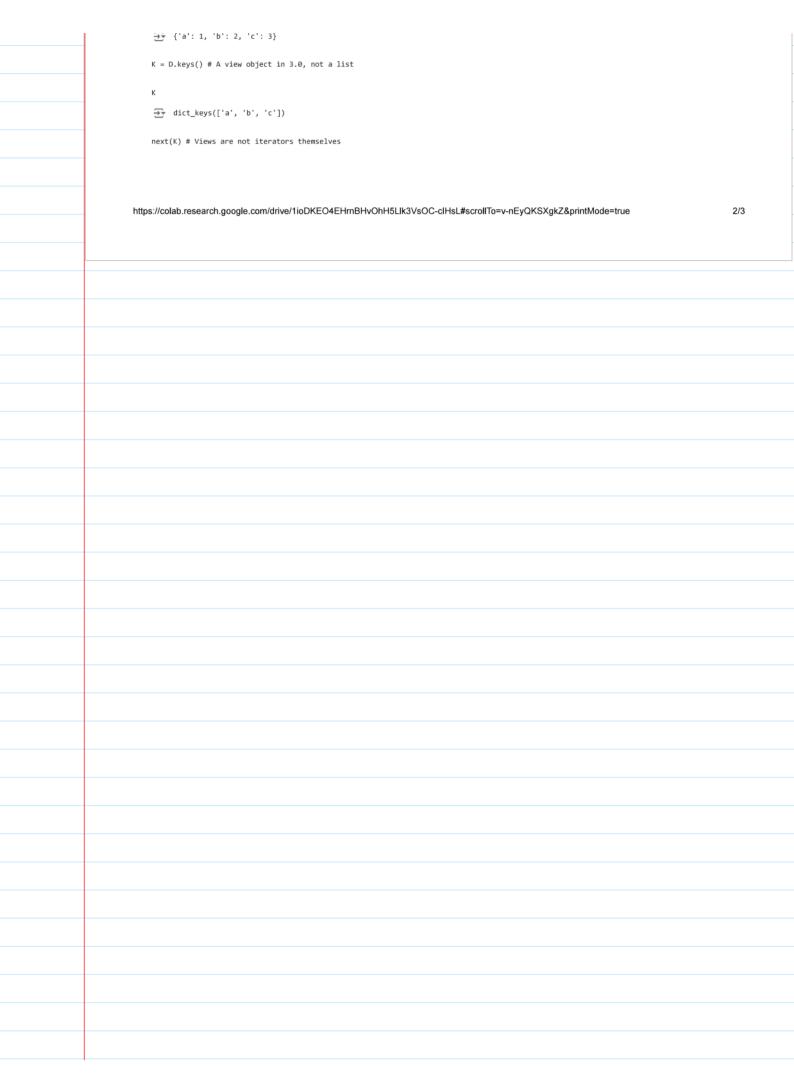


```
exhause
   next(M)
    <del>_</del> → 0
    next(M) # Use iterator manually: exhausts results
            # These do not support len() or indexing
    <u>→</u> 1
                                                                                                                                   1/3
https://colab.research.google.com/drive/1ioDKEO4EHrnBHvOhH5Llk3VsOC-clHsL#scrollTo=v-nEyQKSXgkZ&printMode=true
                  M= (---)
18/10/2024, 07:34
                                                        Lecture_15_New_iterables_in_python.ipynb - Colab
   for x in M: print(x) # map iterator is now empty: one pass only
 M = map(abs, (-1, 0, 1)) # Make a new iterator to scan again
   for x in M: print(x) # Iteration contexts auto call next()
                                                   Im (M)
   list(map(abs, (-1, 0, 1))) # Can force a real list if needed
    → [1, 0, 1]
    Multiple Versus Single Iterators
     = range(3) # range allows multiple iterators
                                             Traceback (most recent call last)
        <ipython-input-28-lef46f494a83> in <cell line: 1>()
----> 1 next(R)
        TypeError: 'range' object is not an iterator
     Next steps: Explain error
                         R ->(0,3)
    I1 = iter(R)
   next(I1)
    next(I1)
    <u>→</u> 1
   I2 = iter(R) # Two iterators on one range
    next(I1) # I1 is at a different spot than I2
    Dictionary View Iterators
   D = dict(a=1, b=2, c=3)
```

_____ 1

→ {'a': 1, 'b': 2, 'c': 3}

K = D.keys() # A view object in 3.0, not a list



Start coding or generate with AI.