BIS 420 PROGRAMMING FOR DATA SCIENCE

PRAJAKTA POHARE
CHAPTER 10 EXERCISE 11.9
ILLINOIS STATE UNIVERSITY

If you did Exercise 10.8, you already have a function named has_duplicates that takes a list as a parameter and returns True if there is any object that appears more than once in the list. Use a dictionary to write a faster, simpler version of has_duplicates. Solution: http: //thinkpython. com/ code/ has_ duplicates. py .

```
from __future__ import print_function, division


def has_duplicates(t):
    d = {}
    for x in t:
        if x in d:
            return True
        d[x] = True
    return False


def has_duplicates2(t):
    return len(set(t)) < len(t)


if __name__ == '__main__':
    t = [1, 2, 3]
    print(has_duplicates(t))
```

t.append(1)

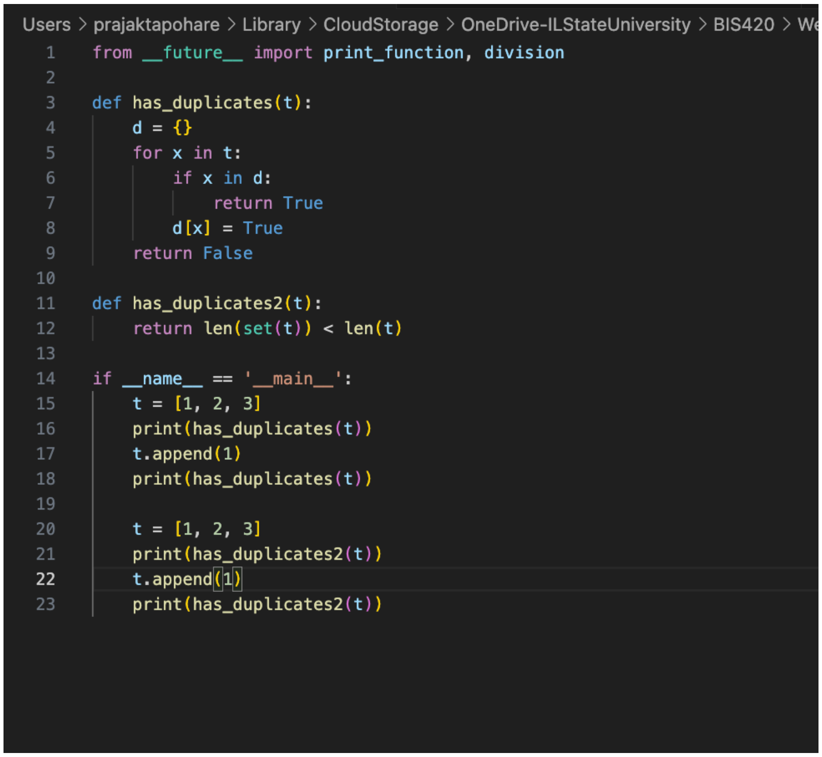
print(has_duplicates(t))

t = [1, 2, 3]

print(has_duplicates2(t))

t.append(1)

print(has_duplicates2(t))



Users > prajaktapohare > Library > CloudStorage > OneDrive-ILStateUniversity > BIS420 > We

```python
from __future__ import print_function, division

def has_duplicates(t):
    d = {}
    for x in t:
        if x in d:
            return True
        d[x] = True
    return False

def has_duplicates2(t):
    return len(set(t)) < len(t)

if __name__ == '__main__':
    t = [1, 2, 3]
    print(has_duplicates(t))
    t.append(1)
    print(has_duplicates(t))

    t = [1, 2, 3]
    print(has_duplicates2(t))
    t.append(1)
    print(has_duplicates2(t))
```