BIS 420 PROGRAMMING FOR DATA SCIENCE

PRAJAKTA POHARE

CHAPTER 6 EXERCISE 6.6

ILLINOIS STATE UNIVERSITY

A palindrome is a word that is spelled the same backward and forward, like “noon” and “redivider”. Recursively, a word is a palindrome if the first and last letters are the same and the middle is a palindrome.

The following are functions that take a string argument and return the first, last, and middle letters:

```
def first(word):

return word[0]

def last(word):

return word[-1]

def middle(word):

return word[1:-1]
```

We’ll see how they work in Chapter 8.

1. Type these functions into a file named palindrome.py and test them out. What happens if you call middle with a string with two letters? One letter? What about the empty string, which is written'' and contains no letters?

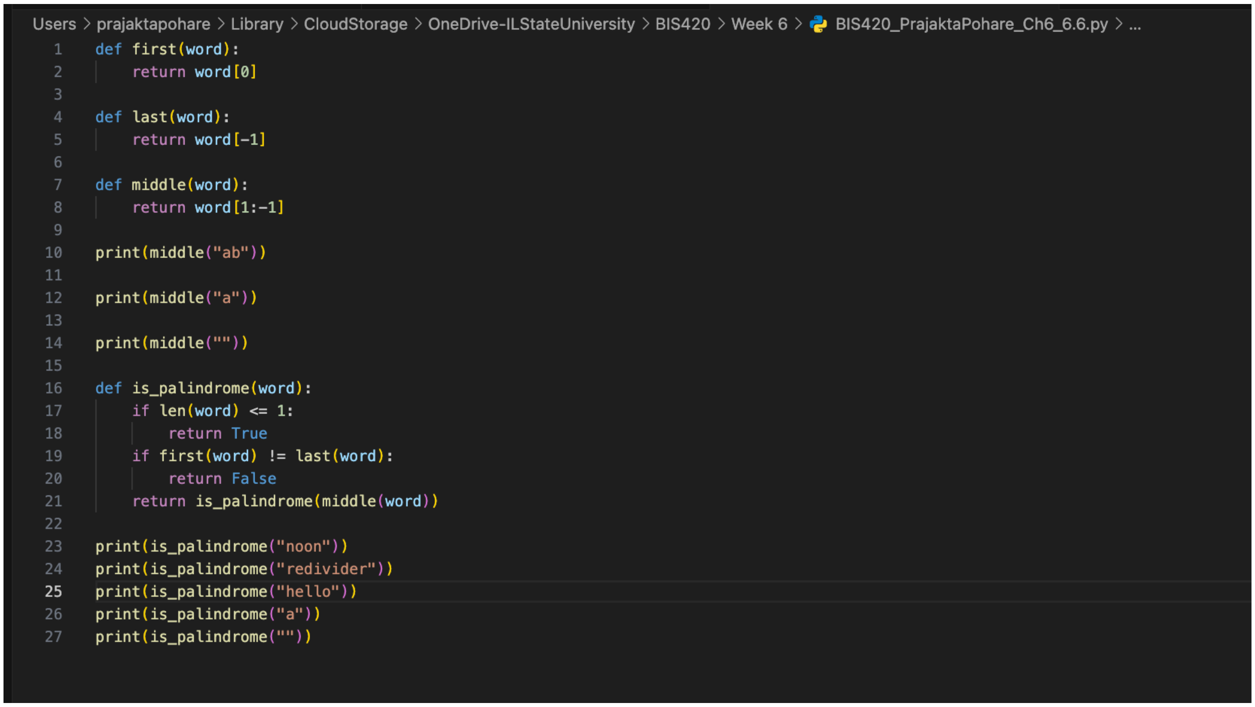
2. Write a function called is_palindrome that takes a string argument and returns True if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.

Code:

```
def first(word):
    return word[0]

def last(word):
    return word[-1]

def middle(word):
    return word[1:-1]

print(middle("ab"))

print(middle("a"))

print(middle(""))

def is_palindrome(word):
    if len(word) <= 1:
        return True
    if first(word) != last(word):
        return False
    return is_palindrome(middle(word))

print(is_palindrome("noon"))
print(is_palindrome("redivider"))
print(is_palindrome("hello"))
```

print(is_palindrome("a"))

print(is_palindrome(""))


Users > prajaktapohare > Library > CloudStorage > OneDrive-ILStateUniversity > BIS420 > Week 6 > BIS420_PrajaktaPohare_Ch6_6.6.py > ...

```python
def first(word):
    return word[0]

def last(word):
    return word[-1]

def middle(word):
    return word[1:-1]

print(middle("ab"))

print(middle("a"))

print(middle(""))

def is_palindrome(word):
    if len(word) <= 1:
        return True
    if first(word) != last(word):
        return False
    return is_palindrome(middle(word))

print(is_palindrome("noon"))
print(is_palindrome("redivider"))
print(is_palindrome("hello"))
print(is_palindrome("a"))
print(is_palindrome(""))
```