Regular Computer Science 1	Lab 14A 1-Day Minor Python Assignment
The "Simple Palindromes" Program	**90 & 100 Point Versions**

Assignment Purpose:

To gain a deeper understanding of string processing

Write a program that determines if an entered string is a *Palindrome*. True Palindromes are strings of characters that read the same backward as forward. Examples of Palindromes are:

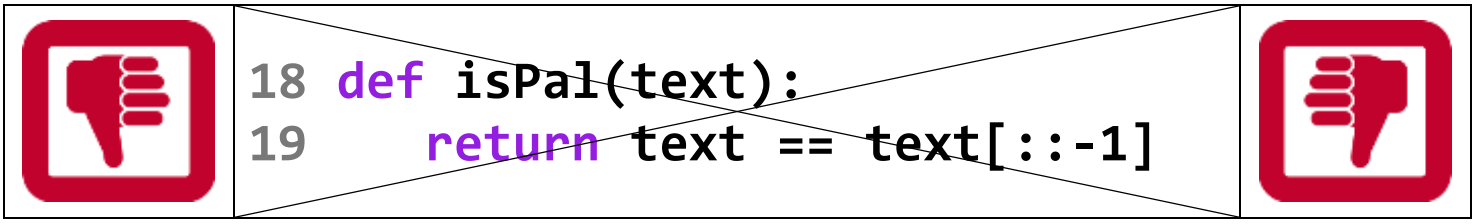
MADAM, RACECAR, BOB, HANNAH, CIVIC, KAYAK, LEVEL, REVIVER

Lab 14A Student Version	Do not copy this file, which is provided.

```
 1 # Lab14Ast.py
 2 # The "Simple Palindromes" Program
 3 # This is the student, starting version of Lab 14A.
 4 # Students need to complete the <isPal> function.
 5 # NOTE: This lab is meant for students in REGULAR CS1.
 6 #       Students in CS1-HONORS will do Lab 14B.
 7
 8
 9 def heading():
10     print()
11     print("*********************************************")
12     print("Lab 14A, Simple Palindromes")
13     print("90 Point Version")
14     print("By: JOHN SMITH") # Substitute your own name here.
15     print("*********************************************")
16
17
18 def isPal(text):
19     return False
20
21
22
23
24
25
```

```
26 ##########
27 #  MAIN  #
28 ##########
29
30 heading()
31 finished = False
32 while not finished:
33     print("\n")
34     text = input("Enter a string  -->  ")
35     print("\nPalindrome:         ",isPal(text))
36     choice = input("\nDo you wish to repeat this program? {Y/N} --> ")
37     if choice.upper()[0] != 'Y':
38         finished = True
39
```

NOTE: There is an advanced form of *String Slicing* that lets you write the **isPal** function with a single command (shown below). This is NOT ALLOWED for this assignment. You will not receive ANY credit for Lab 14A if you code the **isPal** function in this manner.



90 Point Version Specifics

The main thing this program needs to do is determine if an entered string is a *Palindrome*. To do this, you must complete the **isPal** function. Right now, this function has a single line of code, which just returns **False**. This simply allows the program to execute. To complete the **isPal** function, you need to write the necessary code so that it returns **True** if the entered string is a *Palindrome* and returns **False** if it is not.

For this version, the **isPal** function is *case-sensitive* meaning that *madam* and *MADAM* are Palindromes, but *Madam* and *madaM* are not.

90 Point Version Output

```
----jGRASP exec: python Lab14Av90.py

*****************************************
Lab 14A, Simple Palindromes
90 Point Version
By: JOHN SMITH
*****************************************


Enter a string  -->  MADAM

Palindrome:  True

Do you wish to repeat this program? {Y/N} --> Y


Enter a string  -->  qwerty

Palindrome:  False

Do you wish to repeat this program? {Y/N} --> Y


Enter a string  -->  RaceCar

Palindrome:  False

Do you wish to repeat this program? {Y/N} --> N

----jGRASP: operation complete.
```

100 Point Version Specifics and Output

The 100 point version is very similar to the 90 point version except now the **isPal** function is no longer *case sensitive*. So while *madam* and *MADAM* were already Palindromes, now *Madam, mADAM*, and *mADam* are Palindromes as well.

```
----jGRASP exec: python Lab14Av100.py

*****************************************
Lab 14A, Simple Palindromes
100 Point Version
By: JOHN SMITH
*****************************************


Enter a string  -->  MADAM

Palindrome:  True

Do you wish to repeat this program? {Y/N} --> Y


Enter a string  -->  qwerty

Palindrome:  False

Do you wish to repeat this program? {Y/N} --> Y


Enter a string  -->  RaceCar

Palindrome:  True

Do you wish to repeat this program? {Y/N} --> N

----jGRASP: operation complete.
```