class Solution(object):

def removePalindromeSub(self, s):

"""

:type s: str

:rtype: int

"""

def isPalindromeSub(s):

lo = 0;

high = len(s) -1;

while lo < high:

if s[lo] != s[high]: return False

lo += 1

high -= 1

return True

if not s:

return 0

if isPalindromeSub(s): return 1

return 2

####### someone else’s sample code using lo and high index counters, alternative to my approach of i and len(s) – i

class Solution(object):

def removePalindromeSub(self, s):

if s == s[ :: -1]:

return 1

return 2

"""

:type s: str

:rtype: int

"""

Simple approach of comparing if reverse is identical in value to forward

class Solution(object):

def removePalindromeSub(self, s):

"""

:type s: str

:rtype: int

"""

return 2 - (s == s[::-1]) - (s == "")

####### crazy concise logic, true or false binary 1 0 implementation