



Python练习题

入门级别

1. 【Q01】求用十进制，二进制，八进制表示都是回文数的所有数字中，大于十进制20的最小值。

例：9（十进制）= 1001（二进制）= 11（八进制）

回文数：如果把某个数的各个数字按相反的顺序排列，得到的数和原来的数相同，则这个数就是“回文数”。譬如：123454321就是一个回文数。

◦ 解题思路：

■ 涉及到的知识点

二进制转换：bin()

八进制转换：oct()

数值转换成字符串：str()

字符串的切片操作：[:], [:::-1]

■ 切入点

要保证一个二进制转换前和转换后一样，那么末位肯定是1，因为0101010是不对的，以0开始了。而末尾为1的二进制数，肯定是奇数。

解题代码：

```

def hw():
    '''
    取得比20大的最小回文数
    '''
    num = 21 # 因为要取得比20大的数
    while True: # 因为不知道是多少, 所以用了循环
        num_str_10 = str(num) # 将十进制转换成字符串
        num_str_2 = str(bin(num)) # 先将十进制转换成二进制, 再转换成字符串
        num_str_8 = str(oct(num)) # 将八进制的数转换成字符串
        num_str_2 = num_str_2.replace('0b', '') # 因为Python中二进制前面会用0b表示, 我们需要将他去掉
        num_str_8 = num_str_8.replace('0o', '') # 八进制则是以0o开头, 同样要去掉
        if (num_str_10==num_str_10[::-1]) and (num_str_2==num_str_2[::-1]) and (num_str_8==num_str_8[::-1]):
            '''
            将每个进制的数和其反转后的结果进行比较
            '''
            print("大于20的最小的回文数: ", num)
            print('二进制为:', num_str_2)
            print('八进制为:', num_str_8)
            break # 结束循环
        num += 2 # 因为是奇数, 所以每次加2

if __name__ == "__main__":hw()

```

- 答案为: 585, 二进制: 1001001001, 八进制: 1111