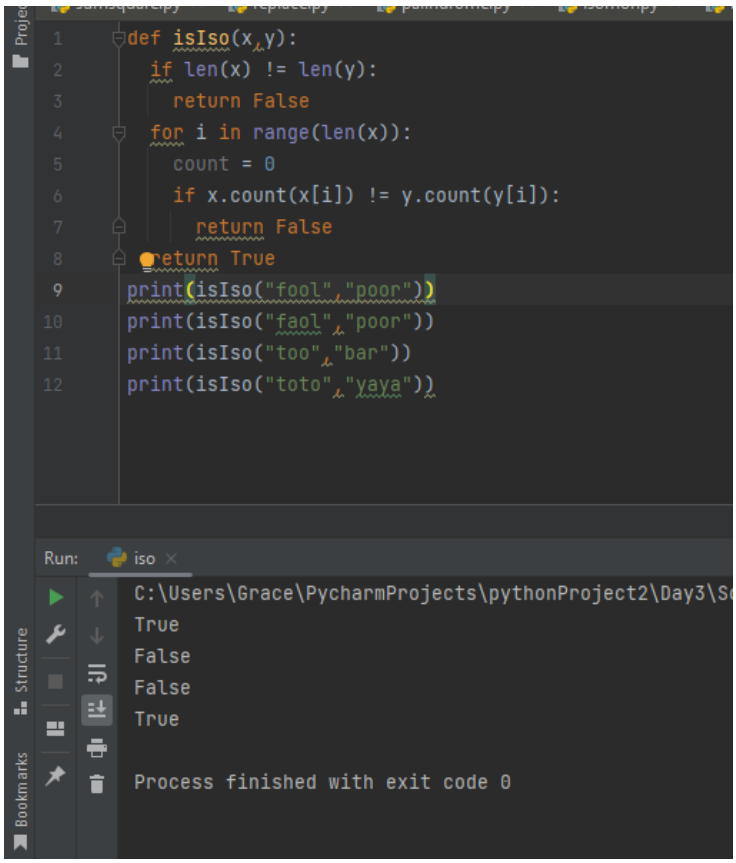
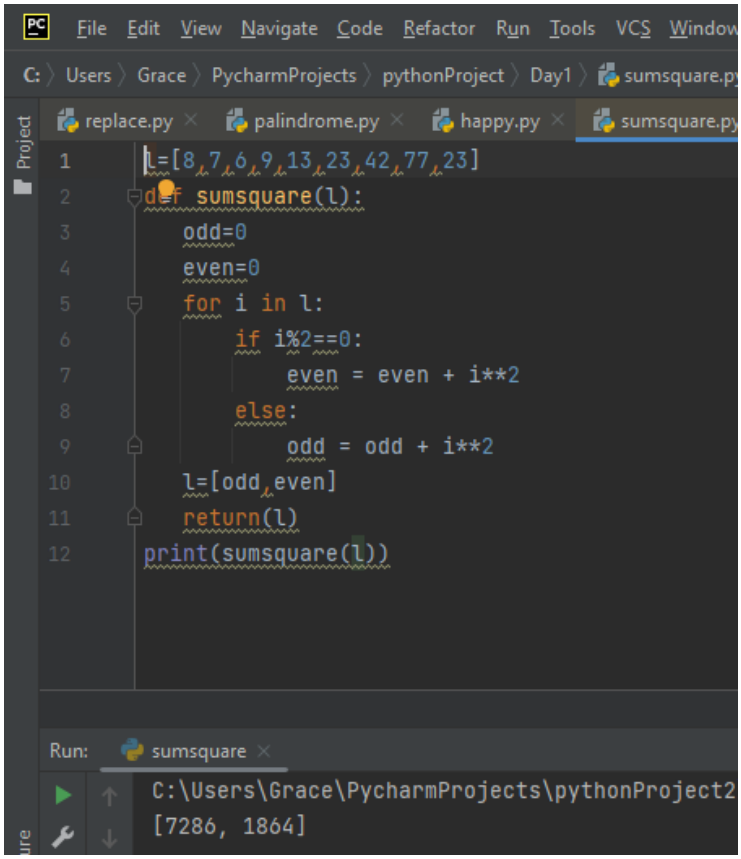
DAY 1 PROGRAMS

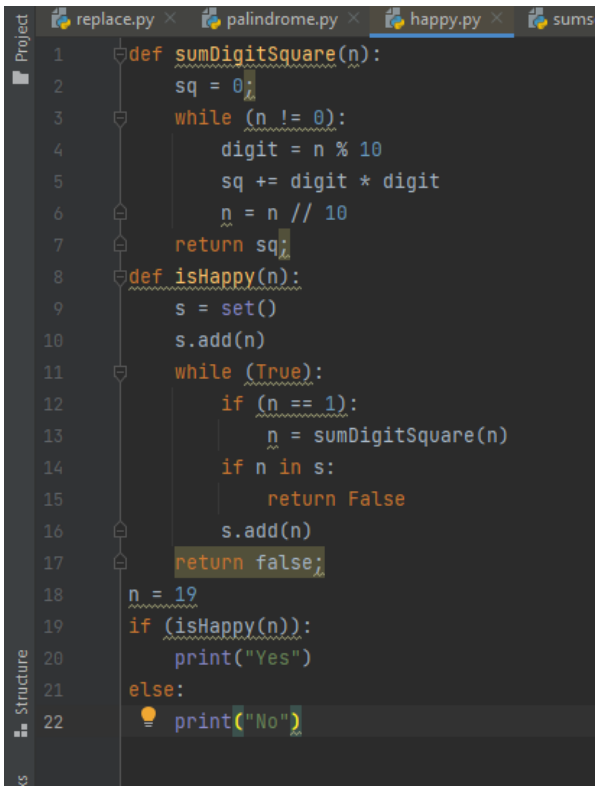
1.



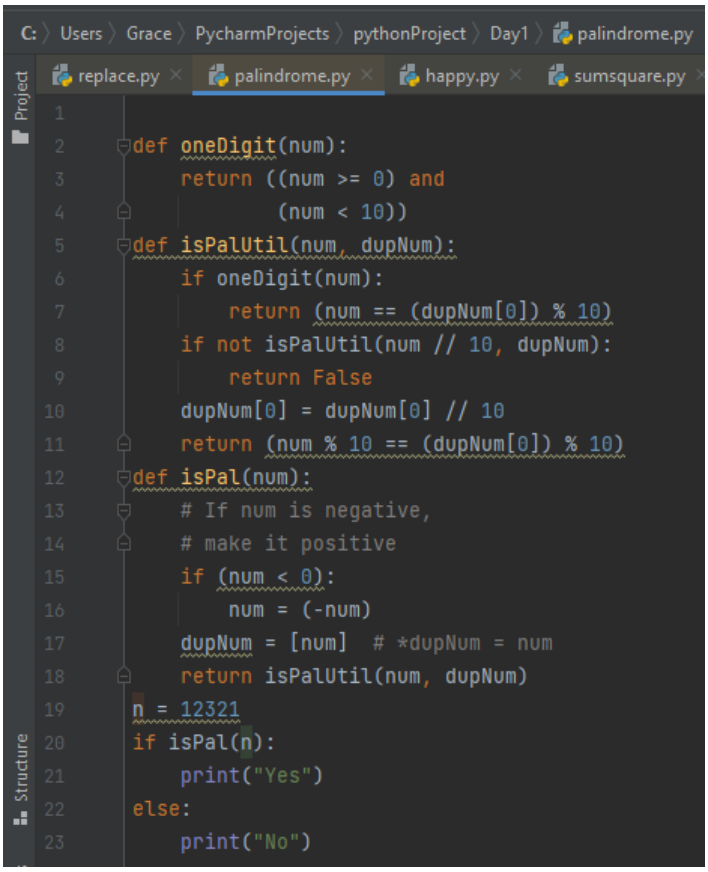
2.



3.


replace.py
palindrome.py
happy.py
sums
Project
Structure
def sumDigitSquare(n):
sq = 0;
while (n != 0):
digit = n % 10
sq += digit * digit
n = n // 10
return sq;
def isHappy(n):
s = set()
s.add(n)
while (True):
if (n == 1):
n = sumDigitSquare(n)
if n in s:
return False
s.add(n)
return false;
n = 19
if (isHappy(n)):
print("Yes")
else:
print("No")

4.


C: Users Grace PycharmProjects pythonProject Day1 palindrome.py
replace.py
palindrome.py
happy.py
sumsquare.py
Project
Structure
def oneDigit(num):
return ((num >= 0) and
(num < 10))
def isPalUtil(num, dupNum):
if oneDigit(num):
return (num == (dupNum[0]) % 10)
if not isPalUtil(num // 10, dupNum):
return False
dupNum[0] = dupNum[0] // 10
return (num % 10 == (dupNum[0]) % 10)
def isPal(num):
# If num is negative,
# make it positive
if (num < 0):
num = (-num)
dupNum = [num] # *dupNum = num
return isPalUtil(num, dupNum)
n = 12321
if isPal(n):
print("Yes")
else:
print("No")

5.

```
fresh=int(input("Enter the number of fresh loves purchased"))
old=int(input("Enter the number of day old loaves purchased"))
print("Regular price: Rs.185.00")
fresh_amount = 185.00*float(fresh)
old_amount = (185*0.4)*float(old)
Total=fresh_amount+old_amount
print("Amount of new loaves:Rs "_,fresh_amount)
print("Amount of day old loaves: Rs"_,old_amount)
print("Total amount: Rs"_,Total)
```

```
Run:  bread
Enter the number of fresh loves purchased5
Enter the number of day old loaves purchased6
Regular price: Rs.185.00
Amount of new loaves:Rs  925.0
Amount of day old loaves: Rs 444.0
Total amount: Rs 1369.0

Process finished with exit code 0
```

6.

```
class Solution:
    def maxArea(self, A):
        maxarea = 0
        l = 0
        r = len(A) - 1
        while l < r:
            base = r - l
            height = min(A[l], A[r])
            area = base * height
            maxarea = max(area, maxarea)
            if A[l] < A[r]:
                l += 1
            else:
                r -= 1
        return maxarea
ob=Solution()
print(ob.maxArea([1,5,4,3]))
print(ob.maxArea([3,1,2,4,5]))
```

7.

```python
def countstrings(n, start):
    if n == 0:
        return 1
    count = 0
    for i in range(start, 5):
        count += countstrings(n - 1, i)
    return count
def countVowelStrings(n):
    #  char arr[5]={'a','e','i','o','u'};
    # starting from index 0 add the vowels to strings
    return countstrings(n, 0)
n = 1
print(countVowelStrings(n))
```

```
vowels
C:\Users\Grace\PycharmProjects\pythonProject2\Day3\Scripts
5
```

8.

```python
def isNumeric(s):
    s = s.strip()
    try:
        s = float(s)
        return True
    except:
        return False
print(isNumeric("0.2"))
print(isNumeric("xyz"))
print(isNumeric("Hello"))
print(isNumeric("-2.5"))
print(isNumeric("10"))
```

9.

```
time=int(input())
entry=list(map(int, input().split()))
exit=list(map(int, input().split()))
Present=0
Total_guests=0
for i in range(time):
    Present += entry[i]-exit[i]
    if Total_guests < Present:
        Total_guests = Present
print(Total_guests, end = " ")
```

10.

```
def addFrequencyToCharacter(s):
    frequency = [0] * 26
    n = len(s)
    for i in range(n):
        frequency[ord(s[i]) - ord('a')] += 1
    for i in range(n):
        add = frequency[ord(s[i]) - ord('a')] % 26
        if (ord(s[i]) + add <= ord('z')):
            s[i] = chr(ord(s[i]) + add)
        else:
            add = (ord(s[i]) + add) - (ord('z'))
            s[i] = chr(ord('a') + add - 1)
    print("".join(s))
if __name__ == '__main__':
    str = "ghee"

    addFrequencyToCharacter([i for i in str])
```