

# Exercises

By Nachai Paramesthanakorn

# Create dict

กำหนดให้ `x = ['apple', 'mango', 'apple', 'banana', 'kiwi', 'peach', 'banana', 'banana']`

สร้าง dict ใหม่โดยนับจำนวน item ของ `x` โดยให้ชื่อตัวแปรคือ `dict_A`

1. ถ้ามี item เป็น key อยู่แล้วให้เพิ่มค่าขึ้น 1 หน่วย
2. ถ้าไม่มี item เป็น key ให้สร้าง key, value ใหม่ โดยให้ item เป็น key และ value เท่ากับ 1

## Output:

แสดงผล `dict_A`

## Hint:

- ใช้ loop ในการแจกแจง item จากลิสต์
- ใช้ if-else เพิ่มแบ่งข้อมูลว่ามี item เป็น key หรือไม่

## ตัวอย่าง

Input	Output
ไม่มี input	{'apple': 2, 'mango': 1, 'banana': 3, 'kiwi': 1, 'peach': 1}

# Get dict value

กำหนดให้ `x = { 'a': 30, 'b': 40, 'c': 50 }`

เข้าถึง value ใน key 'a', 'b', 'c' เพื่อนำ value มาคำนวณ

แล้วสร้างเป็น dict ตัวใหม่ชื่อ `dict_S` มีเงื่อนไขดังนี้

1. นำ value ของ key 'a' มา ยกกำลัง 2 แล้วบวกด้วย 99 แล้วเก็บใน `dict_S` ตัวใหม่โดยมี key = 'a'
2. นำ value ของ key 'b' มา หาร 5 แล้ว ยกกำลัง 4 แล้วเก็บใน `dict_S` ตัวใหม่โดยมี key = 'b'
3. นำ value ของ key 'c' มา ลบ 15 แล้ว ยกกำลัง 2 แล้วเก็บใน `dict_S` ตัวใหม่โดยมี key = 'c'
4. นำค่าที่ได้ทั้งหมด รวมกัน แล้วเก็บใน `dict_S` ตัวใหม่โดยมี key = 'x'

**Output:**

แสดงผล `dict_S`

**Hint:**

- เข้าถึง value แต่ละตัวด้วยการ assign ค่า เช่น เข้าถึงค่าของ key a โดยการกำหนด `a = x['a']`

ตัวอย่าง

Input	Output
ไม่มี input	{'a': 999, 'b': 4096.0, 'c': 1225, 'x': 6320.0}

# Squares

Input:

รับค่าจำนวนเต็มบวกที่มากกว่า 2 ผ่านทาง input เก็บไว้ในตัวแปร n ถ้ารับค่า  $n \leq 2$  ให้แสดงผล invalid

Process:

loop ค่าตั้งแต่ 0 ถึง  $n+5$  (ไม่นับค่า  $n+5$ ) โดยค่า item แต่ละตัวที่ loop เก็บไว้ในตัวแปร i

สร้าง dict ใหม่ โดย

1. มี key = i
2. มี value = i ยกกำลัง 2

Output:

แสดงผล dict ที่สร้างใหม่

Hint:

- ใช้ range() เพื่อกระจายค่าตั้งแต่ 0 ถึง  $n+5$

ตัวอย่าง

Input	Output
5	{0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81}
2	invalid

# Random exam answer

Adrian, Bruno, Goran ต้องทำข้อสอบเป็นจำนวน N ข้อ แต่ละข้อมี 3 ตัวเลือกคือ A, B, C ทั้ง 3 คนต้องการเดาคำตอบที่ถูกต้อง โดยมีวิธีดังนี้

- Adrian จะทำข้อสอบด้วยรูปแบบ A, B, C, A, B, C, A, B, C, ...
- Bruno จะทำข้อสอบด้วยรูปแบบ B, A, B, C, B, A, B, C, B, A, ...
- Goran จะทำข้อสอบด้วยรูปแบบ C, C, A, A, B, B, C, C, A, A, B, B, ...

เขียนโปรแกรม รับเฉลยข้อสอบ และหาว่า วิธีของใครจะได้คะแนนมากที่สุด

Input:

- บรรทัดที่ 1 รับจำนวนคำถามในข้อสอบทั้งหมด เป็นจำนวนเต็มบวก โดย  $N \geq 1$
- บรรทัดที่ 2 รับข้อความความยาว N ที่ประกอบด้วยตัวอักษร 'A', 'B', 'C' ที่เป็นเฉลยของคำถามในแต่ละข้อเรียงกันตามลำดับ

Output:

- บรรทัดที่ 1 แสดงคะแนนที่มากที่สุด
- บรรทัดที่ 2 แสดงชื่อคนที่ได้คะแนนสูงสุดถ้ามีหลายคนให้แสดงบรรทัดละหนึ่งชื่อ

Hint:

- ใช้ dictionary ในการเก็บข้อมูลคะแนนของแต่ละคน
- คำนวณหา pattern ที่แต่ละคนใช้ จากการคูณ list กับผลต่างจำนวนข้อ

ตัวอย่าง

Input	Output
9 AAAABBBBBB	4 Adrian Bruno Goran
5 BABCB	5 Bruno
3 CCA	3 Goran
4 ABCA	4 Adrian