Averill Cate

1. Given a list like myList=[1,2,3,4], your task is to find sum of each number with another number, i.e., 1+2,1+3,1+4,2+3,2+4,3+4 . Write two codes, one using list comprehension and other using for loop.

In [174]:
```
myList = [1, 2, 3, 4]

# As a for-loop
sums = []
for i in myList[0:-1]:
    for j in myList[i:]:
        sums.append(i + j)
print(sums)

# As a list comprehension
sums = [i + j for i in myList[0:-1] for j in myList[i:]]
print(sums)
```

```
[3, 4, 5, 5, 6, 7]
[3, 4, 5, 5, 6, 7]
```

2. Given a list write a code to detect if there is a duplicate element present in the list (of integers) or not. Print yes or no. Write two codes, one using '==' operator and other using exclusive or operator '^'.

In [175]:
```
lst1 = [1, 2, 3, 4, 5, 2]
lst2 = [1, 2, 3, 4, 5,]

dupes = [i for i in lst1 if lst1.count(i) > 1]
print("yes") if len(dupes) > 1 else print("no")

dupes = [i for i in lst2 if lst2.count(i) > 1]
print("yes") if len(dupes) > 1 else print("no")

def has_dupes(l):
    v = [i^j == 0 for i in l[:-1] for j in l[i:]]
    return v

print("yes") if True in has_dupes(lst1) else print("no")
print("yes") if True in has_dupes(lst2) else print("no")
```

```
yes
no
yes
no
```

3. Given below is a 2D matrix, create a list of all the odd numbers present in the matrix. Also sort the list in descending order.

In [176]:
```
myMatrix = [[1, 2, 3, 4], [5, 6, 7], [8, 9, 10]]
odds = [c for r in myMatrix for c in r if c % 2 != 0]
odds.sort(reverse=True)
print(odds)
```

```
[9, 7, 5, 3, 1]
```

4. Given below is a 2D matrix, create a list of squares of all the even numbers present in the matrix.

In [177]:
```
myMatrix = [[1, 2, 'aa',3, 4], ['dd',5, 6, 7], [8, 9, 10,'cc']]
squares = [c**2 for r in myMatrix for c in r if isinstance(c, int) an
d c % 2 == 0]
print(squares)
```

```
[4, 16, 36, 64, 100]
```

5. Given below is a 2D matrix, create a list of squares of all the prime numbers present in the matrix. (Hint: use 6k+1 or 6k-1 formula)

In [178]:
```
"""
Been a while since I worked with 6k +/- 1, so I had to look it up.  F
or academic integrity sake, here is the
url reference:
https://stackoverflow.com/questions/1801391/what-is-the-best-algorith
m-for-checking-if-a-number-is-prime
"""
def is_prime(n):
    if n in [2, 3]:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False

    i = 5
    j = 2

    while i * i <= n:
        if n % i == 0:
            return False

        i += j
        j = 6 - j

    return True

myMatrix = [[21, 22, 23, 4, 16, 17, 18, 19], [5, 6, 7, 14, 15, 20, 1,
 2, 3], [8, 9, 10, 11, 12, 13]]
primes = [c for r in myMatrix for c in r if is_prime(c)]
print(primes)
```

```
[23, 17, 19, 5, 7, 1, 2, 3, 11, 13]
```

6. Make a dictionary of all those words, from the given paragraph, which are having 4 or more characters in it . Key of the dictionary should be word and value should be the number of times that word has appeared in the paragraph. eg. {"feminist":3,"part":2,"campaign":1}

In [179]:
```
mySentence=r"""It's the Spice Girls but not as you know them. Twenty
 years after it was first released, this famous girl power anthem has
 been given a 21st century feminist makeover. The
new video is part of Project Everyone's campaign to improve the lives
 of women and girls everywhere, calling for an end to violence agains
t girls, quality education for all and equal pay for equal work."""

sentence = mySentence.replace(".", "").replace(",", "")
# List of words greater then 4 chars.
words =[w for w in sentence.split() if len(w) > 3]
print(words)

wd = {}
for w in words:
    wd[w] = wd.get(w, 0) + 1
print("\nWord count dictionary:")
print(wd)
```

```
["It's", 'Spice', 'Girls', 'know', 'them', 'Twenty', 'years', 'afte
r', 'first', 'released', 'this', 'famous', 'girl', 'power', 'anthem',
'been', 'given', '21st', 'century', 'feminist', 'makeover', 'video',
'part', 'Project', "Everyone's", 'campaign', 'improve', 'lives', 'wom
en', 'girls', 'everywhere', 'calling', 'violence', 'against', 'girl
s', 'quality', 'education', 'equal', 'equal', 'work']

Word count dictionary:
{"It's": 1, 'Spice': 1, 'Girls': 1, 'know': 1, 'them': 1, 'Twenty':
1, 'years': 1, 'after': 1, 'first': 1, 'released': 1, 'this': 1, 'fam
ous': 1, 'girl': 1, 'power': 1, 'anthem': 1, 'been': 1, 'given': 1,
'21st': 1, 'century': 1, 'feminist': 1, 'makeover': 1, 'video': 1, 'p
art': 1, 'Project': 1, "Everyone's": 1, 'campaign': 1, 'improve': 1,
'lives': 1, 'women': 1, 'girls': 2, 'everywhere': 1, 'calling': 1, 'v
iolence': 1, 'against': 1, 'quality': 1, 'education': 1, 'equal': 2,
'work': 1}
```

7. Given a list, multiply all the elements of the list by 2 without using any arithmetic operator. Hint: use bitwise operator

In [180]:
```
# Re-using myList
myList

mults = [i << 1 for i in myList]
print(mults)

alt_list = range(1, 20)
mults = [i << 1 for i in alt_list]
print(mults)
```

```
[2, 4, 6, 8]
[2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36,
38]
```

8. Given below are two 2D matrix, add them element wise to form a third 2D matrix and print the resultant matrix.

In [181]:
```
mtx1 = [[1, 2, 3, 4], [5, 6, 7, 6], [8, 9, 10,4]]
mtx2 = [[3, 1, 1, 4], [7, 7, 7, 7], [8, 9, 10,11]]

# Combined
mtx3 = [mtx1[i] + mtx2[i] for i in range(len(mtx1))]
print(mtx3)

# Added
print("Summing the elements:\n")
mtx4 = []
for r in range(len(mtx1)):
    s = []
    for c in range(len(mtx1[r])):
        s.append(mtx1[r][c] + mtx2[r][c])
    mtx4.append(s)

print(mtx4)
```

```
[[1, 2, 3, 4, 3, 1, 1, 4], [5, 6, 7, 6, 7, 7, 7, 7], [8, 9, 10, 4, 8,
9, 10, 11]]
Summing the elements:

[[4, 3, 4, 8], [12, 13, 14, 13], [16, 18, 20, 15]]
```