Decision Point Analytics - Python Assesement

In [1]:

```
import pandas as pd
```

In [2]:

```
#Hardcoded input
l=['a','b','c','d','e','f','g','h']
```

Problem

```
    Problem: Here we need to pick 5 unique letters repeatedly (without replacement) and
to frame all possible patterns
        without considering any order.

    Note: It's nothing but we should find different combinations of 5 digit words with no
letter repeated twice or more
        for single combination.

    So, the maximum possible combinations would be like:
        8(Intially we can 8 possible letters to pick from the list) *
        7(after picking 1 letter we left with 7 only ) *
        6(after picking 2 letters we have only 6 unique letters) *
        5(we left with 5 unique letters to pick) *
        4(we have only 4 letters so as to make 5 digit sequence)       = 6720

    In total we will get 8*7*6*5*4 = 6720 combinations without any letter repeated in the
sequence.
```

Approach

```

    Approach: Here i am going with nested for loops(neglecting Time complexity as of now)
    * first I am picking 1 element from list l as a:
          for that I am connecting with same list l with element name b:
              this I am repeating for 5 times because we need 5 letter word.

    * After iterating through all the loops we concating each and every letter from
single "for"
    loop and creating 5 digit sequence.
    * then we will get many such sequences with letters repeated 1 or more number of
times.
    * then we have have to eleminate sequences with duplicate letters if there exists
any.
    * for that I have performed a check and took only unique lettered sequence and stored
in list.
    * finally I have converted the stored string into dataframe.

```

In [3]:

```
#declaring empty list to store output sequence
arr=[]
#iterating through the loops
for a in l:
    for b in l:
        for c in l:
            for d in l:
                for e in l:
                    #taking only unique values
                    if a!=b and b!=c and c!=d and d!=e and e!=a:
                        res=a+b+c+d+e
                        #filtering out if still I had any other duplicate records
                        if len(set(list(res)))==5:
                            arr.append(res)
print(len(arr))
```

```
6720
```

In [8]:

```
#Sample output
arr[:20]
```

Out[8]:

```
['abcde',
 'abcdf',
 'abcdg',
 'abcdh',
 'abced',
 'abcef',
 'abceg',
 'abceh',
 'abcfd',
 'abcfe',
 'abcfg',
 'abcfh',
 'abcgd',
 'abcge',
 'abcgf',
 'abcgh',
 'abchd',
 'abche',
 'abchf',
 'abchg']
```

In [4]:

```
#finding unique sequence count
len(set(arr))
```

Out[4]:

```
6720
```

In [5]:

```
#declaring empty list to store output sequence
l2=[]
for i in arr:
    #storing strings in list format into a new list
    l2.append(list(i))
```

In [6]:

```
#converting multi dimensional list into dataframe and assigning desired column names
output = pd.DataFrame(l2,columns=['I','II','III','IV','V'])
#showing first five rows of data
output.head()
```

Out[6]:

	I	II	III	IV	V
0	a	b	c	d	e
1	a	b	c	d	f
2	a	b	c	d	g
3	a	b	c	d	h
4	a	b	c	e	d

In [7]:

```
#final dataframe we got has 6720 different unique patterns of strings of length 5
output.shape
```

Out[7]:

```
(6720, 5)
```

Thank you