Q1. Append two list in python

list\_fn = ["sangeetha", "sindhu", "varshika", "jyosthna"]

list\_ln = ["r", "s", "v", "d" ]

list\_n = []

for i in range(len(list\_fn)):

for j in range(len(list\_ln)):

if i == j:

list\_n.append(list\_fn[i] + list\_ln[j])

print(list\_n)

Q2. Calculate Previous month, Previous Quarter, Future, Present Quarter with an inputed date and quarter

def example(a , b):

dictionary = {"Previous month" : " ", "Previous Quarter": " ", "Future format" : " ", "Present Quarter" : " "}

month = a % 100

year = int(str(a)[:4])

if month > 1:

pr\_mon = month - 1

f\_year = year

month1 = month - 1

year1 = year + 1

else:

pr\_mon = 12

f\_year = year - 1

month1 = 12

year1 = year

pr\_year = year - 1

#if pr\_mon < 10:

pr\_mon = "{:02d}".format(pr\_mon)

f\_year = str(f\_year)

pr\_fo = f\_year+str(pr\_mon)

#print("previous month -",pr\_fo) #previous month

def qu\_cal(m,y):

if m in range(1,4):

quarter = 'Q1'

pr\_q = 'Q4'

year2 = year - 1

elif m in range(4,7):

quarter = 'Q2'

pr\_q = 'Q1'

year2 = year

elif m in range(7,10):

quarter = 'Q3'

pr\_q = 'Q2'

year2 = year

elif m in range(10,13):

quarter = 'Q4'

pr\_q = 'Q3'

year2 = year

else:

print("error")

p2 = str(pr\_q+'\_'+str(year2))

return quarter,p2

p\_q,p2 = qu\_cal(month,year)

#print("previous quarter",p2)

if month1 < 10:

month1 = "{:02d}".format(month1)

future = str(year1) + str(month1)

#print("Future date -",future)

d = str(p\_q+'\_'+str(year))

#print("present quarter -",d) #present Quarter

dictionary['Previous month'] = pr\_fo

dictionary['Previous Quarter'] = p2

dictionary['Future format'] = future

dictionary['Present Quarter'] = d

return dictionary

print(example(202109, 'Q3'))

Q3. Employee and Manager levels printing

WITH RECURSIVE

emphierarchy (employee\_id, employee\_name, manager\_id, Emplevel)

AS ( SELECT employee\_id, employee\_name, manager\_id, 0 as Emplevel

FROM employee

WHERE employee\_id=1

UNION

select employee\_id, employee\_name, manager\_id, 100 as Emplevel

from Employee

where employee\_id!=1 and employee\_id=manager\_id

UNION

SELECT e.employee\_id, e.employee\_name, e.manager\_id,(h.EmpLevel + 1) as Emplevel

FROM Employee as e

INNER JOIN emphierarchy as h

ON h.employee\_id = e.manager\_id

where e.manager\_id!=e.employee\_id

)

SELECT

\* from emphierarchy

where EmpLevel<=3 and employee\_id!=1;

Q4. Date differents between months

SELECT s1.cust\_id, s1.dat as up,

(((round((s1.dat/100),0) - round((MIN(s2.dat)/100),0))\* 12) + (s1.dat%100 - MIN(s2.dat)%100)) as months\_diff

FROM sales AS s1

JOIN sales AS s2 ON s1.cust\_id = s2.cust\_id

GROUP BY s1.cust\_id, s1.dat order by s1.dat ASC;

Q5. Consequtive 2 week gap between products

Q6. Write 5 - 10 date, string, Int, Pandas, Numpy, RDD, Spark Session object functions

A close up of a book

Description automatically generated with low confidence

A close-up of a notebook

Description automatically generated with low confidence

A hand holding a piece of paper

Description automatically generated with medium confidence

A hand holding a piece of paper

Description automatically generated

A close-up of a book

Description automatically generated with low confidence

Q7. Use Posexplode and collect list

%sql

select cust\_id, t.products, k.price from customer lateral view posexplode(products) t as pos, products

lateral view posexplode(price) k as poss, price where t.pos = k.poss;

%sql

select cust\_id,collect\_list(products), collect\_list(price) from customer1 group by cust\_id;