Python -> (A to Z)

```
print("hello")
print("kfbgjfdrn")

#arithmetic operators
# +,-,*,/, %, //, **

print(2+3)
print(2-3)
print(2*3)
print(13/3) #- Quotient - division
print(13%3)  # remainder  - 0 - modulus
print(13//3)   #  whole number quotient - 4
print(3**2)  # - exponentiation - raise to power


#relational operators -
#<,>,<=,>=,!=,==

print(12 == 3)

#assignment

a = 10

#logical operators
# and, or ,not

# and
#           and    or
# 1      0   0      1
# 0      1   0      1
# 1      1   1      1
# 0      0   0      0

print((2>3) or (13>5))

a = 10
b = 20
print("Sum of the numbers is ", a+b)

a = int(input("Enter the value of a:"))
b = int(input("Enter the value of b:"))
print("Sum of the numbers is ", a+b)

a = 'n'
```

```
b = 'm'
print(a+b)

side = float(input("Enter the length of side of square: "))
area = side**2
print("Perimeter of the square with side ", side, "is",area)

...
                                    # (Assignment-01 )#

...

#(Question-01)#

a = 7
b = 2
# addition
print ('Sum: ', a + b)
# subtraction
print ('Subtraction: ', a - b)
# multiplication
print ('Multiplication: ', a * b)
# division
print ('Division: ', a / b)
# floor division
print ('Floor Division: ', a // b)
# modulo
print ('Modulus: ', a % b)
# a to the power b
print ('Power: ', a ** b)

#(Question-02)# Python program to swap two variables

x = 5
y = 10
# To take inputs from the user
#x = input('Enter value of x: ')
#y = input('Enter value of y: ')
# create a temporary variable and swap the values
temp = x
x = y
y = temp
print('The value of x after swapping: {}'.format(x))
print('The value of y after swapping: {}'.format(y))
#For example
x = 5
y = 10
x, y = y, x
```

```
print("x =", x)
print("y =", y)

#(Question-03)#

w = 22
x = 52
y = 79
print(w+y)
print(y -w)
print(y % x)
print(y // w)
print(y ** w)
print(w * x)
print(y / x)

#(Question-04)# This program adds two numbers

num1 = 1.5
num2 = 6.3
# Add two numbers
sum = num1 + num2
# Display the sum
print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
# Store input numbers
num1 = input('Enter first number: ')
num2 = input('Enter second number: ')
# Add two numbers
sum = float(num1) + float(num2)
# Display the sum
print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))

#(question-05)#

# Python program to check if the input number is odd or even.
# A number is even if division by 2 gives a remainder of 0.
# If the remainder is 1, it is an odd number.
#Even
num = int(input("Enter a number: "))
if (num % 2) == 0:
   print("{0} is Even".format(num))
else:
   print("{0} is Odd".format(num))
   #Odd
num = int(input("Enter a number: "))
if (num % 2) == 0:
   print("{0} is Even".format(num))
else:
```

```
   print("{0} is Odd".format(num))

#(Question-06)#

a = True
b = False
print(not(a))
print(not(b))

a = False
b = False
x = not(a)
y = not(b)
print(a or b)
print(x or y)
print(a or x)
print(x or b)
print(y and b)

a = False
b = False
x = not(a)
y = not(b)
print(a and b)
print(a and x)
print(x and y)

#(Question-08)

age = int(input('Please enter your age in years: '))
income = int(input('Please enter your annual income: '))
if age >= 21 and income >=21000:
   print('You can apply for a loan')
else:
   print('You CANNOT apply for a loan')

#(question-11)#

w = 20
x = 10
y = 15
z = 2
result_1 = (w+x)*y/z
result_2 = ((w+x)*x)/z
result_3 = ((w+x)*(y/z))**z
result_4 = w+(x*y)/z
print('The value of (w+x)* y/z is',result_1)
print('The value of ((w+x)*x)/z is',result_2)
```

```
print('The value of ((w+x)*(y/z))**z is',result_3)
print('The value of w+(x*y)/z is',result_4)

##(Croma Campus tuday)

num = int(input("Enter the number: "))

if num%2 == 0:

   print(num**2)

elif num%3 == 0:

   print(num**3)

else:

   print(num,"it is not a multiple of both 2 and 3")
```

[9:58 AM] Nikhil Rana [Croma Campus]

```
n = int(input("Enter the number:"))

if n>0:

   print("Positive")

else:

   print("negative")


amt = int(input("Enter the amount: "))
if amt>=5000:
   print("The discount is 12.5%")
   disc = amt*.125
   print("The amount of discount is", disc)
   amount_payable = amt - disc
   print("The amount to be paid is", amount_payable)
elif amt> 4000 and amt<5000:
   print("The discount is 10%")
   disc = amt*.1
   print("The amount of discount is", disc)
   amount_payable = amt - disc
   print("The amount to be paid is", amount_payable)
else:
   print("No discount is applicable.")
   print("The amount to be paid is", amt)
```

##

person - category
0-4 - toddler
5-10 - child
11-19 - teenager
20-45 - Adult
46-65 - senior adult
>65 - Senior Citizen/

```
 #age category program
age=int(input("Enter your age"))
if age<=4:
  print("toddler")
elif age<=10:
  print("Child")
elif age<=19:
  print("teenager")
elif age<=45:
  print("adult")
elif age<=65:
  print("Senior Adult")
else:
  print("Senior CItezen")
```

```
Age = int(input('Enter the age of person:'))
if Age>0 and Age<=4:
   print("person is Toddler")
elif Age>5 and  Age<=10:
   print("Person is child")
elif Age>11 and  Age<=19:
   print("Person is teenager")
elif Age>20 and Age<=45:
   print("Person is Adult")
elif Age>46 and Age<=65:
   print("Person is Senior Adult")
elif Age>45 and Age>=65:
   print("Senior citizen")
```

```
else:
   print("no person")



bill = float(input("Enter the bill: "))
deadline = int(input("Enter the date:"))
if deadline <= 25:
   if bill < 1000:
      print("No cashback.")
      print("Bill to be paid is:", bill)
   elif bill > 1000 and bill <=5000:
      print("You have received a cashback of 1.2% upto INR 30.")
      cb = bill*.012
      if cb>30:
         print("You have received a cahback of INR 30")
         bill_paid = bill - cb
         print("Bill to be paid is:", bill_paid)


marks_1 = float(input("Enter the marks: "))
marks_2 = float(input("Enter the marks in the test: "))
if marks_1>=65 and marks_2>=60:
   print("Eligible")
else:
   print("Not eligible")



credit card
<1000, within the deadline - no cashback
>1000  -  <5000 - within the deadline - 1.2% upto INR 30
>5000  -  <10000 - within the deadline - 1.5% upto INR 60
>10000 - within the deadline - 1.5% upto INR 70

deadline has passed - penalty of 10

#Credit card discount and penalty
amt=float(input("Enter bill amount :"))
dl=int(input("Enter day of bill payment"))
if dl<25:
  if amt<1000:
    disc=0
    print("you pay",amt-disc)
  elif amt<5000:
    if amt*1.2/100<=30:
      disc=amt*1.2/100
      print("you pay",amt-disc)
```

```
      else:
        disc=30
        print("you pay",amt-disc)
    elif amt<10000:
      if amt*1.5/100<=60:
        disc=amt*1.5/100
        print("you pay",amt-disc)
      else:
        disc=amt=60
        print("you pay",amt-disc)
    else:
      if amt*1.5<=70:
        disc=amt*1.5/100
        print("you pay",amt-disc)
      else:
        disc=70
        print("you pay",amt-disc)
else:
  disc=-10
  print("you pay",amt-disc
```

```
#Credit card discount and penalty

amt=float(input("Enter bill amount :"))

dl=int(input("Enter day of bill payment"))

if dl<25:
    if amt<1000:
        disc=0
        print("you pay",amt-disc)
    elif amt<5000:
        if amt*1.2/100<=30:
            disc=amt*1.2/100
            print("you pay",amt-disc)
        else:
            disc=30
            print("you pay",amt-disc)
    elif amt<10000:
        if amt*1.5/100<=60:
            disc=amt*1.5/100
            print("you pay",amt-disc)
        else:
            disc=amt=60
            print("you pay",amt-disc)
```

```
    else:
        if amt*1.5<=70:
            disc=amt*1.5/100
            print("you pay",amt-disc)
        else:
            disc=70
            print("you pay",amt-disc)

else:
    disc=-10
    print("you pay",amt-disc)
```

person has to apply for visa
1. Adhaar card
2. Age > 18
3. IELTS score > 6.5
4. scored more than 50% in 10+2
5. passport number

Interview Question 08/01/2023
##1. what is null value in python in function.
#Use the ISNULL function with the IF statement when you want to test whether the value of a variable is the null value.

##2. how to pass null value in python function.
We can also assign a null value to a variable by using the data type None.

##3. what is median in python function.
#calculate the middle value of a given set of numbers.

##4. what is bar chart and histograph in function.
#The bar graph is the graphical representation of categorical data. A histogram is the graphical representation of quantitative data.

##5. what is disconary in python in function.
#it is a sequence of key-value pairs separated by commas and surrounded by curly braces.

##6. what is array in python in function.
#Arrays are a fundamental data structure containers which are able to store more than one item at the same time.

##7. what is tuple.
#A tuple is a finite sequence or ordered list of numbers or, more generally, mathematical objects, which are called the elements of the tuple.

##8. what is list in python function.
#The list() function creates a list object. A list object is a collection which is ordered and changeable.

##9. what is sheborn and line chart in python function.

##9. what is number of columns and rows in python in function.
#len() The len() function can determine the number of the Dataframe's rows and columns. Rows and columns
#are represented by dataframe. axes[0] and dataframe.

##10. what is outlier function in python.

##11. what is EDA in python.

##12. what is boxplot in python function.

```
from datetime import date

today = date.today()

#Change this to your birth date
date_of_birth = date(2003, 2, 25)

birthday = date(today.year, date_of_birth.month, date_of_birth.day)
days_until_birthday = (birthday-today).days
days_alive = (today-date_of_birth).days

print( 'You are ' + str(days_alive) + ' days old')

if days_until_birthday > 0:
        print( 'It\'s ' + str(days_until_birthday) + ' days until your Birthday')
elif days_until_birthday == 0:
        print( 'Happy Birthday!')
else:
        print( 'You\'ll have to wait until next year for another birthday')
```

```
# python program to display calendar of
# given month of the year

# import module
import calendar
yy = 2023
mm =11
# display the calendar
```

```
print(calendar.month(yy,mm))

# Python code to demonstrate the working of
# calendar() function to print calendar
# importing calendar module
# for calendar operations
import calendar
# using calendar to print calendar of year
# prints calendar of 2023
print("The calendar of year 2023 is : ")
print(calendar.calendar(2023))
```

$$###@@@
@@@@@@@@@@@@@@@@@@@&&&&&&&&&&&&&&&&&&&&&&&&&&&********************
*****

EXCEL

$$###@@@
@@@@@@@@@@@@@@@@@@@&&&&&&&&&&&&&&&&&&&&&&&&&&&********************
*****

((Ipl 2008-2022 in Power bi))

```
Average by bowler = DIVIDE(
   sumx(
      FILTER(ipl_ball_by_ball_2008_2022, ipl_ball_by_ball_2008_2022[extra_type]
                          <>"legbyes" && ipl_ball_by_ball_2008_2022[extra_type]<> "byes"),
ipl_ball_by_ball_2008_2022[total_run]),
                          (COUNT(ipl_ball_by_ball_2008_2022[overs]))/6)


                          Batter Runs = CONCATENATE(SUM(ipl_ball_by_ball_2008_2022[batsman_run])," Runs")

                          Bowler Wickets = CONCATENATE(SUM(ipl_ball_by_ball_2008_2022[iswicket_delivery]),"
Wickets")

                          Bowling SR =
COUNT(ipl_ball_by_ball_2008_2022[bowler])/SUM(ipl_ball_by_ball_2008_2022[iswicket_delivery])

                          Economy = DIVIDE(
   sumx(
      FILTER(ipl_ball_by_ball_2008_2022, ipl_ball_by_ball_2008_2022[extra_type]<>"legbyes"
                          && ipl_ball_by_ball_2008_2022[extra_type]<> "byes"),
ipl_ball_by_ball_2008_2022[total_run]),
```

(COUNT(ipl_ball_by_ball_2008_2022[overs]))/6)

Strike Rate for Batsman =
SUM(ipl_ball_by_ball_2008_2022[batsman_run])/COUNT(ipl_ball_by_ball_2008_2022[ball_number])*100

Matches Win on toss decision = CALCULATE(COUNTROWS(ipl_matches_2008_2022),
ipl_matches_2008_2022[toss_winner] = ipl_matches_2008_2022[winning_team])

Title Winner = VAR max_date= CALCULATE(MAX('Calender Table'[Date]),
ALLSELECTED(ipl_matches_2008_2022), VALUES(ipl_matches_2008_2022))
VAR tital_winner = CALCULATE(SELECTEDVALUE(ipl_matches_2008_2022[winning_team]), 'Calender Table'[Date] = max_date)
return tital_winner

###$$$
$$$@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@********************************
*

SQL A to Z

##$$
$$$$@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@****************************
**

crete database wd_4

use wd_4

create table t1 (roll_number int, name varchar(50), city varchar(100)

insert into t1 values(1,'Aman','Delhi')

select*feom t1

insert into t1 values(2,'Arun','Gurgaon'),(3,'Aryan','Noida')

select * from t1

update t1 set roll_number = 5 where name = 'Arun'

delete from t1 where name = 'Aryan'

select * from t1

```
alter table t1 add marks_percentage int

update t1 set marks_percentage = 90 where roll_number = 1

select name from t1 where marks_percentage < 80

select *,marks_percentage + marks_percentage*.05 as final_marks from t1

select * from t1 where name like '%a'

select marks_percentage,name from t1 where marks_percentage not in (90,86)

select * from t1 where marks_percentage between 86 and 100use wd_4

select * from t1

drop table t1

drop database wd_4

truncate table tselect t1.cl_name, t2.col_name,t3.col_name from t1 inner join t2 on t1.id = t2.id inner join t3 on t2.id = t3.id

select * from t1 inner join t2 on t1.roll_number = t2.roll_number
select t1.roll_number, t2.Class from t1 inner join t2 on t1.roll_number = t2.roll_number
select t1.cl_name, t2.col_name,t3.col_name from t1 inner join t2 on t1.id = t2.id inner joint t3 on t2.id = t3.id


select * from t1 cross join t2

--constraints

create table t3 (id int unique not null, name varchar(100) not null, Salary int not null)

--primary key

create table t4 (empid int primary key identity(101,5), empname varchar(100), department varchar(100))

--foreign key

--t1 - all employees data
--t2 - account department data

create table t5 (id int, address varchar(200) constraint fk foreign key (id) references t3 (id))

create table <table_name> (col1, col2 consraint fk foreign key (child table col) references parent_table(parent table col))
```

```
select sum(list_price) from production.products
select min(list_price) from production.products
select max(list_price) from production.products
select avg(list_price) from production.products
select count(product_id) from production.products

select sin(90)
select len('fgjbfjbngnf')
select len(product_name) from production.products
select CEILING(10.2)
select floor(10.2)
select log(10)
select LOG10(10)
select EXP(4)
select sqrt(9)
select concat('ankit','kumar')
select concat_ws('_','ankit','kumar')
select trim ('    ksnbksnfkgbdg      ')
select ltrim('    ksnbksnfkgbdg      ')
select rtrim('    ksnbksnfkgbdg      ')
select getdate()
select sysdatetime()
select datepart(month,'2022-09-04')
select datepart(day,'2022-09-04')
select datepart(year,'2022-09-04')
select datediff(month,'2022-02-14',getdate))
select datediff(day,'2022-02-14',getdate))
select datediff(year,'2022-02-14',getdate))
select dateadd(day,16,getdate())
select dateadd(month,16,getdate())
select dateadd(year,16,getdate())


((question noumber (1))  create production id, name,qquantity.....

select production.products.product_id,production.products.product_name, production.stocks.quantity from
production.products inner join production.stocks on production.products.product_id =
production.stocks.product_id


select datename(dw,Order_Date), Order_Date from sales.orders
select datepart(dw,Order_Date), Order_Date from sales.orders
select order_Date, Shipped_Date, datediff(dd,order_Date,shipped_Date) from sales.orders
select order_Date, Shipped_Date, datediff(ww,order_Date,shipped_Date) from sales.orders
select order_Date, Shipped_Date, datediff(mm,order_Date,shipped_Date) from sales.orders
```

```
select order_Date, Shipped_Date, datediff(yy,order_Date,shipped_Date) from sales.orders
```

dssb

```
select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) = 2;

 select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) < -2;

select order_Date, Shipped_Date, datediff(dd,order_Date,shipped_Date)   from sales.orders
 where datediff(dd,order_Date,shipped_Date) > 2;

 select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) < -2;

  select Datediff(dd,required_date,shipped_date)as late_delivery, sales.order_items.order_id,
  sales.order_items.product_id,production.products.product_name from
  sales.orders inner join sales.order_items on sales.orders.order_id = sales.order_items.order_id inner join
production.products on
  production.products.product_id = sales.order_items.product_id  where
Datediff(dd,required_date,shipped_date)>=2
```

1. Display all the managers.

```
select empname from tblemployee where empdesignation='manager'
```

2. Display all the employees except managers.

```
select empname from tblemployee where empdesignation not in ('manager')
```

3. Display the salary of employee 'Mathew'.

```
select empsalary from tblemployee where empname='mathew'
```

4. Display the manager of department 'D001'.

```
select empname from tblemployee where deptid= 'D001' and empdesignation='manager'
```

5. Display total number of employees in department 'D003'.

```
select empname from tblemployee where deptid='d003'
```

6. Display the grade of salary '33000'.

```
select empsalary from tblemployee where empsalary=33000
```

7. Display the average salary of 'D006' department.

```
select avg ( empsalary ) from  tblemployee where empid='D006'
```

8. Display the emplyee who taking maximum salary.

```
select empname ,empsalary from tblemployee where empsalary =(select max (empsalary) from tblemployee)
```

9. Display employees who have salary between 25000 and 35000.

select empsalary from tblemployee where empsalary =(25000-35000)

10. Display all the employees with 10 percent increment in their salary.

select empsalary+empsalary *.1 as incremented_salary from tblemployee

11. Make permanent changes with 10 percent increment of every employee salary.

12. Give the 20 percent increment of employee with id 'E0005'.

13. Display employee records whos name start from 'M'.

select empname from tblemployee where empname like 'M%'

14. Display employee records whos name start from 'M' and end with 'W'.

select empname from tblemployee where empname like 'M%w'

15. Display employees whos name is 5 characters long.

select empname from tblemployee where len(empname)=5

16. Display employees whos name contains 5 or more characters.

select empname from tblemployee where len(empname) between 5 and 100

17. Display employee records whos empEmail contains _ . ex: abc_11@example.com

select * from tblemployee where empemail= 'brian@yahoo.com'

18. Display employees with palindrom name like 'nitin'.

22. Display the empId, empname and Age of employee

select empname,empid,empdob from tblemployee

23. Display employees who are elder than their manager.

24. Display deptId, deptName and no. of employees in that particular department.

select deptid, deptname from tbldepartment

25. Display the 2nd highest salary.

26. Display all employees name along with their departments.

27. Display all the employees who born in 'March' month along with their department.

select empdob,empname from tblemployee inner join where DATEPART(month,(empdob))=3

=IF(RIGHT(LEFT($A$3:$A$45,3),2)="fc","facility",IF(RIGHT(LEFT($A$3:$A$45,3),2)="st","staff"))

Croma Campus to Everyone 2:53 PM

=IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$6,Lookups!$C$6,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$1,Lookups!$C$1,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$2,Lookups!$C$2,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$3,Lookups!$C$3,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$4,Lookups!$C$4,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$5,Lookups!$C$5,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$7,Lookups!$C$7,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$8,Lookups!$C$8,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$9,Lookups!$C$9,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$10,Lookups!$C$10,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$11,Lookups!$C$11))))))))))

Anshul to Everyone 3:01 PM

sir voice bht km ari h

Croma Campus to Everyone 3:19 PM

=VLOOKUP($G44,Lookups!$I:$J,2,0)

Croma Campus 3:46 PM

=SUMIFS(I$3:I$45,$E$3:$E$45,$E$50,$F$3:$F$45,$F50)

1--fetch the name in the format : FirstName-MiddleName-LastName of those customers with Address type as 'Main Office' .

```
Select CONCAT_WS (' - ',FirstName,MiddleName,LastName) as full_name from SalesLT.Customer where CustomerID in
(select CustomerID from SalesLT.CustomerAddress where AddressType ='main office')
```

2--fetch the product ids of those products with orderqty between 5 and 10 with any discount.

```
 select ProductId, OrderQty, UnitPriceDiscount from SalesLT.SalesOrderDetail where OrderQty between 5 and 10 and
 UnitPriceDiscount >0
```

3--fetch the details of those orders which are shipped more than 2 days late..

```
select * from salesLT.salesorderheader where datediff (day,shipdate,duedate)>=2
```

4--fetch the addresstype of the customers of Canada region

```
select addresstype from SalesLT.CustomerAddress where AddressID in (select AddressID from SalesLT.Address where CountryRegion like 'canada')

SELECT salesLT.CustomerAddress.AddressType,SalesLT.Address.CountryRegion from SalesLT.CustomerAddress
inner join SalesLT.Address on SalesLT.Address.AddressID = SalesLT.CustomerAddress.AddressID
where CountryRegion = 'canada'
```

5--how may products are without any parentproductid?

```
select
SalesLT.Product.Name,SalesLT.ProductCategory.ProductCategoryID,SalesLT.ProductCategory.ParentProductCategoryID
from SalesLT.Product inner join SalesLT.ProductCategory on SalesLT.Product.ProductCategoryID =
SalesLT.ProductCategory.ProductCategoryID
WHERE ParentProductCategoryID != ProductID
```

6--fetch the name,standardcost, difference between standardcost and list price as price_difference and salesorderid of the products with back color

```
select name,StandardCost ,ListPrice-StandardCost as price_diff,SalesOrderID
from SalesLT.Product inner join SalesLT.SalesOrderDetail on
SalesLT.Product.ProductID=SalesLT.SalesOrderDetail.ProductID
where Color='black'
SELECT NAME,standrdcost, listprice-standerdcost as price_diff,saleso where color='black'


select
  city,
  count (customer_id) customer_count
from
  sales.customers
```

```
group by
  city



  declare @n as int
set @n = (select count(*) from production.products)
--set @n = 10
--print @n
if @n>300
        begin
                print @n
                print 'product count is greater than 300'
        end
else
        begin
                print @n
                print'product count is less than 300'
        end



        declare @n as int
set @n = (select count(*) from production.products)
--set @n = 10
--print @n
if @n>300
        begin
                print @n
                print 'product count is greater than 300'
        end
else if @n = 300
BEGIN
        print 'hello'
end
else
        begin
                print @n
                print'product count is less than 300'
        end


ALTER PROCEDURE uspFindProducts(@min_list_price AS DECIMAL,@max_list_price as decimal,@product as
varchar(10))
AS
BEGIN
```

```
    SELECT
        product_name,
        list_price
    FROM
        production.products
    WHERE
        list_price >= @min_list_price and
                            list_price<=@max_list_price
                            and product_name like '%'+@product+'%'
    ORDER BY
        list_price;
END;

exec uspFindProducts 500,1000,'Trek'



create procedure product_quantity
as
begin
            select product_name, quantity from production.products
            inner join production.stocks on production.products.product_id=production.stocks.product_id

            end

exec product_quantity


create procedure propro
as
begin
select
            product_name
            from
            production.products
            order by
            product_name
end

exec propro



declare @s as int = 0
while (@s<=5)
begin
            print @s
            set @s = @s+1
```

```
end


declare @s as int = 0
while (@s<=10)
begin
	set @s = @s+1
	if @s = 5
		continue
	print @s
end

create view let_delivery
as
select Datediff(dd,required_date,shipped_date)as
  late_delivery, sales.order_items.order_id, sales.order_items.product_id,production.products.product_name
  from sales.orders inner join sales.order_items
  on sales.orders.order_id = sales.order_items.order_id inner join production.products
  on production.products.product_id = sales.order_items.product_id where
Datediff(dd,required_date,shipped_date)>=2


   CREATE TABLE sales.rank_demo (
	v VARCHAR(10)
);
INSERT INTO sales.rank_demo(v)
VALUES('A'),('B'),('B'),('C'),('C'),('D'),('E');


SELECT
	v,
	RANK () OVER (
		ORDER BY v
	) rank_no
FROM
	sales.rank_demo;

	CREATE VIEW sales.vw_staff_sales(
   staff_id,
   year,
   net_sales
) AS
SELECT
   staff_id,
   YEAR(order_date),
   ROUND(SUM(quantity*list_price*(1-discount)),0)
FROM
   sales.orders o
```

```
INNER JOIN sales.order_items i on i.order_id = o.order_id
WHERE
   staff_id IS NOT NULL
GROUP BY
   staff_id,
   YEAR(order_date);

            SELECT
   CONCAT_WS(' ',first_name,last_name) full_name,
   net_sales,
   PERCENT_RANK() OVER (
      ORDER BY net_sales DESC
   ) percent_rank
FROM
   sales.vw_staff_sales t
INNER JOIN sales.staffs m on m.staff_id = t.staff_id
WHERE
   YEAR = 2016;

            SELECT
            v,
            NTILE (4) OVER (
                        ORDER BY v
            ) buckets
FROM
            sales.ntile_demo;

CREATE VIEW sales.vw_netsales_brands
AS
            SELECT
                        c.brand_name,
                        MONTH(o.order_date) month,
                        YEAR(o.order_date) year,
                        CONVERT(DEC(10, 0), SUM((i.list_price * i.quantity) * (1 - i.discount))) AS net_sales
            FROM sales.orders AS o
                        INNER JOIN sales.order_items AS i ON i.order_id = o.order_id
                        INNER JOIN production.products AS p ON p.product_id = i.product_id
                        INNER JOIN production.brands AS c ON c.brand_id = p.brand_id
            GROUP BY c.brand_name,
                                    MONTH(o.order_date),
                                    YEAR(o.order_date);

create procedure min1(@year as int)
as begin SELECT
   category_name,
   year,
   qty,
   FIRST_VALUE(category_name) OVER(
```

```
      ORDER BY qty
   ) lowest_sales_volume
FROM
   sales.vw_category_sales_volume
WHERE
   year = @year
end

exec min1 2018
```

Assignmnent (2)

---

1 . How many olympics games have been held?
1। कितने ओलंपिक खेल आयोजित किए गए हैं?

```
select count(distinct(games)) from dbo.athlete_events
```

2 . a) List down all Olympics games held so far.
2 . क) अब तक आयोजित सभी ओलंपिक खेलों की सूची बनाएं।

```
select distinct year , season , city from dbo.athlete_events order by year
```

b) list down all olympics in year-season-city format. (ex: 1960-summer-roma)
बी) साल-मौसम-शहर प्रारूप में सभी ओलंपिक को सूचीबद्ध करें। (उदा: 1960-ग्रीष्म-रोमा)

```
        select distinct concat_ws('-',year,season, city) all_olympic from athlete_events
```

3 . Mention the total no of nations who participated in each olympics game?
3। प्रत्येक ओलंपिक खेल में भाग लेने वाले देशों की कुल संख्या का उल्लेख करें?

```
select games,count(distinct(noc)) as total_countries from dbo.athlete_events group by games order by
total_countries
```

4 . Which year saw the highest and lowest no of countries participating in olympics?
4। किस वर्ष ओलंपिक में भाग लेने वाले देशों की उच्चतम और निम्नतम संख्या देखी गई?

```
 select COUNT(distinct(team)) as nation,Year from dbo.athlete_events group by Year,Season having year =
MAX(year) order by nation desc
 select COUNT(distinct(noc)) as nation,Year from dbo.athlete_events group by Year having year = Min(year)
order by nation
```

5 . Which nation has participated in all of the olympic games?
5। किस देश ने सभी ओलंपिक खेलों में भाग लिया है?

```
with c as
(select noc,games
```

```
from athlete_events group by noc,games
)
select noc,count(games) from c
group by noc
having count(games) = (select count(distinct games) from athlete_events)

with nations as (select NOC,year,season from athlete_events group by Year,NOC,season)
select NOC from nations group by NOC having COUNT(NOC)=51
```

6 . Identify the sport which was played in all summer olympics.
6 . उस खेल की पहचान करें जो सभी ग्रीष्मकालीन ओलंपिक में खेला गया था।

```
select sport, Season from dbo.athlete_events where Season like 'summer'
group by Sport,Season order by Sport
```

7 . Which Sports were just played only once in the olympics?
7। कौन से खेल ओलंपिक में केवल एक बार खेले गए?

```
with c as(
select sport,games from athlete_events
group by sport,games)
select sport,count(games) from c
group by sport
having count(sport)=1
```

8 . Fetch the total no of sports played in each olympic games.
8 . प्रत्येक ओलंपिक खेलों में खेले गए खेलों की कुल संख्या प्राप्त करें।

```
select COUNT(sport) as totalsports,games,Team from dbo.athlete_events group by Games,Team order by Games

select games,count(distinct(sport))as no_of_sports
from dbo.athlete_events
group by games
order by no_of_sports desc
```

9 . Fetch details of the oldest athletes to win a gold medal.
9 . स्वर्ण पदक जीतने वाले सबसे पुराने एथलीटों का विवरण प्राप्त करें।

```
select * from dbo.athlete_events where Medal = 'gold' and Age in (select max(age) from dbo.athlete_events
where Medal = 'gold' )
```

10 . Find the Ratio of male and female athletes participated in all olympic games.
10। सभी ओलंपिक खेलों में भाग लेने वाले पुरुष और महिला एथलीटों का अनुपात ज्ञात कीजिए।

```
with male_count as
(select count(sex) as male_count,games from dbo.athlete_events
where sex='M'
```

```
group by games),

female_count as
(select count(sex) as female_count,games from dbo.athlete_events
where sex='F'
group by games)

select * from male_count
join female_count on male_count.games=female_count.games


with a as (
select distinct [name],sex from dbo.athlete_events
)
select
try_convert(float,(select count(name) n1  from a where sex = 'f'
))/try_convert(float,(select count(name) n2 from a where sex = 'm'
))
```

11 . Fetch the top 5 athletes who have won the most gold medals.
11 । सबसे अधिक स्वर्ण पदक जीतने वाले शीर्ष 5 एथलीटों को देखें।

```
select name , team , count(medal)as medal from dbo.athlete_events
where medal ='gold'
group by name,team
order by medal desc
```

12 . Fetch the top 5 athletes who have won the most medals (gold/silver/bronze).
12 . सबसे अधिक पदक (स्वर्ण/रजत/कांस्य) जीतने वाले शीर्ष 5 एथलीटों को प्राप्त करें।

```
select * from dbo.athlete_events where Medal != 'NA' and Name = any(select top 5 Name from
dbo.athlete_events where Medal != 'NA'
group by name)
 order by id
```

13 . Fetch the top 5 most successful countries in olympics. Success is defined by no of medals won.
13। ओलंपिक में शीर्ष 5 सबसे सफल देशों को खोजें। सफलता को जीते गए पदकों की संख्या से परिभाषित किया जाता है।

```
select TOp (5) team from dbo.athlete_events where Medal != 'NA' and Medal =(select COUNT(distinct(medal))
from dbo.athlete_events)
```

14 . List down total gold, silver and broze medals won by each country.
14। प्रत्येक देश द्वारा जीते गए कुल स्वर्ण, रजत और कांस्य पदकों की सूची बनाएं

```
  select noc , gold,silver,bronze from(
select noc,medal,count(id) c from athlete_events
where medal != 'NA'
```

```
group by noc,medal) as m
pivot(sum(c) for medal in(gold,silver,bronze)) as pt
order by gold desc
```

SQL A to Z

```
crete database wd_4

use wd_4

create table t1 (roll_number int, name varchar(50), city varchar(100)

insert into t1 values(1,'Aman','Delhi')

select*feom t1

insert into t1 values(2,'Arun','Gurgaon'),(3,'Aryan','Noida')

select * from t1

update t1 set roll_number = 5 where name = 'Arun'

delete from t1 where name = 'Aryan'

select * from t1

alter table t1 add marks_percentage int

update t1 set marks_percentage = 90 where roll_number = 1

select name from t1 where marks_percentage < 80

select *,marks_percentage + marks_percentage*.05 as final_marks from t1

select * from t1 where name like '%a'

select marks_percentage,name from t1 where marks_percentage not in (90,86)

select * from t1 where marks_percentage between 86 and 100use wd_4

select * from t1

drop table t1

drop database wd_4

truncate table tselect t1.cl_name, t2.col_name,t3.col_name from t1 inner join t2 on t1.id = t2.id inner join t3 on
```

t2.id = t3.id

select * from t1 inner join t2 on t1.roll_number = t2.roll_number
select t1.roll_number, t2.Class from t1 inner join t2 on t1.roll_number = t2.roll_number
select t1.cl_name, t2.col_name,t3.col_name from t1 inner join t2 on t1.id = t2.id inner joint t3 on t2.id = t3.id

select * from t1 cross join t2

--constraints

create table t3 (id int unique not null, name varchar(100) not null, Salary int not null)

--primary key

create table t4 (empid int primary key identity(101,5), empname varchar(100), department varchar(100))

--foreign key

--t1 - all employees data
--t2 - account department data

create table t5 (id int, address varchar(200) constraint fk foreign key (id) references t3 (id))

create table <table_name> (col1, col2 consraint fk foreign key (child table col) references parent_table(parent table col))

select sum(list_price) from production.products
select min(list_price) from production.products
select max(list_price) from production.products
select avg(list_price) from production.products
select count(product_id) from production.products

select sin(90)
select len('fgjbfjbngnf')
select len(product_name) from production.products
select CEILING(10.2)
select floor(10.2)
select log(10)
select LOG10(10)
select EXP(4)
select sqrt(9)
select concat('ankit','kumar')
select concat_ws('_','ankit','kumar')
select trim ('   ksnbksnfkgbdg   ')
select ltrim('   ksnbksnfkgbdg   ')

```
select rtrim('    ksnbksnfkgbdg      ')
select getdate()
select sysdatetime()
select datepart(month,'2022-09-04')
select datepart(day,'2022-09-04')
select datepart(year,'2022-09-04')
select datediff(month,'2022-02-14',getdate))
select datediff(day,'2022-02-14',getdate))
select datediff(year,'2022-02-14',getdate))
select dateadd(day,16,getdate())
select dateadd(month,16,getdate())
select dateadd(year,16,getdate())
```

((question noumber (1)) create production id, name,qquantity.....

```
select production.products.product_id,production.products.product_name, production.stocks.quantity from
production.products inner join production.stocks on production.products.product_id =
production.stocks.product_id
```

```
select datename(dw,Order_Date), Order_Date from sales.orders
select datepart(dw,Order_Date), Order_Date from sales.orders
select order_Date, Shipped_Date, datediff(dd,order_Date,shipped_Date) from sales.orders
select order_Date, Shipped_Date, datediff(ww,order_Date,shipped_Date) from sales.orders
select order_Date, Shipped_Date, datediff(mm,order_Date,shipped_Date) from sales.orders
select order_Date, Shipped_Date, datediff(yy,order_Date,shipped_Date) from sales.orders
```

dssb

```
select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) = 2;

 select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) < -2;

select order_Date, Shipped_Date, datediff(dd,order_Date,shipped_Date)   from sales.orders
 where datediff(dd,order_Date,shipped_Date) > 2;

 select required_Date, Shipped_Date, datediff(dd,required_Date,shipped_Date)   from sales.orders
 where datediff(dd,required_Date,shipped_Date) < -2;

  select Datediff(dd,required_date,shipped_date)as late_delivery, sales.order_items.order_id,
  sales.order_items.product_id,production.products.product_name from
  sales.orders inner join sales.order_items on sales.orders.order_id = sales.order_items.order_id inner join
production.products on
```

production.products.product_id = sales.order_items.product_id where
Datediff(dd,required_date,shipped_date)>=2

1. Display all the managers.

select empname from tblemployee where empdesignation='manager'

2. Display all the employees except managers.

select empname from tblemployee where empdesignation not in ('manager')

3. Display the salary of employee 'Mathew'.

select empsalary from tblemployee where empname='mathew'

4. Display the manager of department 'D001'.

select empname from tblemployee where deptid= 'D001' and empdesignation='manager'

5. Display total number of employees in department 'D003'.

select empname from tblemployee where deptid='d003'

6. Display the grade of salary '33000'.

select empsalary from tblemployee where empsalary=33000

7. Display the average salary of 'D006' department.

select avg (empsalary) from tblemployee where empid='D006'

8. Display the emplyee who taking maximum salary.

select empname ,empsalary from tblemployee where empsalary =(select max (empsalary) from tblemployee)

9. Display employees who have salary between 25000 and 35000.

select empsalary from tblemployee where empsalary =(25000-35000)

10. Display all the employees with 10 percent increment in their salary.

select empsalary+empsalary *.1 as incremented_salary from tblemployee

11. Make permanent changes with 10 percent increment of every employee salary.

12. Give the 20 percent increment of employee with id 'E0005'.

13. Display employee records whos name start from 'M'.

select empname from tblemployee where empname like 'M%'

14. Display employee records whos name start from 'M' and end with 'W'.

select empname from tblemployee where empname like 'M%w'

15. Display employees whos name is 5 characters long.

select empname from tblemployee where len(empname)=5

16. Display employees whos name contains 5 or more characters.

select empname from tblemployee where len(empname) between 5 and 100

17. Display employee records whos empEmail contains _ . ex: abc_11@example.com

select * from tblemployee where empemail= 'brian@yahoo.com'

18. Display employees with palindrom name like 'nitin'.

22. Display the empId, empname and Age of employee

select empname,empid,empdob from tblemployee

23. Display employees who are elder than their manager.

24. Display deptId, deptName and no. of employees in that particular department.

select deptid, deptname from tbldepartment

25. Display the 2nd highest salary.
26. Display all employees name along with their departments.
27. Display all the employees who born in 'March' month along with their department.

```
select empdob,empname from tblemployee inner join   where DATEPART(month,(empdob))=3
```

```
=IF(RIGHT(LEFT($A$3:$A$45,3),2)="fc","facility",IF(RIGHT(LEFT($A$3:$A$45,3),2)="st","staff"))
```

Croma Campus to Everyone 2:53 PM

```
=IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$6,Lookups!$C$6,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$1,Lookups!$C$1,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$2,Lookups!$C$2,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$3,Lookups!$C$3,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$4,Lookups!$C$4,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$5,Lookups!$C$5,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$7,Lookups!$C$7,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$8,Lookups!$C$8,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$9,Lookups!$C$9,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$10,Lookups!$C$10,
IF(RIGHT(LEFT($A$3:$A$45,3),2)=Lookups!$B$11,Lookups!$C$11)))))))))))
```

Anshul to Everyone 3:01 PM

sir voice bht km ari h

Croma Campus to Everyone 3:19 PM

```
=VLOOKUP($G44,Lookups!$I:$J,2,0)
```

Croma Campus 3:46 PM

```
=SUMIFS(I$3:I$45,$E$3:$E$45,$E$50,$F$3:$F$45,$F50)
```

1--fetch the name in the format : FirstName-MiddleName-LastName of those customers with Address type as 'Main Office' .

```
Select CONCAT_WS (' - ',FirstName,MiddleName,LastName) as full_name from SalesLT.Customer where CustomerID in
(select CustomerID from SalesLT.CustomerAddress where AddressType ='main office')
```

2--fetch the product ids of those products with orderqty between 5 and 10 with any discount.

```
 select ProductId, OrderQty, UnitPriceDiscount from SalesLT.SalesOrderDetail where OrderQty between 5 and 10 and
 UnitPriceDiscount >0
```

3--fetch the details of those orders which are shipped more than 2 days late..

```
select * from salesLT.salesorderheader where datediff (day,shipdate,duedate)>=2
```

4--fetch the addresstype of the customers of Canada region

```
select addresstype from SalesLT.CustomerAddress where AddressID in (select AddressID from SalesLT.Address where CountryRegion like 'canada')
```

```
SELECT salesLT.CustomerAddress.AddressType,SalesLT.Address.CountryRegion from SalesLT.CustomerAddress
inner join SalesLT.Address on SalesLT.Address.AddressID = SalesLT.CustomerAddress.AddressID
where CountryRegion = 'canada'
5--how may products are without any parentproductid?
select
SalesLT.Product.Name,SalesLT.ProductCategory.ProductCategoryID,SalesLT.ProductCategory.ParentProductCategor
yID
from SalesLT.Product inner join SalesLT.ProductCategory on SalesLT.Product.ProductCategoryID =
SalesLT.ProductCategory.ProductCategoryID
WHERE ParentProductCategoryID != ProductID

6--fetch the name,standardcost, difference between standardcost and list price as price_difference and
salesorderid of the products with back color

select name,StandardCost ,ListPrice-StandardCost as price_diff,SalesOrderID
from SalesLT.Product inner join SalesLT.SalesOrderDetail on
SalesLT.Product.ProductID=SalesLT.SalesOrderDetail.ProductID
where Color='black'
SELECT NAME,standrdcost, listprice-standerdcost as price_diff,saleso where color='black'


select
  city,
  count (customer_id) customer_count
from
  sales.customers
group by
  city




  declare @n as int
set @n = (select count(*) from production.products)
--set @n = 10
--print @n
if @n>300
        begin
                print @n
                print 'product count is greater than 300'
        end
else
        begin
                print @n
                print'product count is less than 300'
        end
```

```
	declare @n as int
set @n = (select count(*) from production.products)
--set @n = 10
--print @n
if @n>300
	begin
		print @n
		print 'product count is greater than 300'
	end
else if @n = 300
BEGIN
	print 'hello'
end
else
	begin
		print @n
		print'product count is less than 300'
	end


ALTER PROCEDURE uspFindProducts(@min_list_price AS DECIMAL,@max_list_price as decimal,@product as
varchar(10))
AS
BEGIN
  SELECT
    product_name,
    list_price
  FROM
    production.products
  WHERE
    list_price >= @min_list_price and
		list_price<=@max_list_price
		and product_name like '%'+@product+'%'
  ORDER BY
    list_price;
END;

exec uspFindProducts 500,1000,'Trek'



create procedure product_quantity
as
begin
	select product_name, quantity from production.products
	inner join production.stocks on production.products.product_id=production.stocks.product_id
```

```
        end

exec product_quantity


create procedure propro
as
begin
select
        product_name
        from
        production.products
        order by
        product_name
end

exec propro



declare @s as int = 0
while (@s<=5)
begin
        print @s
        set @s = @s+1
end


declare @s as int = 0
while (@s<=10)
begin
        set @s = @s+1
        if @s = 5
                continue
        print @s
end

create view let_delivery
as
select Datediff(dd,required_date,shipped_date)as
  late_delivery, sales.order_items.order_id, sales.order_items.product_id,production.products.product_name
  from sales.orders inner join sales.order_items
  on sales.orders.order_id = sales.order_items.order_id inner join production.products
  on production.products.product_id = sales.order_items.product_id where
Datediff(dd,required_date,shipped_date)>=2
```

```
   CREATE TABLE sales.rank_demo (
            v VARCHAR(10)
);
INSERT INTO sales.rank_demo(v)
VALUES('A'),('B'),('B'),('C'),('C'),('D'),('E');


SELECT
            v,
            RANK () OVER (
                        ORDER BY v
            ) rank_no
FROM
            sales.rank_demo;

            CREATE VIEW sales.vw_staff_sales(
   staff_id,
   year,
   net_sales
) AS
SELECT
   staff_id,
   YEAR(order_date),
   ROUND(SUM(quantity*list_price*(1-discount)),0)
FROM
   sales.orders o
INNER JOIN sales.order_items i on i.order_id = o.order_id
WHERE
   staff_id IS NOT NULL
GROUP BY
   staff_id,
   YEAR(order_date);

            SELECT
   CONCAT_WS(' ',first_name,last_name) full_name,
   net_sales,
   PERCENT_RANK() OVER (
      ORDER BY net_sales DESC
   ) percent_rank
FROM
   sales.vw_staff_sales t
INNER JOIN sales.staffs m on m.staff_id = t.staff_id
WHERE
   YEAR = 2016;

            SELECT
            v,
            NTILE (4) OVER (
```

```
                ORDER BY v
        ) buckets
FROM
        sales.ntile_demo;

CREATE VIEW sales.vw_netsales_brands
AS
        SELECT
                c.brand_name,
                MONTH(o.order_date) month,
                YEAR(o.order_date) year,
                CONVERT(DEC(10, 0), SUM((i.list_price * i.quantity) * (1 - i.discount))) AS net_sales
        FROM sales.orders AS o
                INNER JOIN sales.order_items AS i ON i.order_id = o.order_id
                INNER JOIN production.products AS p ON p.product_id = i.product_id
                INNER JOIN production.brands AS c ON c.brand_id = p.brand_id
        GROUP BY c.brand_name,
                        MONTH(o.order_date),
                        YEAR(o.order_date);

create procedure min1(@year as int)
as begin SELECT
   category_name,
   year,
   qty,
   FIRST_VALUE(category_name) OVER(
      ORDER BY qty
   ) lowest_sales_volume
FROM
   sales.vw_category_sales_volume
WHERE
   year = @year
end

exec min1 2018
```

Assignmnent (2)

---

1 . How many olympics games have been held?
1। कितने ओलंपिक खेल आयोजित किए गए हैं?

```
select count(distinct(games)) from dbo.athlete_events
```

2 . a) List down all Olympics games held so far.
2 . क) अब तक आयोजित सभी ओलंपिक खेलों की सूची बनाएं।

```
select distinct year , season , city from dbo.athlete_events order by year
```

b) list down all olympics in year-season-city format. (ex: 1960-summer-roma)
बी) साल-मौसम-शहर प्रारूप में सभी ओलंपिक को सूचीबद्ध करें। (उदा: 1960-ग्रीष्म-रोमा)

```
select distinct concat_ws('-',year,season, city) all_olympic from athlete_events
```

3 . Mention the total no of nations who participated in each olympics game?
3। प्रत्येक ओलंपिक खेल में भाग लेने वाले देशों की कुल संख्या का उल्लेख करें?

```
select games,count(distinct(noc)) as total_countries from dbo.athlete_events group by games order by total_countries
```

4 . Which year saw the highest and lowest no of countries participating in olympics?
4। किस वर्ष ओलंपिक में भाग लेने वाले देशों की उच्चतम और निम्नतम संख्या देखी गई?

```
 select COUNT(distinct(team)) as nation,Year from dbo.athlete_events group by Year,Season having year = MAX(year) order by nation desc
 select COUNT(distinct(noc)) as nation,Year from dbo.athlete_events group by Year having year = Min(year) order by nation
```

5 . Which nation has participated in all of the olympic games?
5। किस देश ने सभी ओलंपिक खेलों में भाग लिया है?

```
with c as
(select noc,games
from athlete_events group by noc,games
)
select noc,count(games) from c
group by noc
having count(games) = (select count(distinct games) from athlete_events)

with nations as (select NOC,year,season from athlete_events group by Year,NOC,season)
select NOC from nations group by NOC having COUNT(NOC)=51
```

6 . Identify the sport which was played in all summer olympics.
6 . उस खेल की पहचान करें जो सभी ग्रीष्मकालीन ओलंपिक में खेला गया था।

```
select sport, Season from dbo.athlete_events where Season like 'summer'
group by Sport,Season order by Sport
```

7 . Which Sports were just played only once in the olympics?
7। कौन से खेल ओलंपिक में केवल एक बार खेले गए?

```
with c as(
select sport,games from athlete_events
group by sport,games)
select sport,count(games) from c
```

```
group by sport
having count(sport)=1
```

8 . Fetch the total no of sports played in each olympic games.
8 . प्रत्येक ओलंपिक खेलों में खेले गए खेलों की कुल संख्या प्राप्त करें।

```
select COUNT(sport) as totalsports,games,Team from dbo.athlete_events group by Games,Team order by Games

select games,count(distinct(sport))as no_of_sports
from dbo.athlete_events
group by games
order by no_of_sports desc
```

9 . Fetch details of the oldest athletes to win a gold medal.
9 . स्वर्ण पदक जीतने वाले सबसे पुराने एथलीटों का विवरण प्राप्त करें।

```
select * from dbo.athlete_events where Medal = 'gold' and Age in (select max(age) from dbo.athlete_events
where Medal = 'gold' )
```

10 . Find the Ratio of male and female athletes participated in all olympic games.
10। सभी ओलंपिक खेलों में भाग लेने वाले पुरुष और महिला एथलीटों का अनुपात ज्ञात कीजिए।

```
with male_count as
(select count(sex) as male_count,games from dbo.athlete_events
where sex='M'
group by games),

female_count as
(select count(sex) as female_count,games from dbo.athlete_events
where sex='F'
group by games)

select * from male_count
join female_count on male_count.games=female_count.games


with a as (
select distinct [name],sex from dbo.athlete_events
)
select
try_convert(float,(select count(name) n1  from a where sex = 'f'
))/try_convert(float,(select count(name) n2 from a where sex = 'm'
))
```

11 . Fetch the top 5 athletes who have won the most gold medals.
11 । सबसे अधिक स्वर्ण पदक जीतने वाले शीर्ष 5 एथलीटों को देखें।

```
select name , team , count(medal)as medal from dbo.athlete_events
where medal ='gold'
group by name,team
order by medal desc
```

12 . Fetch the top 5 athletes who have won the most medals (gold/silver/bronze).
12 . सबसे अधिक पदक (स्वर्ण/रजत/कांस्य) जीतने वाले शीर्ष 5 एथलीटों को प्राप्त करें।

```
select * from dbo.athlete_events where Medal != 'NA' and Name = any(select top 5 Name from
dbo.athlete_events where Medal != 'NA'
group by name)
 order by id
```

13 . Fetch the top 5 most successful countries in olympics. Success is defined by no of medals won.
13। ओलंपिक में शीर्ष 5 सबसे सफल देशों को खोजें। सफलता को जीते गए पदकों की संख्या से परिभाषित किया जाता है।

```
select TOp (5) team from dbo.athlete_events where Medal != 'NA' and Medal =(select COUNT(distinct(medal))
from dbo.athlete_events)
```

14 . List down total gold, silver and broze medals won by each country.
14। प्रत्येक देश द्वारा जीते गए कुल स्वर्ण, रजत और कांस्य पदकों की सूची बनाएं

```
  select noc , gold,silver,bronze from(
select noc,medal,count(id) c from athlete_events
where medal != 'NA'
group by noc,medal) as m
pivot(sum(c) for medal in(gold,silver,bronze)) as pt
order by gold desc
```

##$$$$$$$$$$$$$$$$$$$@$
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@&&&&&&******&&&&&&&&******
******

XAMPP

###@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@&&&&&&&&&&&&&&&&&&&&&&&$$$$$$$$$$$$
$$$$$$$*************

Setting environment for using XAMPP for Windows.
Welcome@DESKTOP-6SKBR3G c:\xampp
mysql -u root

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.28-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| phpmyadmin         |
| test               |
+--------------------+
5 rows in set (0.118 sec)

MariaDB [(none)]> create database wd915;
Query OK, 1 row affected (0.006 sec)
MariaDB [(none)]> use wd915;
Database changed
MariaDB [wd915]> create table t1(id int, name varchar(50), salary int);
Query OK, 0 rows affected (0.028 sec)

MariaDB [wd915]> desc t1;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| id     | int(11)     | YES  |     | NULL    |       |
| name   | varchar(50) | YES  |     | NULL    |       |
| salary | int(11)     | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
3 rows in set (0.035 sec)

MariaDB [wd915]> insert into t1 values(1,"Ankit",230000);
Query OK, 1 row affected (0.106 sec)
MariaDB [wd915]> select * from t1;
+------+-------+--------+
| id   | name  | salary |
+------+-------+--------+
|    1 | Ankit | 230000 |
+------+-------+--------+
1 row in set (0.002 sec)

MariaDB [wd915]> insert into t1 values(2,"Aman",34000),(3,"Abhishek",45000);
```

```
Query OK, 2 rows affected (0.006 sec)
Records: 2  Duplicates: 0  Warnings: 0

MariaDB [wd915]> select * from t1;
+------+----------+--------+
| id   | name     | salary |
+------+----------+--------+
|    1 | Ankit    | 230000 |
|    2 | Aman     |  34000 |
|    3 | Abhishek |  45000 |
+------+----------+--------+
3 rows in set (0.002 sec)

MariaDB [wd915]> select salary from t1 where nam = "Ankit";
```