**Creating Tables with SQL**

1. Gallery (GalleryID, Name)

CREATE TABLE Gallery (

GalleryID varchar(10) NOT NULL,

Name varchar(20),

CONSTRAINT pk\_Gallery Primary Key (GalleryID)

)

INSERT INTO Gallery (

GalleryID, Name)

VALUES (487175, "Tiles"

)

INSERT INTO Gallery (

GalleryID, Name)

VALUES (714276, "Miles"

)

Table

Description automatically generated

1. Gallery\_Address (GalleryID(fk), Address, Capacity, NumOfFloors, NumOfExhibitions)

CREATE TABLE Gallery\_Address (

GalleryID varchar(10) NOT NULL,

Address varchar (30) NOT NULL,

Capacity NUMBER,

NumOfFloors NUMBER,

NumOfExhibitions NUMBER,

CONSTRAINT pk\_Gallery\_Address Primary Key (Address),

CONSTRAINT fk\_Gallyer\_Address FOREIGN KEY (GalleryID) REFERENCES Gallery (GalleryID)

)

INSERT INTO Gallery\_Address

VALUES (714276,"3412 24 AVE NY","55", "2", "110"

)

INSERT INTO Gallery\_Address

VALUES (487175,"1457 39 AVE NY", "30", "1", "60"

)

Table

Description automatically generated

1. Exhibition (ExhibitionID, Name, Year, Genre, NumOfDisplay, GalleryID(fk))

CREATE TABLE Exhibition (

ExhibitionID varchar (30) NOT NULL,

Name varchar(20),

Year Number,

Genre varchar(15),

NumOfDisplay NUMBER,

GalleryID varchar(10) NOT NULL,

CONSTRAINT pk\_Exhibition PRIMARY KEY (ExhibitionID),

CONSTRAINT fk\_Exhibition FOREIGN KEY (GalleryID) REFERENCES Gallery (GalleryID)

)

INSERT INTO Exhibition

(ExhibitionID, Name, Year, Genre, NumOfDisplay, GalleryID)

VALUES ( 4572, "Stock", "2019", "Distant", 55, 487175)

INSERT INTO Exhibition

(ExhibitionID, Name, Year, Genre, NumOfDisplay, GalleryID)

VALUES ( 4573, "Mark", "2019", "Area", 30, 714276)

Table

Description automatically generated

1. Staff (StaffID, Wage, Workhours, PhoneNumber(fk))

CREATE TABLE Staff (

StaffID varchar (10) NOT NULL,

Wage NUMBER,

Workhours NUMBER,

CONSTRAINT pk\_Staff PRIMARY KEY (StaffID)

)

ALTER TABLE Staff

ADD COLUMN PhoneNumber Number NOT NULL

ALTER TABLE Staff

ADD CONSTRAINT fk\_Staff Foreign Key (PhoneNumber) REFERENCES Staff\_Phone (PhoneNumber)

INSERT INTO Staff

(StaffID, Wage, Workhours, PhoneNumber)

VALUES ( 102, 16, 20, 144754344)

INSERT INTO Staff

(StaffID, Wage, Workhours, PhoneNumber)

VALUES ( 101, 16, 20, 457440789)

Table

Description automatically generated

1. Staff\_Phone(PhoneNumber, Name, Address)

CREATE TABLE Staff\_Phone (

PhoneNumber Number NOT NULL,

Name varchar (20),

Address varchar(30),

Constraint pk\_Staff\_Phone PRIMARY KEY (PhoneNumber)

)

INSERT INTO Staff\_Phone

(PhoneNumber, Name, Address)

VALUES ( 144754344, "Jack Sant", "414 Blk Ave")

INSERT INTO Staff\_Phone

(PhoneNumber, Name, Address)

VALUES ( 457440789, "Mike Trout", "414 Amsterdam Ave")

Table, calendar

Description automatically generated

1. ArtDisplay(DisplayID, Name)

CREATE TABLE ArtDisplay (

DisplayID VARCHAR(10) NOT NULL,

Name Varchar(20),

Constraint pk\_ArtDisplay PRIMARY KEY (DisplayID)

)

INSERT INTO ArtDisplay

(DisplayID, Name)

VALUES ( 2458, "SHAM")

INSERT INTO ArtDisplay

(DisplayID, Name)

VALUES ( 2459, "MASH")

Table

Description automatically generated

1. Artist (ArtistID, Age, Gender, PhoneNumber, Email)

CREATE TABLE Artist (

ArtistID VARCHAR (10) NOT NULL,

Age Number,

Gender VARCHAR (10),

PhoneNumber Number,

Email VARCHAR (30),

Constraint pk\_Artist PRIMARY KEY (ArtistID)

)

ALTER TABLE Artist

ADD Constraint fk\_Artist FOREIGN KEY (Email) REFERENCES Artist\_Email (Email)

INSERT INTO Artist

(ArtistID, Age, Gender, PhoneNumber, Email)

VALUES ( 84575, 45, "Male", 7189472478, "mikedek@gmail.com" )

INSERT INTO Artist

(ArtistID, Age, Gender, PhoneNumber, Email)

VALUES ( 84576, 40, "Female", 7189477223, "Michellef@gmail.com" )

Table

Description automatically generated

1. Artist\_Email (Email, Name, Address)

CREATE TABLE Artist\_Email (

Email VARCHAR (30) NOT NULL,

Name VARCHAR (20),

Address VARCHAR (30),

Constraint pk\_Artist\_Email Primary Key (Email)

)

INSERT INTO Artist\_Email

(Email, Name, Address)

VALUES ( "Michellef@gmail.com", "Michelle Fi", "4578 8th Street NY")

INSERT INTO Artist\_Email

(Email, Name, Address)

VALUES ( "mikedek@gmail.com" , "Mike Deker", "424 7th AVE NY")

A picture containing graphical user interface

Description automatically generated

1. ArtWork(ArtworkID,Title, YearMade, Genre, Price, ArtistID(fk))

CREATE TABLE ArtWork (

ArtWorkID VARCHAR (10) NOT NULL,

Title VARCHAR (20),

YearMade Number,

Genre VARCHAR (10),

Price Number,

ArtistID VARCHAR (10) NOT NULL,

Constraint pk\_ArtWork Primary Key (ArtWorkID),

CONSTRAINT fk\_Artwork FOREIGN KEY (ArtistID) REFERENCES Artist (ArtistID)

)

INSERT INTO ArtWork

(ArtworkID,Title, YearMade, Genre, Price, ArtistID)

VALUES ( 4572475, "Tall", 2014, "Distant", 255, 84576)

INSERT INTO ArtWork

(ArtworkID,Title, YearMade, Genre, Price, ArtistID)

VALUES ( 4572474, "Mall", 2014, "Distant", 250, 84575)

Table

Description automatically generated

1. ArtWork\_Artist(ArtistID(fk), ArtistName)

CREATE TABLE ArtWork\_Artist (

ArtistID VARCHAR (10) NOT NULL,

ArtistName VARCHAR (20),

Constraint pk\_ArtWork\_Artist Primary Key (ArtistID),

CONSTRAINT fk\_ArtWork\_Artist FOREIGN KEY (ArtistID) REFERENCES Artist (ArtistID)

)

INSERT INTO ArtWork\_Artist

(ArtistID, ArtistName)

VALUES (84575, "Selena Gomez")

INSERT INTO ArtWork\_Artist

(ArtistID, ArtistName)

VALUES (84576, "Jay Z")

Table

Description automatically generated

1. ArtworkCopy (ArtworkCopyNum, ArtWorkID(fk))

CREATE TABLE ArtworkCopy (

ArtworkCopyNum VARCHAR (10) NOT NULL,

ArtWorkID VARCHAR (10) NOT NULL,

CONSTRAINT pk\_ArtworkCopy PRIMARY KEY (ArtworkCopyNum),

CONSTRAINT fk\_ArtworkCopy FOREIGN KEY (ArtWorkID) REFERENCES ArtWork (ArtWorkID)

)

INSERT INTO ArtworkCopy

(ArtworkCopyNum, ArtWorkID)

VALUES ( 47572,4572475)

INSERT INTO ArtworkCopy

(ArtworkCopyNum, ArtWorkID)

VALUES ( 47573,4572474)

Table

Description automatically generated

1. ArtDisplayName(Name, NumOfArtworks, ArtworkCopyNum(fk))

CREATE TABLE ArtDisplayName (

Name VARCHAR (20) NOT NULL,

NumOfArtworks NUMBER,

ArtworkCopyNum VARCHAR (10) NOT NULL,

CONSTRAINT pk\_ArtDisplayName PRIMARY KEY (Name),

CONSTRAINT fk\_ArtDisplayName FOREIGN KEY (ArtworkCopyNum) REFERENCES ArtworkCopy (ArtworkCopyNum)

)

INSERT INTO ArtDisplayName(Name, NumOfArtworks, ArtworkCopyNum)

VALUES ("MASH", 4,47572)

INSERT INTO ArtDisplayName(Name, NumOfArtworks, ArtworkCopyNum)

VALUES ("TAN", 3,47573)

Table

Description automatically generated

1. Customer (CustomerID, Membership, Age, Gender,)

CREATE TABLE Customer (

CustomerID VARCHAR (10) NOT NULL,

Membership VARCHAR (3),

Age Number,

Gender VARCHAR (10),

Constraint pk\_Cusomter PRIMARY KEY (CustomerID)

)

INSERT INTO Customer (CustomerID, Membership, Age, Gender, EMAIL)

VALUES (98247, "Non-Member", 24, "Female", “[Jamesemacvoy@gmail.com](mailto:Jamesemacvoy@gmail.com)”)

INSERT INTO Customer (CustomerID, Membership, Age, Gender, EMAIL)

VALUES (98246, "Member", 29, "Male", “IanMckalin@gmail.com”)

ALTER TABLE Customer

ADD EMAIL VARCHAR(30)

ALTER TABLE Customer

ADD CONSTRAINT fk\_Customer FOREIGN KEY (EMAIL) References Cusomter\_info (Email)

Table

Description automatically generated

1. Customer\_info (Email, Name, Address)

CREATE TABLE Cusomter\_info (

Email VARCHAR (30) NOT NULL,

Name VARCHAR (20),

Address VARCHAR (30),

Constraint pk\_Cusomter\_info Primary Key (Email)

)

INSERT INTO Cusomter\_info (Email, Name, Address)

VALUES ("Jamesemacvoy@gmail.com", "James Macvoy", "342 Harris AVE NY")

INSERT INTO Cusomter\_info (Email, Name, Address)

VALUES ("IanMckalin@gmail.com", "Iac Mckalin", "341 Harris AVE NY")

Table

Description automatically generated

1. Orders (OrderID, TotalItems, TotalPrice, PurchaseMethod, ArtworkCopyNum(fk), CustomerID(fk))

CREATE TABLE Orders (

OrderID VARCHAR (10) NOT NULL,

TotalItems Number,

TotalPrice Number,

PurchaseMethod VARCHAR (20),

ArtworkCopyNum VARCHAR (10) NOT NULL,

CustomerID VARCHAR (10) NOT NULL,

CONSTRAINT pk\_Order PRIMARY KEY (OrderID),

CONSTRAINT fk1\_Order FOREIGN KEY (ArtworkCopyNum) REFERENCES ArtworkCopy (ArtworkCopyNum),

CONSTRAINT fk2\_Order FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID)

)

INSERT INTO Orders

VALUES ("IM246", 2, 510, "Credit", "47572", "98246")

INSERT INTO Orders

VALUES ("IM247", 2, 500, "Credit", 47573, 98247)

Table

Description automatically generated

1. Ticket (TicketID, Price, Entry\_Day, Entry\_Time)

CREATE TABLE Ticket (

TicketID VARCHAR(10) NOT NULL,

Price NUMBER,

Entry\_Day Date,

Entry\_Time DATETIME,

CONSTRAINT pk\_Ticket PRIMARY KEY (TicketID)

)

INSERT INTO Ticket

VALUES ("TIanM46", 8, #2020-05-18#, #15:30#)

INSERT INTO Ticket

VALUES ("TJamM47", 10, #2020-05-18#, #15:30#)

Table

Description automatically generated

1. CusomterTicket(CustomerID(fk), TicketID(fk))

CREATE TABLE CusomterTicket (

CustomerID VARCHAR (10) NOT NULL,

TicketID VARCHAR(10) NOT NULL,

CONSTRAINT pk\_CustomerTicket PRIMARY KEY (CustomerID),

CONSTRAINT fk1\_CusomterTicket FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID),

CONSTRAINT fk2\_CusomterTicket FOREIGN KEY (TicketID) REFERENCES Ticket (TicketID)

)

INSERT INTO CusomterTicket

VALUES ("98246", "TIanM46")

INSERT INTO CusomterTicket

VALUES ("98247", "TJamM47")

Table

Description automatically generated

1. Tour (TourID, ExhibitionVisited)

CREATE TABLE Tour (

TourID VARCHAR (10) NOT NULL,

ExhibitionVisited NUMBER,

CONSTRAINT pk\_TourID PRIMARY KEY (TourID)

)

INSERT INTO Tour

VALUES ("Tour18", 4)

INSERT INTO Tour

VALUES ("Tour19", 2)

A picture containing text, cabinet

Description automatically generated

1. Tour\_Visited (ExhibitionVisited, ExhibitionID)

CREATE TABLE Tour\_Visited (

ExhibitionVisited NUMBER NOT NULL,

ExhibitionID varchar (30) NOT NULL,

CONSTRAINT fk\_Tour\_Visited FOREIGN KEY (ExhibitionID) REFERENCES Exhibition (ExhibitionID)

)

INSERT INTO Tour\_Visited

VALUES (4, 4572)

INSERT INTO Tour\_Visited

VALUES (2, 4573)

Table

Description automatically generated

1. Appointment\_Size (TourID, GroupSize)

CREATE TABLE Appointment\_Size (

TourID VARCHAR (10) NOT NULL,

GroupSize NUMBER,

CONSTRAINT pk\_Appointment\_Size PRIMARY KEY (TourID),

CONSTRAINT fk\_Appointment\_Size FOREIGN KEY (TourID) REFERENCES Tour (TourID)

)

INSERT INTO Appointment\_Size

VALUES ("Tour18", 10)

INSERT INTO Appointment\_Size

VALUES ("Tour19", 20)

Table

Description automatically generated

1. Tour\_Appointment (AppointmentID, AppointmentDate, AppointmentTime, TourID(fk))

CREATE TABLE Tour\_Appointment (

AppointmentID VARCHAR (10) NOT NULL,

AppointmentDate DATE,

AppointmentTime DATETIME,

TourID VARCHAR (10) NOT NULL,

CONSTRAINT pk\_Tour\_Appointment PRIMARY KEY (AppointmentID),

CONSTRAINT fk\_Tour\_Appointment FOREIGN KEY (TourID) REFERENCES Tour (TourID)

)

INSERT INTO Tour\_Appointment

VALUES ("TA20”, #2020-05-18#, #16:00#, “Tour18”)

INSERT INTO Tour\_Appointment

VALUES ("TA21", #2020-05-18#, #17:00#, "Tour19")

Table

Description automatically generated

1. Exhibition\_Art (ExhibitionID (fk), DisplayID(fk))

CREATE TABLE Exhibition\_Art (

ExhibitionID VARCHAR (30) NOT NULL,

DisplayID VARCHAR(10) NOT NULL,

CONSTRAINT pk\_Exhibition\_Art PRIMARY KEY (ExhibitionID, DisplayID),

CONSTRAINT fk1\_Exhibition\_Art FOREIGN KEY (ExhibitionID) REFERENCES Exhibition (ExhibitionID),

CONSTRAINT fk2\_Exhibition\_Art FOREIGN KEY (DisplayID) REFERENCES ArtDisplay (DisplayID)

)

INSERT INTO Exhibition\_Art

VALUES ("4572", "2458")

INSERT INTO Exhibition\_Art

VALUES ("4573", "2459")

Table

Description automatically generated

1. Exhibition\_Staff (StaffID(fk), ExhibitionID (fk))

CREATE TABLE Exhibition\_Staff (

StaffID varchar (10) NOT NULL,

ExhibitionID VARCHAR (30) NOT NULL,

CONSTRAINT pk\_Exhibition\_Staff PRIMARY KEY (StaffID, ExhibitionID),

CONSTRAINT fk1\_Exhibition\_Staff FOREIGN KEY (StaffID) REFERENCES Staff (StaffID),

CONSTRAINT fk2\_Exhibition\_Staff FOREIGN KEY (ExhibitionID) REFERENCES Exhibition (ExhibitionID)

)

INSERT INTO Exhibition\_Staff

VALUES ("101", "4573")

INSERT INTO Exhibition\_Staff

VALUES ("102", "4572")

Table

Description automatically generated

1. Organizer (StaffID(fk))

CREATE TABLE Organizer (

StaffID varchar (10) NOT NULL,

CONSTRAINT pk\_Organizer PRIMARY KEY (StaffID),

CONSTRAINT fk\_Organizer FOREIGN KEY (StaffID) REFERENCES Staff (StaffID)

)

INSERT INTO Organizer

VALUES ("101")

INSERT INTO Organizer

VALUES ("102")

Table

Description automatically generated

1. Exhibition\_Ticket (TicketID(fk), ExhibitionID(fk))

CREATE TABLE Exhibition\_Ticket (

TicketID VARCHAR(10) NOT NULL,

ExhibitionID VARCHAR (30) NOT NULL,

CONSTRAINT pk\_Exhibition\_Ticket PRIMARY KEY (TicketID,ExhibitionID),

CONSTRAINT fk1\_Exhibition\_Ticket FOREIGN KEY (TicketID) REFERENCES Ticket (TicketID),

CONSTRAINT fk2\_Exhibition\_Ticket FOREIGN KEY (ExhibitionID) REFERENCES Exhibition (ExhibitionID)

)

INSERT INTO Exhibition\_Ticket

VALUES ("TIanM46", "4573")

INSERT INTO Exhibition\_Ticket

VALUES ("TJamM47", "4572")

Table

Description automatically generated

1. TourGuide\_Tour (StaffID(fk), TourID(fk))

CREATE TABLE TourGuide\_Tour (

StaffID varchar (10) NOT NULL,

TourID VARCHAR (10) NOT NULL,

CONSTRAINT pk\_TourGuide\_Tour PRIMARY KEY (StaffID,TourID),

CONSTRAINT fk1\_TourGuide\_Tour FOREIGN KEY (StaffID) REFERENCES Staff (StaffID),

CONSTRAINT fk2\_TourGuide\_Tour FOREIGN KEY (TourID) REFERENCES Tour (TourID)

)

INSERT INTO TourGuide\_Tour

VALUES ("102", "Tour19")

INSERT INTO TourGuide\_Tour

VALUES ("101", "Tour18")

Table

Description automatically generated

1. Customer\_TourAppointment (CustomerID(fk), AppointmentID(fk))

CREATE TABLE Customer\_TourAppointment (

CustomerID VARCHAR (10) NOT NULL,

AppointmentID VARCHAR (10) NOT NULL,

CONSTRAINT pk\_Customer\_TourAppointment PRIMARY KEY (CustomerID, AppointmentID),

CONSTRAINT fk1\_Cusomter\_TourAppointment FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID),

CONSTRAINT fk2\_Cusomter\_TourAppointment FOREIGN KEY (AppointmentID) REFERENCES Tour\_Appointment (AppointmentID)

)

INSERT INTO Customer\_TourAppointment

VALUES ("98246", "TA20")

INSERT INTO Customer\_TourAppointment

VALUES ("98247", "TA21")

Table

Description automatically generated with medium confidence

1. TourGuide (StaffID(fk), NumOfTours)

CREATE TABLE TourGuide (

StaffID varchar (10) NOT NULL,

NumOfTours NUMBER,

CONSTRAINT pk\_TourGuide PRIMARY KEY (StaffID),

CONSTRAINT fk\_TourGuide FOREIGN KEY (StaffID) REFERENCES Staff (StaffID)

)

INSERT INTO TourGuide

VALUES ("103", 5)

INSERT INTO TourGuide

VALUES ("105", 10)

A screenshot of a computer

Description automatically generated with low confidence

**BUSINESS SCENARIOS**

**Question 1: The City wants to know if more members or nonmembers are coming to the art gallery events. This will help them focus their marketing efforts on attracting more customers for the group with the least customers. They also want to know the average age of the two groups based on gender.**

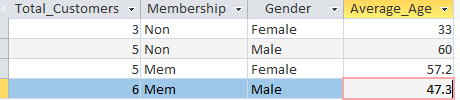
SELECT COUNT(Membership) AS Total\_Customers, Membership, Gender,

ROUND(AVG(Age), 1) AS Average\_Age

FROM Customer

GROUP BY Membership, Gender

ORDER BY COUNT(Membership)



**Question 2a: The city wants to extend its hours for the most popular day of the art gallery. They want to see which day of the week the most tickets were redeemed.**

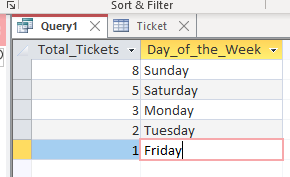
SELECT COUNT(Entry\_Day) AS Total\_Tickets,

FORMAT(Entry\_Day, "DDDD") AS Day\_of\_the\_Week

FROM Ticket

GROUP BY FORMAT(Entry\_Day, "DDDD")

ORDER BY COUNT(Entry\_Day) DESC



**Question 2b: They also want to see how this is connected to the entry time for that specific day.**

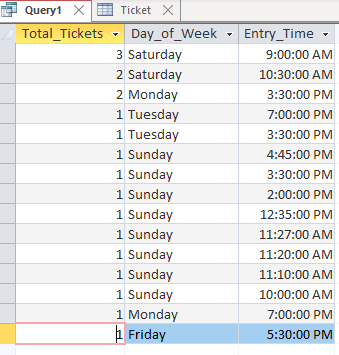
SELECT COUNT(Entry\_Day) AS Total\_Tickets,

FORMAT(Entry\_Day, "DDDD") AS Day\_of\_Week, Entry\_Time

FROM Ticket

GROUP BY FORMAT(Entry\_Day, "DDDD"), Entry\_Time

ORDER BY COUNT(Entry\_Day) DESC



**Question 3: The city has asked us to find the total of artwork copies sold by genre and their total price. This will help artists create more artworks within the popular genre to increase their sales.**

SELECT SUM(o.TotalItems) AS Total\_Art\_Copies\_Sold, a.Genre,

SUM(o.TotalPrice) AS Total\_Price\_per\_Genre

FROM (( Orders o INNER JOIN ArtworkCopy ac

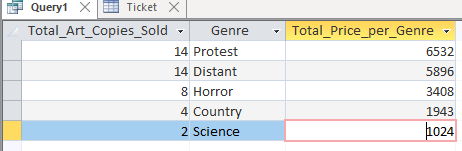
ON o.ArtworkCopyNum = ac.ArtworkCopyNum)

INNER JOIN ArtWork a

ON ac.ArtWorkID= a.ArtWorkID )

GROUP BY a.Genre

ORDER BY SUM(o.TotalItems) DESC



**Question 4: The Artists have asked us to create a command which will allow them to see how well other artists are doing in sales. They want to see where they stand and hope to use the information to improve their overall popularity and sales.**

SELECT o.TotalItems, a.ArtWorkID, a.Genre, an.ArtistName, o.TotalPrice

FROM ((( Orders o INNER JOIN ArtworkCopy ac

ON o.ArtworkCopyNum = ac.ArtworkCopyNum)

INNER JOIN ArtWork a

ON ac.ArtWorkID= a.ArtWorkID)

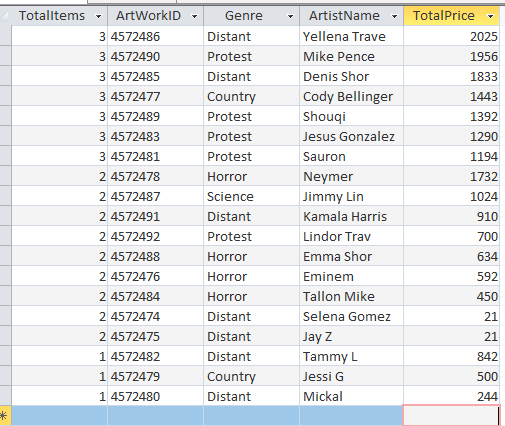
INNER JOIN Artist at

ON a.ArtistID = at.ArtistID)

INNER JOIN ArtWork\_Artist an

ON an.ArtistID= a.ArtistID

ORDER BY o.TotalItems DESC, o.TotalPrice DESC



**Question 5: The art gallery allows tour guides a maximum of five tours. Their total wages should not exceed $450 as well, since this will exceed the tour guide budget. Please find the tour guide staff who should not give tours next week based on the restrictions.**

SELECT sp.Name, s.StaffID, t.NumOfTours, (Wage \* Workhours) AS Total\_Wage

FROM (Staff\_Phone sp INNER JOIN Staff s

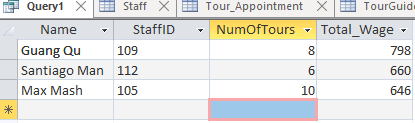
ON sp.PhoneNumber = s.PhoneNumber)

INNER JOIN TourGuide t

ON t.StaffID = s.StaffID

WHERE (Wage \* Workhours) >450 AND t.NumOfTours > 5

ORDER BY (Wage \* Workhours) DESC

****

**Question 6: The city has asked us to find galleries which ran in 2020. They want to rerun these galleries because the exhibitions were cut short in that year due to COVID-19. We need to find the galleries within the exhibitions that ran in 2020 so the city can include them for next year’s show.**

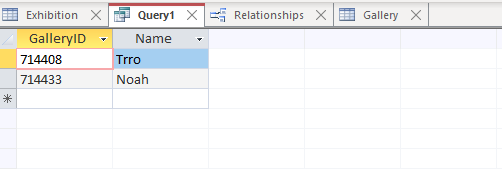
SELECT \*

FROM Gallery

WHERE GalleryID IN (SELECT GalleryID

FROM Exhibition

WHERE Year = (2020))

****