**Passang Tshering \_ 34120824**

Text, letter

Description automatically generated

A picture containing text, whiteboard

Description automatically generated

**Question 2.**

SELECT LastName, FirstName, title, medium description

FROM dtoohey.artist A, dtoohey.work W

WHERE A.artistid = W.artistid and w.copy > '3'

SELECT lastName, firstName, title, medium, description

FROM dtoohey.artist A join dtoohey.work W

on A.artistid = W.artistid

WHERE description = 'Surrealist style'

SELECT a.firstname, a.lastname, w.\*

FROM (dtoohey.artist A join dtoohey.work W

on A.artistid = W.artistiD) join dtoohey.trans T

on T.workid = W.workid

WHERE T.datesold IS NULL

SELECT c.\* ,t.\*,w.Title as 'Work of Art'

FROM (dtoohey.artist A join dtoohey.work W

on t.workID = w.workID) join Customer c

on t.CustomerID=c.CustomerID

WHERE dateSold IS NOT NULL

AND SalesPrice IS NOT NULL

SELECT firstname, lastname "artistname", DateDeceased-DateOfBirth as Age

FROM dtoohey.artist

WHERE dateOfBirth > 1800

SELECT extract (year from dateacquired) as Year, sum(acquisitionPrice)

FROM dtoohey.trans group by extract (year from dateacquired);

SELECT (sum(salesprice) - sum(acquisitionPrice)) as TotalProfit

FROM dtoohey.trans

WHERE datesold is not null;

SELECT A.firstname, A.lastname as ArtistName, count(T.datesold)as TotalSold

FROM (dtoohey.artist A join dtoohey.work W on A.artistid = w.artistid) join dtoohey.trans T on T.workid = W.Workid

WHERE W.workid in

(select workid FROM

(select workid, count(datesold) as maximum

FROM dtoohey.trans group by workid

order by maximum DESC)

WHERE rownum = 1)

group by A.firstname, A.lastname

SELECT C.Firstname, C.LastName as CustomerName

FROM dtoohey.customer C join dtoohey.Customer\_artist\_int CAI

on c.customerid = CAI.customerid

WHERE CAI.customerid in

(Select Customerid

FROM dtoohey.Customer\_artist\_int

group by customerid having count(\*) =

(SELECT count (\*)

FROM dtoohey.artist))

group by C.firstname, C.lastname

**Question 3.**

INSERT INTO Customer(CustomerFirstName, CustomerLastName street, City, State, areacode,Country, email)

VALUES ('Join', 'Smith', '47 Moodle Street', 'Highwater', 'WA', '6709', 'Autralia', 'jsmith@someisp.com');

INSERT INTO Artist(ArtistFirstName, ArtistLastName, nationality, DateOfBirth)

VALUES ('Gallic', 'Symbol', 'French', '1972');

INSERT INTO Work(Title, copy, medium, decription)

VALUES ('Gorillas in the Mist', 'unique', 'WaterColour on the paper', '45 \* 35cm signed by theArtist');

INSERT INTO Transaction(DateSold, SalePrice)

VALUES ('27/07/2020', '4600');

Create table MATCH

(MatchID Number(38),

MatchDate varchar(60) not null

MatchResults varchar(50) not null

HomeTeamID VARCHAR2(2000) not null

AwayTeamID warchar2(2000) not null

Constraint MatchPK primary key(MatchID)

);

ALTER TABLE MATCH

ADD TOTAL SCORE

varchar(10) NOT NULL

Constraint TOTAL SCORE check (TOTAL SCORE = 'W', TOTAL SCORE = 'L', TOTAL SCORE = 'D')

**Question 4.**

* Data duplication or redundant: There are lots of duplication in multiple rows. For example, name and the date of birth of the patient is recorded in multiple times. There are also few redundant data like the provider number for Doctor’s name and item description for item.
* Insertion:the primary key attribute, null values cannot be inserted. Therefore, representation cannot be updated to include a new doctor's information without patient information. Similar to this, it is impossible to record new patient information without a doctor's name.
* Deletion: If we remove the doctor information associated with a certain doctor name, we will also lose all the patient information associated with that doctor. This is a significant loss for a company that makes an effort to keep all patient records current. To prevent this, we need redesign our database.
* Update:We must update numerous rows of the specified table in order to update our database to reflect that change. Due to the table's storage of redundant data, there is a significant danger of database inconsistency. In order to prevent situations like this, our system must be designed

I will design the tables to store Patient data, Doctor data, and Item data separately:

Patient table:

|  |  |  |
| --- | --- | --- |
| PATIENT ID | PATIENT NAME | PATIENT DOB |
| 437 | REARDON | 4/08/1989 |
| 562 | BILSTEIN | 12/05/2001 |
| 261 | WILSON | 13/08/1997 |
| 895 | LEMOS | 13/10/1952 |
| 678 | MORTAL | 27/02/1969 |
| 432 | SOAPIER | 1/08/1976 |

Item Table:

|  |  |  |
| --- | --- | --- |
| ITEM NO. | ITEM DESCRIPTION | FEE |
| A012 | standard | 56 |
| A013 | short | 35 |
| A014 | Long | 75 |

Doctors Table:

|  |  |
| --- | --- |
| PROVIDER NO. | DOCTORS NAME |
| S55768 | Leahy |
| D42433 | Meuller |

Patient appointment details Table:

|  |  |  |  |
| --- | --- | --- | --- |
| PATIENT ID | DOCTOR NAME | CONSULT DATE | ITEM NUMBER |
| 437 | Leahy | 19/07/2020 | A012 |
| 437 | Meuller | 20/07/2020 | A012 |
| 562 | Leahy | 20/07/2020 | A013 |
| 361 | Leahy | 20/07/2020 | A014 |
| 895 | Meuller | 20/07/2020 | A013 |

Question 5. (1)



(2)

Few assumptions that I considered:

* The flight has multiple reservation
* An Airplane may have multiple maintenance record
* Customer might has multiple reservation or none.