# ICT 285

# Databases

# S2 2017

# Assignment 1

BY ANANTH KADEKODI (32920719)

**Table of Contents**

1. Question 1- Relational Algebra Page 3
2. Question 2- SQL Select Queries Page 4
3. Question 3- Further SQL Page 7
4. Question 4- Normalisation Page 8
5. Question 5- Conceptual Design Page 10

**Question 1**

1. Π PresenterName, InstitutionName(Presenter)
2. Π WorkshopName, Description (σState = “Big Data” (WORKSHOP))
3. Π Attendeename(σTheme = “Big Data”((WORKSHOP\*WORKSHOP.WorkshopNo = PARTICIPANT.WorkshopNoPARTICIPANT)\*PARTICIPANT.AttendeeNo = ATTENDEE.AttendeeNoATTENDEE))

UNION

Π Attendeename(σTheme = “IoT”((WORKSHOP\*WORKSHOP.WorkshopNo = PARTICIPANT.WorkshopNoPARTICIPANT)\*PARTICIPANT.AttendeeNo = ATTENDEE.AttendeeNoATTENDEE))

1. Π PresenterName, Biography, InstitutionName (σState= “WA”(INSTITUTION\*INSTITUTION.InstitutionName = PRESENTER.InstitutionNamePRESENTER))
2. Π AttendeeName (σState = “Queensland” ((((INSTITUTION\*INSTITUTION.InstitutionName = PRESENTER.InstitutionNamePRESENTER)\*PRESENTER.Presentername = WORKSHOP.PresenterNameWORKSHOP)\*WORKSHOP.WorkshopNo = PARTICIPANT.WorkshopNoPARTICIPANT)\*PARTICIPANT.AttendeeNo = ATTENDEE.AttendeeNoATTENDEE))
3. Π WorkshopNo, WorkshopName, AttendeeName ((WORKSHOP\*WORKSHOP.WorkshopNo = PARTICIPANT.WorkshopNoPARTICIPANT)\*PARTICIPANT.AttendeeNo = ATTENDEE.AttendeeNoATTENDEE)
4. Π AttendeeName(σInstitutionName= “Murdoch University”AND Theme = “Green IT” AND State = “Tasmania”((((Institution\*INSTITUTION.InstitutionName=PRESENTER.InstitutionName\*PRESENTER)\*PRESENTER.PresenterName=WORKSHOP.PresenterName\*WORKSHOP)\*WORKSHOP.WorkshopNo=PARTICIPANT.WorkshopNo\*PARTICIPANT)\*PARTICIPANT.AttendeeNo=ATTENDEE.AttendeeNoATTENDEE))
5. Π AttendeeName(σWorkshopName = “Analytics for Everyone” AND WorkshopName = “Games and Reality” ((ATTENDEE\*ATTENDEE.AttendeeNo = PARTICIPANT.AttendeeNoPARTICIPANT)\* PARTICIPANT.WorkshopNo = WORKSHOP.WorkshopNoWORKSHOP)))
6. Π AttendeeName (σTheme = “Conference”((ATTENDEE\*ATTENDEE.AttendeeNo = PARTICIPANT.AttendeeNoPARTICIPANT)\*PARTICIPANT.WorkshopNo = WORKSHOP.WorkshopNoWORKSHOP))

MINUS

Π AttendeeName(σTheme = “Conference”AND “Description “Relational Algebra MasterClass”((ATTENDEE\*ATTENDEE.AttendeeNo = PARTICIPANT.AttendeeNoPARTICIPANT)\*PARTICIPANT.WorkshopNo = WORKSHOP.WorkshopNoWORKSHOP)))

1. Π AttendeeName(σTheme = “Conference” ((ATTENDEE\*ATTENDEE.AttendeeNo=PARTICIPANT.AttendeeNoPARTICIPANT)\* PARTICIPANT.WorkshopNo=WORKSHOP.WorkshopNoWORKSHOP))

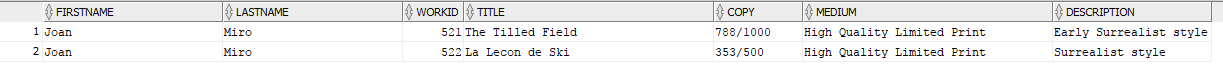
**Question 2**

1. SELECT FIRSTNAME, LASTNAME, WORKID, TITLE, COPY, MEDIUM, DESCRIPTION

FROM dtoohey.ARTIST A, dtoohey.WORK W

WHERE A.ARTISTID = W.ARTISTID

AND W.DESCRIPTION like '%Surrealist style%';



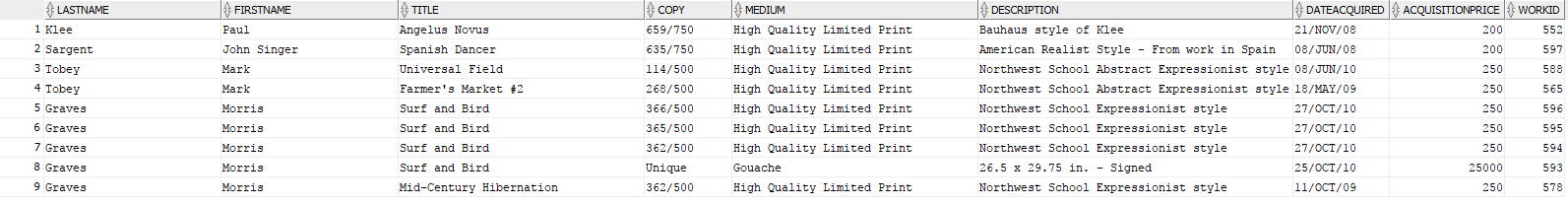
1. SELECT LASTNAME, FIRSTNAME, TITLE, COPY, MEDIUM, DESCRIPTION, DATEACQUIRED, ACQUISITIONPRICE, W.WORKID

FROM dtoohey.ARTIST A, dtoohey.WORK W, dtoohey.TRANS T

WHERE A.ARTISTID = W.ARTISTID

AND W.WORKID = T.WORKID

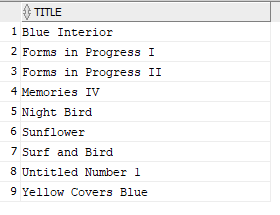
AND T.DATESOLD IS NULL;



1. SELECT TITLE

FROM dtoohey.WORK

WHERE COPY = 'Unique';



1. SELECT FIRSTNAME, LASTNAME, 2017-DATEOFBIRTH AS AGE

FROM dtoohey.ARTIST

WHERE DATEDECEASED IS NULL;



1. SELECT FIRSTNAME, LASTNAME, COPY AS NUMBEROFARTWORKS

FROM dtoohey.ARTIST A, dtoohey.WORK W

WHERE A.ARTISTID = W.ARTISTID

ORDER BY COPY DESC;



1. SELECT SUM(SALESPRICE - ACQUISITIONPRICE) AS TOTALPROFIT

FROM dtoohey.TRANS;



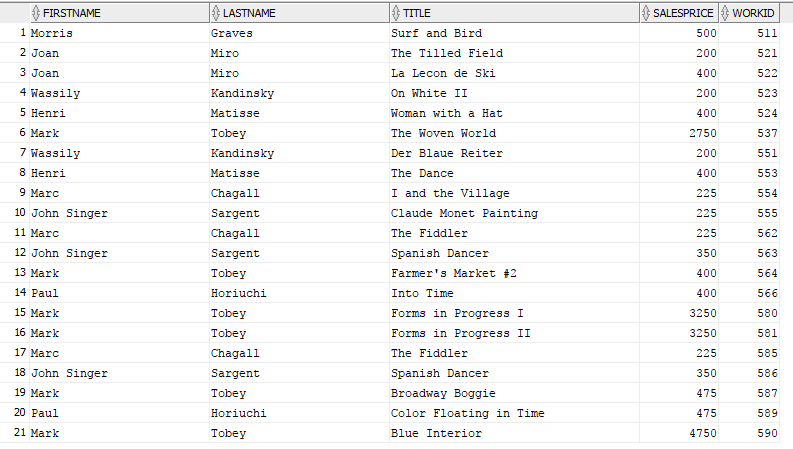
1. SELECT FIRSTNAME, LASTNAME, TITLE, SALESPRICE, T.WORKID

FROM dtoohey.ARTIST A, dtoohey.WORK W, dtoohey.TRANS T

WHERE A.ARTISTID = W.ARTISTID

AND W.WORKID = T.WORKID

AND SALESPRICE < (SELECT AVG(SALESPRICE) FROM dtoohey.TRANS);



1. SELECT DISTINCT FIRSTNAME, LASTNAME

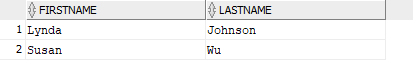
FROM dtoohey.CUSTOMER

MINUS

SELECT FIRSTNAME, LASTNAME

FROM dtoohey.CUSTOMER C, dtoohey.TRANS T

WHERE C.CUSTOMERID = T.CUSTOMERID;

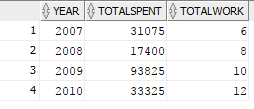


1. SELECT EXTRACT (YEAR FROM DATEACQUIRED) AS YEAR, SUM(ACQUISITIONPRICE) as TOTALSPENT, COUNT(TRANSACTIONID) as TOTALWORK

FROM dtoohey.TRANS

GROUP BY EXTRACT (YEAR FROM DATEACQUIRED)

ORDER BY YEAR;



1. SELECT DISTINCT C.CUSTOMERID, C.FIRSTNAME, C.LASTNAME

FROM dtoohey.ARTIST A, dtoohey.CUSTOMER C, dtoohey.CUSTOMER\_ARTIST\_INT T

WHERE A.ARTISTID = T.ARTISTID

AND C.CUSTOMERID = T.CUSTOMERID

AND A.NATIONALITY = 'Spanish'

ORDER BY CUSTOMERID ;



**Question 3**

1. CREATE TABLE PRESENTER

( PresenterName VARCHAR2(40) CONSTRAINT PresNamePK PRIMARY KEY

Biography VARCHAR2(500) NOT NULL,

InstitutionName VARCHAR2(40) NOT NULL);

1. CREATE TABLE WORKSHOP

(WorkshopNo NUMBER(5) NOT NULL,

CONSTRAINT WorkSpNoPK PRIMARY KEY(WorkshopNo),

WorkshopName VARCHAR2(50) NOT NULL,

Description VARCHAR2(1000),

Theme VARCHAR2(100) ,

Capacity NUMBER(4),

DateAndTime VARCHAR2(40) NOT NULL,

PresenterName VARCHAR2(40) NOT NULL,

CONSTRAINT PrsntrNmFK FOREIGN KEY (PresenterName) REFERENCES PRESENTER (PresenterName) ON DELETE CASCADE);

1. INSERT INTO PRESENTER (PresenterName, Biography, InstitutionName)

values('Ánanth Kadekodi', 'I have a passion for coding and software engineering. I am studying a Bachelor of Computer Science', 'Murdoch University');

1. ALTER TABLE WORKSHOP

ADD VenueName VARCHAR2(40);

1. UPDATE WORKSHOP

SET Capacity = 30

WHERE PresenterName = 'Laura Lopez'

AND WorkshopName = 'Smarter Cloud Computing';

**Question 4**

1. The existing design might exhibit three potential anomalies. These include: deletion anomaly, Insertion anomaly and Update anomaly

**Deletion Anomaly:** A deletion anomaly is the unintended loss of data due to deletion of other data. Deletion anomaly occurs, for example by removing Stan Wawrinka, not only removes his name but also removes the venue her played at, date of the event, location of the event, tournament. Therefore, there is a lot of information about the venues and the information about the player. This results in database inconsistencies and is an example of how combining information that does not really belong together into one table can cause problems. Hence, if we delete Stan Wawrinka we lose all the information regarding that offering.

**Insertion Anomaly:** An insertion anomaly is the inability to add data to the database due to absence of other data. Inserting a new player in the tournament means that, information about the player’s country, sponsor and events are also needed.

**Update Anomaly:** An update anomaly is a data inconsistency that results from data redundancy and a partial update. This occurs for example, by updating data in location, surface and venue can make the data in the table inconsistent. Eg. Changing the venue data in Melbourne tournament (18-31 Jan 2016) from Melbourne Park to Albert Park only for Men’s Singles can make the data inconsistent as this is the only tournament event with this venue. The remaining Melbourne Open tournament events are held at Melbourne Park. Additionally, changing the venue address means that we need to change more than one row and may end up doing it inconsistently.

1. The candidate key is Tournament, Winner, Dates

The normal form of the relation is currently in 1NF. This is due to the following reasons:

* 1. Rows contain data bout an entity
  2. Columns contain data about attributes of an entity
  3. All entities in a column are one of a kind
  4. Each column has a unique name
  5. The cells of the table hold a single value
  6. The order of the columns is unimportant
  7. The order of the rows is unimportant
  8. No two rows may be identical
  9. Has potential for modification anomalies
  10. Has a valid relation
  11. Has a partial FD (Winner -> country, sponsor )

1. Relation List:
   1. Tournament -> venue, location, surface

TournamentRelation(tournament, venue, location, surface)

Winner -> country, sponsor,

WinnerRelation(winner, country, sponsor)

Dates -> Year

DateRelation (dates, year)

Tournament, Winner, Dates -> Event

Hence, 3NF Key is Tournament, Winner, Dates

3NFRelation (**tournament, winner, dates,** event)

The above key assumes that the winner can be in both single’s and double’s event (not demonstrated in the table). This makes prevents the event from solely relying on one of the primary key (tournament or winner or dates) and is essential for creating the 3NF relation.

1. The above relation is in 3NF as all the key-attributes are fully and non-transitively dependent on the candidate key. Additionally, there are also no modification anomalies present. Additionally, the above relationship is 2NF PLUS no transitive functional dependencies. It also has the dependency preserving and lossless join properties.

For eg – Country and sponsor which depend on the winner, are within the WINNER table. Additionally, venue, location and surface which depend on the tournament are within the tournament table. This allows for the amount of data duplication to be reduced (database is smaller) and data integrity. There Is increased data integrity as the data is not spread out unlike an increased risk of updating only some of the data if it is spread out.

For eg: If the sponsor of a winner changes, we just change a single record in the WINNER databases (update anomaly). It is also possible to add the winner details within the WINNER table independent of the specific tournament he/she has won (insertion anomaly). Lastly, we can also delete the information about the winner without altering the values of the tournament and the dates (deletion anomaly).

**Question 5**

