Name: Celebrate

Surname: Mashaba

Student number: ST10055395

Module name: DATABASE

Module code: DATA6222

Assessment type: Assignment 1

Group 1

CREATE DATABASE ST10055395ASSIGNMENT1;

Name: Celebrate

Surname: Mashaba

Student number: ST10055395

Module name: Database

Module code: Data6211

Assessment type: Assignment

Group 1

CREATE DATABASE ST10055395ASSIGNMENT1;

--CREATING TABLES

--CREATE Author TABLE

CREATE TABLE Author(

Author\_ID INT PRIMARY KEY not null,

name VARCHAR(50) not null,

surname VARCHAR(50) not null

);

--CREATE Genre TABLE

CREATE TABLE Genre(

Genre\_ID INT PRIMARY KEY not null,

Description VARCHAR(250) not null

);

--CREATE Book TABLE

CREATE TABLE Book(

Book\_ID INT PRIMARY KEY not null,

Title VARCHAR(100) not null,

Genre\_ID INT FOREIGN KEY REFERENCES Genre(Genre\_ID),

Author\_ID INT FOREIGN KEY REFERENCES Author(Author\_ID)

);

--CREATE Competition TABLE

CREATE TABLE Competition(

CompetitionID INT PRIMARY KEY not null,

Description VARCHAR(250) not null

);

--CREATE Entry TABLE

CREATE TABLE Entry(

Entry\_ID INT PRIMARY KEY not null,

Book INT FOREIGN KEY REFERENCES Book(BooK\_ID),

CompetitionID INT FOREIGN KEY REFERENCES Competition(CompetitiOnID)

);

--STATEMENT TO INSERT RELEVANT DATA INTO THE TABLE

--INSERT AUTHOR INTO THE TABLE

insert into Author(Author\_ID,name,surname) Values

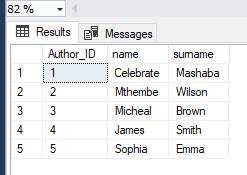
(1, 'Celebrate', 'Mashaba'),

(2, 'Mthembe', 'Wilson'),

(3, 'Micheal', 'Brown'),

(4, 'James', 'Smith'),

(5, 'Sophia', 'Emma');



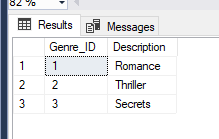
--INSERT GENRE INTO THE TABLE

insert into Genre(Genre\_ID,Description) Values

(1 , 'Romance'),

(2, 'Thriller'),

(3, 'Secrets');



--INSERT AT LEAST 20 RECORDS IN THE BOOK TABLE

--INSERT BOOK INTO THE TABLE

insert into Book(Book\_ID,Title,Genre\_ID,Author\_ID) Values

(1, 'The One After Mars',1,1),

(2, 'Dangerous moon', 2,2),

(3, 'Earth to unknown', 3,3),

(4, 'Horror Sories in the dark',2,2),

(5, 'Skeletons in the house', 1,1),

(6, 'Watch out', 1,1),

(7, 'A party for ghouls', 3,3),

(8, 'Blank paper', 1,1),

(9, 'Long road to go',2,2),

(10, 'Time to go', 3,3),

(11, 'Unsolved crimws', 3,3),

(12, 'The secrets of mr Miller',1,1),

(13, 'The meaning of Z',3,3),

(14, 'Who stole my Homework',3,3),

(15, 'The lost portrait',2,2),

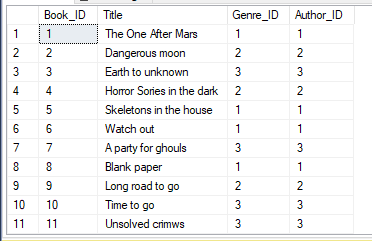
(16, 'Never to be solved',2,2),

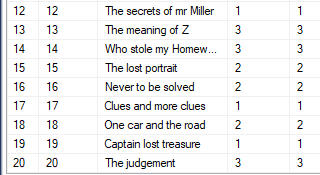
(17, 'Clues and more clues',1,1),

(18, 'One car and the road',2,2),

(19, 'Captain lost treasure',1,1),

(20, 'The judgement',3,3);





--INSERT AT LEAST 15 RECORDS IN THE COMPETITION TABLE

--INSERT COMPETITION INTO THE TABLE

insert into Competition(CompetitionID,Description) Values

(1, 'Best mystery novel'),

(2, 'Romantic book of the year'),

(3, 'The Deadly Game'),

(4, 'Secrets unveiled'),

(5, 'In cold blood'),

(6, 'Love at first sight'),

(7, 'Thriller writing challenge'),

(8, 'The mystery of the missing key'),

(9, 'The bad guy'),

(10, 'Cowboys in a storm'),

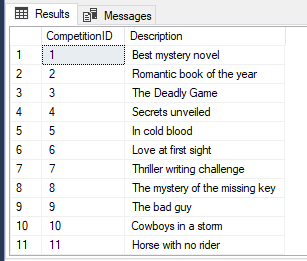
(11, 'Horse with no rider'),

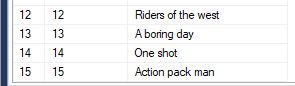
(12, 'Riders of the west'),

(13, 'A boring day'),

(14, 'One shot'),

(15, 'Action pack man');





--INSERT AT LEAST ONE BOOK ENTRY FOR MORE THAN ONE COMPETITION

--INSERT ENTRY INTO THE TABLE

insert into Entry(Entry\_ID,Book\_ID,CompetitionID) VALUES

(1,1,1),

(2,2,2),

(3,1,1),

(4,2,2),

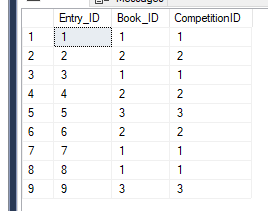
(5,3,3),

(6,2,2),

(7,1,1),

(8,1,1),

(9,3,3);



SELECT \* FROM Author;

SELECT \* FROM Genre;

SELECT \* FROM Book;

SELECT \* FROM Competition;

SELECT \* FROM Entry;

--USING UPDATE STATEMENT

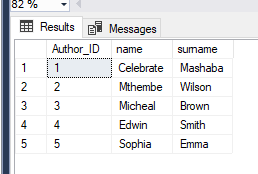
--update Author statement

UPDATE Author

SET name = 'Edwin'

WHERE Author\_ID = 4;

SELECT \* FROM Author;



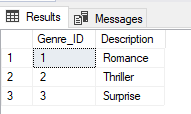
--update Genre statement

UPDATE Genre

SET Description = 'Surprise'

WHERE Genre\_ID = 3;

SELECT \* FROM Genre



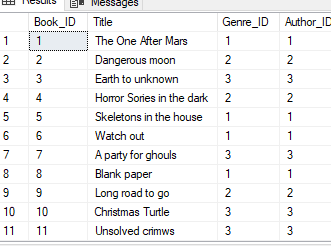
--update Book statement

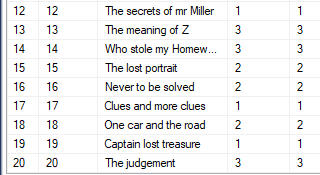
UPDATE Book

SET Title = 'Christmas Turtle'

WHERE Book\_ID = 10;

SELECT \* FROM Book;





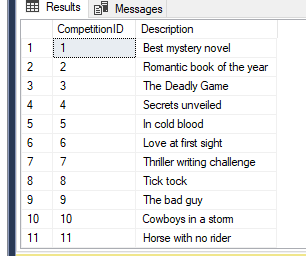
--update competition statement

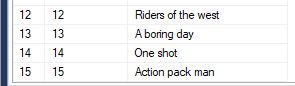
UPDATE Competition

SET Description = 'Tick tock'

WHERE CompetitionID = 8;

SELECT \* FROM Competition;





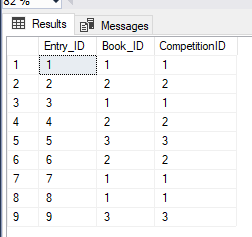
--update Entry statement

UPDATE Entry

SET CompetitionID = 2

WHERE Book\_ID = 4;

SELECT \* FROM Entry;



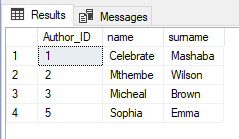
--USING DELETE STATEMENT

--DELETE Author statement

DELETE FROM Author

WHERE Author\_ID = 4;

SELECT \* FROM Author;



--DELETE GENRE STATEMENT

DELETE FROM Genre

WHERE Genre\_ID =2;

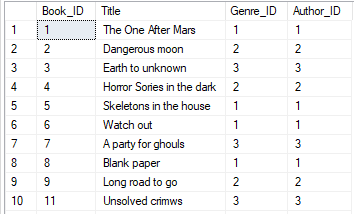
SELECT \* FROM Genre;

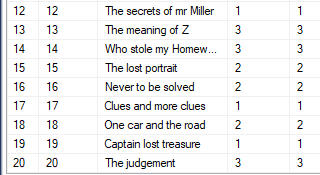
--DELETE Book statement

DELETE FROM Book

WHERE Book\_ID = 10;

SELECT \* FROM Book;



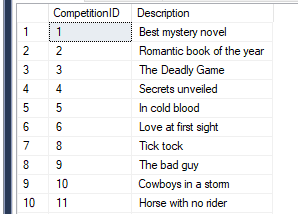


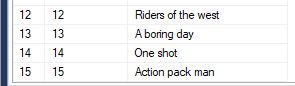
--DELETE COMPETITION STATEMENT

DELETE FROM Competition

WHERE CompetitionID= 7;

SELECT \* FROM Competition;



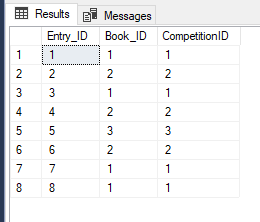


--DELETE ENTRY STATEMENT

DELETE FROM Entry

WHERE Entry\_ID = 9;

SELECT \* FROM Entry;



--DISPLAY A RPORT OF THE BOOK, TITILES, AUTHOR AND GENRE

SELECT b.Title AS 'Title',

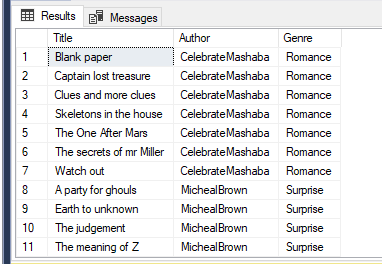
CONCAT (a.name, '',a.surname) AS 'Author', g.Description AS 'Genre'

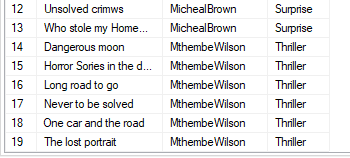
From Book b

JOIN Author a on b.Author\_ID = a.Author\_ID

JOIN Genre g ON b.Genre\_ID = g.Genre\_ID

ORDER BY g.Description ASC , B.Title ASC;



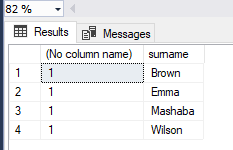


--Group by

SELECT COUNT (Author\_ID),surname

FROM Author

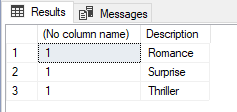
GROUP BY surname



SELECT COUNT (Genre\_ID),Description

FROM Genre

GROUP BY Description

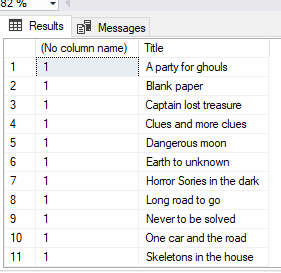


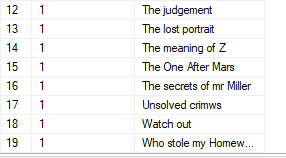
--Group by

SELECT COUNT (Book\_ID),Title

FROM Book

GROUP BY Title



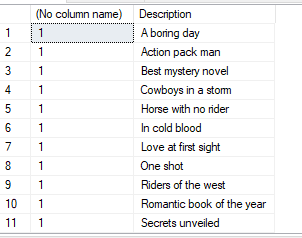


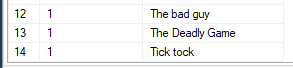
--Group by

SELECT COUNT (CompetitionID),Description

FROM Competition

GROUP BY Description





SELECT COUNT (Entry\_ID),Book\_ID

FROM Entry

GROUP BY Book\_ID

--having clause

SELECT Author\_ID, MIN (surname) AS surname

FROM Author

Group by Author\_ID

HAVING COUNT (Author\_ID) <50

ORDER BY Author\_ID

SELECT Genre\_ID, MIN ( Description) AS Descrption

FROM Genre

GROUP BY Genre\_ID

HAVING COUNT (Genre\_ID) < 250

ORDER BY Genre\_ID

SELECT CompetitionID, MIN (Description) AS Descrption

FROM Competition

GROUP BY CompetitionID

HAVING COUNT (CompetitionID) < 250

ORDER BY CompetitionID

SELECT Entry\_ID, MIN (Book\_ID) AS Book\_ID

FROM Entry

GROUP BY Entry\_ID

HAVING COUNT ( Entry\_ID) <50

ORDER BY Entry\_ID

--JOIN STATEMENT

--INNER JOIN STATEMENT BETWEEN AUTHOR AND GENRE

SELECT Author\_ID

FROM Author

inner join Genre

ON Author.Author\_ID = Genre.Genre\_ID

SELECT Description

From Competition

left join Book

ON Competition.Description = Book.Title

SELECT Title

FROM Book

left join Competition

on Book.Title =Competition.Description