Create Database IF NOT EXISTS University;

Use University;

Create Table if not exists Student (

S\_ID VARCHAR(10) NOT NULL PRIMARY KEY,

S\_NAME VARCHAR(50) NOT NULL,

S\_SURNAME VARCHAR(50) NOT NULL,

C\_ID INT NOT NULL,

Foreign key (C\_ID) References COURSE (C\_ID),

A\_ID VARCHAR(10) NOT NULL,

Foreign key (A\_ID) References advisors (A\_ID)

);

CREATE TABLE IF NOT EXISTS Advisors (

A\_ID VARCHAR(10) PRIMARY KEY NOT NULL,

A\_NAME VARCHAR(50),

D\_NAME VARCHAR(50)

);

CREATE TABLE IF NOT EXISTS COURSE (

C\_ID INT PRIMARY KEY NOT NULL,

C\_NAME VARCHAR(50),

D\_Name VARCHAR(50) NOT NULL

);

CREATE TABLE IF NOT EXISTS Grade (

S\_ID VARCHAR(10) NOT NULL,

S\_GRADE VARCHAR(50),

C\_ID INT NOT NULL

);

Alter Table Grade

ADD CONSTRAINT fk\_S\_ID

FOREIGN KEY (S\_ID) REFERENCES Student (S\_ID);

Alter Table Grade

ADD CONSTRAINT fk\_C\_ID

FOREIGN KEY (C\_ID) REFERENCES COURSE (C\_ID);

INSERT INTO Advisors (A\_ID, A\_NAME, D\_NAME)

VALUES ('HUD122', 'Arthur Ink', '101'),

('HUD134', 'Sean Gill', '102'),

('HUD156', 'Tariq Alhasaan', '103'),

('HUD214', 'Yazeed Janani', '104'),

('HUD129', 'Fatimah Usen', '105'),

('HUD176', 'Andre Mo', '106');

INSERT INTO Course (C\_NAME, D\_NAME, C\_ID)

VALUES ('English', 'Languages', '1011'),

('French', 'Languages', '1012'),

('Spanish', 'Languages', '1013'),

('Foundations in Mathematics', 'Mathematics', '1021'),

('Advanced Mathematics', 'Mathematics', '1022'),

('Accounting', 'Mathematics', '1023'),

('Biology', 'Science', '1031'),

('Chemistry', 'Science', '1032'),

('Physics', 'Science', '1034'),

('Sports Science', 'Science', '1035'),

('Law', 'Social Sciences', '1041'),

('Sociology', 'Social Sciences', '1042'),

('History', 'Social Sciences', '1043'),

('Criminology', 'Social Sciences', '1044'),

('Fine Art', 'Art', '1051'),

('Art History', 'Art', '1052'),

('Architecture', 'Art', '1053'),

('Foundations in It', 'IT', '1061'),

('Cyber Security', 'IT', '1062'),

('Software Engineering', 'IT', '1063');

INSERT INTO Grade (S\_ID, S\_GRADE, C\_ID)

VALUES ('ADA435', 'Pass', '1022'),

('ADE145', 'Pass', '1035'),

('BRI594', 'Fail', '1044'),

('CLI985', 'Pass', '1035'),

('DOU596', 'Pass', '1032'),

('IFE405', 'Fail', '1063'),

('IND895', 'Fail', '1022'),

('INO023', 'Fail', '1011'),

('KUN980', 'Pass', '1022'),

('MOH897', 'Pass', '1051'),

('NEI903', 'Pass', '1044'),

('SUL432', 'Pass', '1061'),

('WIN345', 'Fail', '1043'),

('YOS513', 'Fail', '1011'),

('YUP894', 'Fail', '1044'),

('SUR145', 'Fail', '1061'),

('BDU567', 'Pass', '1062'),

('GUS980', 'Fail', '1063'),

('UGO678', 'Pass', '1011'),

('HEU768', 'Pass', '1043');

INSERT INTO Student (S\_ID, S\_NAME, S\_SURNAME, C\_ID, A\_ID)

VALUES ('ADA435', 'Charles', 'Adams', '1022', 'HUD122'),

('ADE145', 'Muhammed', 'Adel', '1035', 'HUD134'),

('BRI594', 'Susan', 'Bright', '1044', 'HUD156'),

('CLI985', 'David', 'Cliff', '1035', 'HUD214'),

('DOU596', 'Miraj', 'Dou', '1032', 'HUD129'),

('IFE405', 'Maurice', 'Ifeanyi', '1063', 'HUD134'),

('IND895', 'Vit', 'Indel', '1022', 'HUD156'),

('INO023', 'Goodness', 'Innocent', '1011', 'HUD214'),

('KUN980', 'Owans', 'Kung', '1022', 'HUD176'),

('MOH897', 'Sobur', 'Moh', '1051', 'HUD122'),

('NEI903', 'Ada', 'Neil', '1044', 'HUD134'),

('SUL432', 'Raheel', 'Sultan', '1061', 'HUD156'),

('WIN345', 'Uty', 'Winnesh', '1043', 'HUD214'),

('YOS513', 'Adeline', 'Yosuef', '1011', 'HUD176'),

('YUP894', 'Taygha', 'Yul', '1044', 'HUD122'),

('SUR145', 'Lilian', 'Vel', '1061', 'HUD134'),

('BDU567', 'Jo', 'Thomas', '1062', 'HUD156'),

('GUS980', 'Afifah', 'Rohaan', '1063', 'HUD214'),

('UGO678', 'Minsah', 'Atfat', '1011', 'HUD129'),

('HEU768', 'Fahmi', 'Yeraz', '1043', 'HUD134');

SELECT \* FROM Student;

DELIMITER //

CREATE FUNCTION countpassandfail(C\_ID INT)

RETURNS VARCHAR(100)

READS SQL DATA

BEGIN

DECLARE passCount INT;

DECLARE failCount INT;

DECLARE result VARCHAR(100);

SELECT

COUNT(CASE WHEN Grade = 'pass' THEN 1 ELSE NULL END) AS passCount,

COUNT(CASE WHEN Grade = 'fail' THEN 1 ELSE NULL END) AS failCount

INTO passCount, failCount

FROM Grade

WHERE Course\_ID = C\_ID;

SET result = CONCAT('Passes: ', passCount, ', Fails: ', failCount);

-- Return the result as a single string

RETURN result;

END //

DELIMITER ;