DBS PROJECT : STUDENT MANAGEMENT SYSTEM

PROCEDURES.

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewTracksHabit`( cms\_id INT)

BEGIN

select \*

from tracks

where cms\_id=S\_CMS;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewTakes`( cms\_id INT)

BEGIN

select C\_ID,Course\_Name

from takes a join course c on a.C\_ID=c.Course\_ID

where cms\_id=S\_CMS;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewProjects`(\_planner\_ID int)

BEGIN

select \* from project

where \_planner\_ID=CP\_ID;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewPersonalActivity`()

BEGIN

select \*

from personal\_activity;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewPerformsActivity`( cms\_id INT)

BEGIN

select \*

from performs

where cms\_id=S\_CMS;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewJobs`(\_planner\_ID int)

BEGIN

select \* from job\_application

where \_planner\_ID=CP\_ID;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewInternships`(\_planner\_ID int)

BEGIN

select \* from internship

where \_planner\_ID=CP\_ID;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewHabit`()

BEGIN

select \*

from habit;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `viewCourses\_1Student`(\_cms int)

BEGIN

select C\_ID,Course\_Name, T\_Assignment, T\_Quiz, T\_Exam, Percentage, C\_Grade

from takes t join course c on c.Course\_ID= t.C\_ID

where \_cms=S\_CMS;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewCourse`()

BEGIN

select Course\_ID,Course\_Name

from course;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewAttemptsQuiz`(

\_course\_ID int,cms\_id INT)

BEGIN

select \*

from attempts\_quiz

where cms\_id=S\_CMS and (Q\_ID=\_course\_ID\*2+1 or Q\_ID=\_course\_ID\*2+2);

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewAttemptsExam`(\_course\_ID INT, cms\_id INT)

BEGIN

select \*

from attempts\_exam

where cms\_id=S\_CMS and (E\_ID=\_course\_ID\*2+1 or E\_ID=\_course\_ID\*2+2);

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ViewAttemptsAssignment`(\_course\_ID INT, cms\_id INT)

BEGIN

select \*

from attempts\_assignment

where cms\_id=S\_CMS and (A\_ID=\_course\_ID\*2+1 or A\_ID=\_course\_ID\*2+2);

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `Takes\_Course`(

\_cms int ,\_course\_ID int)

BEGIN

insert into takes(S\_CMS,C\_ID)

values(\_cms,\_course\_ID);

insert into attempts\_quiz(Q\_ID,S\_CMS,Q\_Status,Obtained\_Marks,Grade\_Point)

values(\_course\_ID\*2+1,\_cms,"pending",0,0),(\_course\_ID\*2+2,\_cms,"pending",0,0);

insert into attempts\_assignment(A\_ID,S\_CMS,A\_Status,Obtained\_Marks,Grade\_Point)

values(\_course\_ID\*2+1,\_cms,"pending",0,0),(\_course\_ID\*2+2,\_cms,"pending",0,0);

insert into attempts\_exam(E\_ID,S\_CMS,E\_Status,Obtained\_Marks,Grade\_Point)

values(\_course\_ID\*2+1,\_cms,"pending",0,0),(\_course\_ID\*2+2,\_cms,"pending",0,0);

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `StudentAddOrEdit`(

\_cms int(11),

\_firstname varchar(10),

\_lastname varchar(10),

\_bday date,

\_sex varchar(10),

\_semesternum int(11),

\_numcourses int(11),

\_city varchar(15),

\_department varchar(20)

)

BEGIN

if \_cms not in (select CMS FROM student)

then

insert into Student(CMS,Firstname,Lastname,Birthdate,Sex,Semester\_Num,Num\_Courses,City,Department)

values (\_cms,\_firstname,\_lastname,\_bday,\_sex ,\_semesternum,\_numcourses,\_city,\_department);

END if;

End

CREATE DEFINER=`root`@`localhost` PROCEDURE `QuizAttempt`(

\_courseID int,

\_cms int,

\_QID int,

\_status varchar(10),

\_obtainedmarks int

)

BEGIN

declare GP int;

set GP=(\_obtainedmarks/10)\*10;

if \_QID in(select Quiz\_ID from quiz)

then

update attempts\_quiz

set Q\_Status=\_status,Obtained\_Marks=\_obtainedmarks,

Grade\_Point=GP

where \_QID=Q\_ID and S\_CMS=\_cms;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `GPA`(\_cms int)

BEGIN

select\*from result where S\_CMS=\_cms;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `GP\_Quiz`( \_cms int)

BEGIN

DECLARE GradePoint float;

SET GradePoint = ((SELECT Obtained\_Marks FROM attempts\_quiz where S\_CMS=\_cms)/

(select Total\_Marks from quiz join takes using (C\_ID) where S\_CMS=\_cms))\*

(select Weightage from quiz join takes using (C\_ID) where S\_CMS=\_cms);

update attempts\_quiz set Grade\_Point = GradePoint where S\_CMS=\_cms;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `ExamAttempt`(

\_courseID int,

\_cms int,

\_EID int,

\_status varchar(10),

\_obtainedmarks int

)

BEGIN

declare GP int;

set GP=(\_obtainedmarks/100)\*25;

if \_EID in(select Exam\_ID from exam)

then

update attempts\_exam

set E\_Status=\_status,Obtained\_Marks=\_obtainedmarks,

Grade\_Point=GP

where \_EID=E\_ID and S\_CMS=\_cms;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `CourseResult`(

\_cms int,\_cid int)

BEGIN

Declare P int;

Declare T int;

set T=(select Grade\_Point from Attempts\_Quiz where Q\_ID=\_cid\*2+1 and S\_CMS=\_cms)

+(select Grade\_Point from Attempts\_Quiz where Q\_ID=\_cid\*2+2 and S\_CMS=\_cms);

update takes

set T\_Quiz= T

where \_cms=S\_CMS and C\_ID=\_cid;

set T=(select Grade\_Point from Attempts\_assignment where A\_ID=\_cid\*2+1 and S\_CMS=\_cms)

+(select Grade\_Point from Attempts\_assignment where A\_ID=\_cid\*2+2 and S\_CMS=\_cms);

update takes

set T\_Assignment= T

where \_cms=S\_CMS and C\_ID=\_cid;

set T=(select Grade\_Point from Attempts\_exam where E\_ID=\_cid\*2+1 and S\_CMS=\_cms)

+(select Grade\_Point from Attempts\_exam where E\_ID=\_cid\*2+2 and S\_CMS=\_cms);

update takes

set T\_Exam=T

where \_cms=S\_CMS and C\_ID=\_cid;

set P=(select T\_Exam from takes where S\_CMS=\_cms and C\_ID=\_cid)+

(select T\_Assignment from takes where S\_CMS=\_cms and C\_ID=\_cid)+

(select T\_Quiz from takes where S\_CMS=\_cms and C\_ID=\_cid);

update takes

set Percentage=P

where \_cms=S\_CMS and C\_ID=\_cid;

if((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)<50)

then update takes set C\_Grade='F' where \_cms=S\_CMS and C\_ID=\_cid;

end if;

if((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)>=50)

then update takes set C\_Grade='E'where \_cms=S\_CMS and C\_ID=\_cid;

end if;

if((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)>=60)

then update takes set C\_Grade='D'where \_cms=S\_CMS and C\_ID=\_cid;

end if;

if((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)>=70)

then update takes set C\_Grade='C'where \_cms=S\_CMS and C\_ID=\_cid;

end if;

if((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)>=80)

then update takes set C\_Grade='B'where \_cms=S\_CMS and C\_ID=\_cid;

end if;

if ((select Percentage from takes where S\_CMS=\_cms and C\_ID=\_cid)>=90)

then

update takes set Grade='A'where \_cms=S\_CMS and C\_ID=\_cid;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `CourseAddOrEdit`(

\_courseID INT,

\_cms int(11),

\_credithrs int,

\_coursename varchar(50),

\_instrName varchar(50),

\_instrEmail varchar(50),

\_OfficeLoc varchar(50),

\_mblNum int(11)

)

BEGIN

if \_courseId not in (select Course\_ID FROM course)

then

insert into Course(Course\_ID,Course\_Name,Credit\_Hours,Teacher\_Name,Email,Office\_Location,Mobile)

values (\_courseID, \_credithrs, \_coursename,\_instrName,\_instrEmail,\_OfficeLoc,\_mblNum);

insert into takes(S\_CMS,C\_ID,T\_Assignment,T\_Quiz) values(\_cms,\_courseID,0,0,0,0);

insert into quiz(C\_ID,Quiz\_ID,Total\_Marks,Q\_Date,Weightage)

values(\_courseID, \_courseID\*2+1, 10, "2021-12-11",10 ), (\_courseID, \_courseID\*2+2, 10, "2021-10-11",10 );

insert into assignment(CID, Assignment\_ID,Total\_Marks,Deadline,Weightage)

values(\_courseID, \_courseID\*2+1, 10, "2012-06-15", 10),(\_courseID, \_courseID\*2+2, 10, "2012-08-15", 10);

insert into exam(C\_ID,Exam\_ID,Venue,Total\_Marks,E\_Date,Weightage)

values(\_courseID, \_courseID\*2+1, "Exam Hall", 10, "2012-06-15", 10),(\_courseID, \_courseID\*2+2, "SEECS",10, "2012-08-15", 10);

insert into takes(S\_CMS,C\_ID)

values (\_cms,\_courseID);

END IF;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `CareerAddOrEdit`(

\_cms int(11),

\_plannerID int,

\_JobId int,

\_JStatus varchar(20),

\_JComp varchar(60),

\_JDeadline date,

\_JExp varchar(100),

\_ProjId int,

\_PStatus varchar(20),

\_PComp varchar(60),

\_PTimePeriod varchar(20),

\_Ptitle varchar(50),

\_IID int,

\_IStatus varchar(20),

\_IComp varchar(60),

\_ITimePeriod varchar(20)

)

BEGIN

if \_plannerID not in (select P\_ID FROM careerplanner)

then

insert into careerplanner(S\_CMS,P\_ID)

values (cms,\_plannerID);

END IF;

IF \_JobId not in (select Application\_ID from job\_application)

then

insert into job\_application(CP\_ID, Application\_ID, J\_Status, Company, Deadline, Experience )

values(\_plannerID, \_JobId,\_JStatus,\_JComp,\_JDeadline,\_JExp );

else

update job\_application

set J\_Status=\_JStatus

where \_plannerID=CP\_ID and \_JobId=Application\_ID;

END IF;

IF \_ProjId not in (select Project\_ID from project)

then

insert into project(CP\_ID, Project\_ID,P\_Status, CompanyorInstitute,Time\_period,Title )

values(\_plannerID, \_ProjId,\_PStatus,\_PComp ,\_PTimePeriod ,\_Ptitle);

else

update project

set P\_Status=\_PStatus

where \_plannerID=CP\_ID and \_ProjId=Project\_ID;

END IF;

IF \_IID not in (select Internship\_ID from internship)

then

insert into internship(CP\_ID, Internship\_ID,I\_Status,Company,Time\_period)

values(\_plannerID, \_IID,\_IStatus,\_IComp, \_ITimePeriod);

else

update internship

set I\_Status=\_IStatus

where \_plannerID=CP\_ID and \_IID=Internship\_ID;

END IF;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `Calculate\_GPA`(\_cms int)

BEGIN

declare \_GPA float;

declare sum float;

set \_GPA=((select sum((Percentage/100)\*Credit\_Hours) from takes join course on takes.C\_ID = course.Course\_ID

group by S\_CMS having S\_CMS=\_cms)/(select sum(Credit\_Hours) from course join takes on takes.C\_ID = course.Course\_ID

where S\_CMS=\_cms))\*4;

if \_cms not in(select S\_CMS from result)

then insert into result(S\_CMS,GPA)

values(\_cms,\_GPA);

else

update result

set GPA=\_GPA where S\_CMS=\_cms;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `AssignmentAtttempt`(

\_courseID int,

\_cms int,

\_AID int,

\_status varchar(10),

\_obtainedmarks int

)

BEGIN

if \_AID in(select Assignment\_ID from Assignment)

then

update attempts\_assignment

set A\_Status=\_status,Obtained\_Marks=\_obtainedmarks,

Grade\_Point= (\_obtainedmarks/

(select Total\_Marks from Assignment join takes using (C\_ID) where S\_CMS=\_cms))\*

(select Weightage from Assignment join takes using (C\_ID) where S\_CMS=\_cms)

where A\_ID=\_AID;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `AssignmentAttempt`(

\_courseID int,

\_cms int,

\_AID int,

\_status varchar(10),

\_obtainedmarks int

)

BEGIN

declare GP int;

set GP=(\_obtainedmarks/20)\*15;

if \_AID in(select Assignment\_ID from Assignment)

then

update attempts\_assignment

set A\_Status=\_status,Obtained\_Marks=\_obtainedmarks,

Grade\_Point=GP

where \_AID=A\_ID and S\_CMS=\_cms;

end if;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddorEditPersonalAct`(

\_ActivityID int,

\_cms int,

\_ActName varchar(30),

\_AStatus varchar (20),

ATimePeriod varchar (20)

)

BEGIN

if \_ActivityID not in (select Activity\_ID FROM personal\_activity)

then

insert into personal\_activity (Activity\_ID,Act\_Name)

values (\_ActivityID,\_ActName);

insert into performs(S\_CMS, A\_ID, A\_Status, Time\_Period)

values(\_cms, \_ActivityID, \_AStatus, ATimePeriod);

else

if \_ActivityID not in (select A\_ID FROM performs where S\_CMS=\_cms)

then insert into performs(S\_CMS, A\_ID, A\_Status, Time\_Period)

values(\_cms, \_ActivityID, \_AStatus, ATimePeriod);

end if;

update performs

set A\_ID=\_ActivityID,

A\_Status=\_AStatus,

Time\_Period=ATimePeriod

where S\_CMS=\_cms AND A\_ID=\_ActivityID;

end if;

end

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddorEditHabit`(

\_habitID int,

\_cms int,

\_habitName varchar(30),

\_Duration time,

\_Allocated\_Time time,

\_Time\_Remaining time

)

BEGIN

if \_habitID not in (select Habit\_ID FROM habit)

then

insert into Habit(Habit\_ID,Habit\_Name)

values (\_habitID,\_habitName);

insert into tracks(S\_CMS, H\_ID, Duration, Allocated\_Time,Time\_Remaining)

values(\_cms, \_habitID, \_Duration, \_Allocated\_Time, \_Time\_Remaining);

else

if \_habitID not in (select H\_ID FROM tracks where S\_CMS=\_cms)

then insert into tracks(S\_CMS, H\_ID, Duration, Allocated\_Time,Time\_Remaining)

values(\_cms, \_habitID, \_Duration, \_Allocated\_Time, \_Time\_Remaining);

end if;

update tracks

set

Duration=\_Duration,

Allocated\_Time=\_Allocated\_Time,

Time\_Remaining=\_Time\_Remaining

where S\_CMS=\_cms and H\_ID=\_habitID;

end if;

end

SQL QUERY

CREATE DATABASE Student\_Management\_System;

CREATE TABLE Student

(

CMS int not NULL,

Firstname varchar(10),

Lastname varchar(10),

Birthdate date not null,

Sex varchar(10) check ( Sex in('Other','Male','Female',NULL)),

Semester\_Num int,

Num\_Courses int,

City varchar(15),

Department varchar(20),

CONSTRAINT pk\_student

PRIMARY KEY (CMS)

);

INSERT INTO Student

VALUES (1,"Iman","Imtiaz","2001-06-09","Female","3","4","Islamabad","SEECS"),

(2,"Bisma","Ijaz","2000-02-04","Female","3","4","Lahore","SEECS"),

(3,"Iman","Imtiaz","1999-09-21","Female","3","4","Islamabad","SEECS"),

(4,"Awais","Khan","2000-07-31","Male","3","5","Rawalpindi","SNS"),

(5,"Ahtisham","Malik","1998-01-13","Male","6","2","Karachi","S3H");

CREATE TABLE Course

(

Course\_ID int not NULL,

Course\_Name varchar(50),

Credit\_Hours int,

Teacher\_Name varchar(50),

Email varchar(50),

Office\_Location varchar(50),

Mobile bigint,

CONSTRAINT pk\_Course

PRIMARY KEY (Course\_ID)

);

INSERT INTO Course

VALUES (1,"Object Oriented Programming",4,"Mr.Jaudat Mamoon","J.Mamoon24680@gmail.com","Room-No:321,Faculty Block,SEECS",03468952460),

(2,"Applied Physics",4,"Dr.Muhammad Imran","M.Imran213@gmail.com","Room-No:18,Faculty Block,SNS",03183620582),

(3,"Fundamentals Of ICT",4,"Mr.Saad Qureshi","SaadQureshi5438@gmail.com","Room-No:638,Faculty Block,IAEC",03028502849),

(4,"Discrete",3,"Atifa Malik","Atifa.Malik80@gmail.com","Room-No:239,Faculty Block SNS",03839103103),

(5,"Communication Skills",2,"Mr.Sohaib Riaz","SohaibRaiz783@yahoo.com","Room-No:294,Faculty Block S3H",03309492903),

(6,"Technical and Buisness",2,"Huma Malik","HumaMalik@gmail.com","Room-No:8,Faculty Block S3H",03234848922),

(7,"Communication Skills",3,"Mr.Hamza Tahir","HamzaTahir73@gmail.com","Room-No:212,Faculty Block S3H",03735948939),

(8,"Technical and Buisness",3,"Hassan Javed","HassanJaved236@gmail.com","Room-No:23,Faculty Block S3H",03392849493),

(9,"Data Structure and algorithms",4,"Mr.Yasir Faheem","Y.Faheem220@gmail.com","Room-No:351,Faculty Block SEECS",03824949024),

(10,"Linear Algebra",4,"Syed Afzal","S.Afzal680@gmail.com","Room-No:42,Faculty Block S3H",03194008480);

CREATE TABLE Takes

(

S\_CMS int not NULL,

C\_ID int not NULL,

T\_Assignment float,

T\_Quiz float,

T\_Exam float,

Percentage float,

C\_Grade char check ( C\_grade in('A','B','C','D','E','F',NULL)),

CONSTRAINT pk\_Takes

PRIMARY KEY (S\_CMS, C\_ID),

CONSTRAINT fk1\_Takes

FOREIGN KEY (C\_ID) references Course (Course\_ID),

CONSTRAINT fk2\_Takes

FOREIGN KEY (S\_CMS) references Student(CMS)

);

CREATE TABLE Quiz

(

C\_ID int NOT NULL,

Quiz\_ID int not NULL,

Total\_Marks int,

Q\_Date Date,

Weightage float,

CONSTRAINT pk\_Quiz

PRIMARY KEY (Quiz\_ID),

CONSTRAINT fk1\_Quiz

FOREIGN KEY (C\_ID) references Course (Course\_ID)

);

INSERT INTO Quiz

VALUES (1,1,10,"2021-11-15",10),

(1,2,10,"2021-12-25",10),

(2,3,10,"2021-11-11",10),

(2,4,10,"2021-12-21",10),

(3,5,10,"2021-11-12",10),

(3,6,10,"2021-12-22",10),

(4,7,10,"2021-11-13",10),

(4,8,10,"2021-12-23",10),

(5,9,10,"2021-11-14",10),

(5,10,10,"2021-12-24",10),

(6,11,10,"2021-11-16",10),

(6,12,10,"2021-12-26",10),

(7,13,10,"2021-11-17",10),

(7,14,10,"2021-12-27",10),

(8,15,10,"2021-11-18",10),

(8,16,10,"2021-12-28",10),

(9,17,10,"2021-11-19",10),

(9,18,10,"2021-12-29",10),

(10,19,10,"2021-11-10",10),

(10,20,10,"2021-12-20",10);

CREATE TABLE Assignment

(

C\_ID int not NULL,

Assignment\_ID int not NULL,

Total\_Marks int,

Deadline Date,

Weightage float,

CONSTRAINT pk\_Assignment

PRIMARY KEY (Assignment\_ID),

CONSTRAINT fk1\_Assignment

FOREIGN KEY (C\_ID) references Course (Course\_ID)

);

INSERT INTO Assignment

VALUES

(1,1,10,"2021-09-15",10),

(1,2,10,"2021-10-25",10),

(2,3,10,"2021-09-11",10),

(2,4,10,"2021-10-21",10),

(3,5,10,"2021-09-12",10),

(3,6,10,"2021-10-22",10),

(4,7,10,"2021-09-13",10),

(4,8,10,"2021-10-23",10),

(5,9,10,"2021-09-14",10),

(5,10,10,"2021-10-24",10),

(6,11,10,"2021-09-16",10),

(6,12,10,"2021-10-26",10),

(7,13,10,"2021-09-17",10),

(7,14,10,"2021-10-27",10),

(8,15,10,"2021-09-18",10),

(8,16,10,"2021-10-28",10),

(9,17,10,"2021-09-19",10),

(9,18,10,"2021-10-29",10),

(10,19,10,"2021-09-10",10),

(10,20,10,"2021-10-20",10);

CREATE TABLE Exam(

C\_ID int not NULL,

Exam\_ID int not NULL,

Venue varchar(50) ,

Total\_Marks int,

E\_Date Date,

Weightage float,

CONSTRAINT pk\_Exam

PRIMARY KEY (Exam\_ID),

CONSTRAINT fk1\_Exam

FOREIGN KEY (C\_ID) references Course (Course\_ID)

);

INSERT INTO Exam

VALUES

(1,1,"NET Hall",10,"2021-11-08",10),

(1,2,"Seminar Hall",10,"2022-01-10",10),

(2,3,"CR-12 SEECS",10,"2021-11-12",10),

(2,4,"Computing Lab 10",10,"2022-01-14",10),

(3,5,"NET Hall",10,"2021-11-12",10),

(3,6,"Computing Lab 5",10,"2022-01-14",10),

(4,7,"Seminar Hall",10,"2021-11-09",10),

(4,8,"NET Hall",10,"2022-01-11",10),

(5,9,"Computing Lab 1",10,"2021-11-10",10),

(5,10,"Lecture Hall RIMMS",10,"2022-01-12",10),

(6,11,"CR-7 SEECS",10,"2021-11-11",10),

(6,12,"CR-8 SEECS",10,"2022-01-13",10),

(7,13,"CR-2 SEECS",10,"2021-11-10",10),

(7,14,"CR-2 NBS",10,"2022-01-12",10),

(8,15,"Seminar Hall",10,"2021-11-11",10),

(8,16,"NET Hall",10,"2022-01-13",10),

(9,17,"Linked Lists",10,"2021-11-08",10),

(9,18,"NET Hall",10,"2022-01-10",10),

(10,19,"NET Hall",10,"2021-11-09",10),

(10,20,"NET Hall",10,"2022-01-11",10);

CREATE TABLE Attempts\_Quiz

(

Q\_ID int not NULL,

S\_CMS int not NULL,

Q\_Status varchar(10),

Obtained\_Marks int,

Grade\_Point float,

CONSTRAINT pk\_AQ

PRIMARY KEY (Q\_ID,S\_CMS),

CONSTRAINT fk1\_AQ

FOREIGN KEY (S\_CMS) references Student(CMS),

CONSTRAINT fk2\_AQ

FOREIGN KEY (Q\_ID) references Quiz(Quiz\_ID)

);

CREATE TABLE Attempts\_Assignment

(

A\_ID int not NULL,

S\_CMS int not NULL,

A\_Status varchar(10),

Obtained\_Marks int ,

Grade\_Point float,

CONSTRAINT pk\_AA

PRIMARY KEY (A\_ID,S\_CMS),

CONSTRAINT fk1\_AA

FOREIGN KEY (S\_CMS) references Student(CMS),

CONSTRAINT fk2\_AA

FOREIGN KEY (A\_ID) references Assignment(Assignment\_ID)

);

CREATE TABLE Attempts\_Exam

(

E\_ID int not NULL,

S\_CMS int not NULL,

E\_Status varchar(10),

Obtained\_Marks int ,

Grade\_Point float,

CONSTRAINT pk\_AE

PRIMARY KEY (E\_ID,S\_CMS),

CONSTRAINT fk1\_AE

FOREIGN KEY (S\_CMS) references Student(CMS),

CONSTRAINT fk2\_AE

FOREIGN KEY (E\_ID) references Exam(Exam\_ID)

);

CREATE TABLE Result

(

S\_CMS int not NULL,

GPA float,

CONSTRAINT pk\_Result

PRIMARY KEY (S\_CMS),

CONSTRAINT fk1\_Result

FOREIGN KEY (S\_CMS) references Student(CMS)

);

Create Table CareerPlanner

(

S\_CMS int unique,

P\_ID int not NULL,

CONSTRAINT pk\_plans

PRIMARY KEY (P\_ID),

CONSTRAINT fk1\_plans

FOREIGN KEY (S\_CMS) references Student(CMS)

);

INSERT INTO careerplanner

VALUES (1,1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);

Create Table Internship

(

CP\_ID int not NULL,

Internship\_ID int not NULL,

I\_Status varchar(20),

Company varchar(60),

Time\_period varchar(20),

CONSTRAINT pk\_intern

PRIMARY KEY (Internship\_ID,CP\_ID),

CONSTRAINT fk1\_Internship

FOREIGN KEY (CP\_ID) references CareerPlanner (P\_ID)

);

INSERT INTO internship

VALUES (1,1,"In Progress","NESCOM","6 Months"),

(2 ,1,"Pending" , "MOL" ,"3 Months"),

(3 ,1, "In Progress" ,"KRL" ,"1 Years"),

(4 ,1, "Done" ,"Fauji Foundation", "3 Months"),

(1 ,2, "Done","Care", "2 Months"),

(3, 2, "Pending" ,"Fauji Foundation", "8 Months"),

(3, 3, Null ,"MCC", "2 Months");

Create Table Project

(

CP\_ID int not NULL,

Project\_ID int not NULL,

P\_Status varchar(20),

CompanyorInstitute varchar(60),

Time\_period varchar(20),

Title varchar(50),

CONSTRAINT pk\_project

PRIMARY KEY (Project\_ID,CP\_ID),

CONSTRAINT fk1\_Project

FOREIGN KEY (CP\_ID) references CareerPlanner (P\_ID)

);

Insert into Project VALUES

(1 ,1 ,"In Progress", "NUST" ,"15 Days" ,"Patient Care And Medicine Reminder System" ),

(2 ,1 ,"Pending" ,"NADRA" ,"3 Months" ,"Speed Calculator System"),

(3 ,1 ,"In Progress", "NUST" ,"1 Month" ,"Shopping System"),

(4 ,1 ,"Done" ,"FAST" ,"2 Weeks" ,"Music Light integerated System"),

(5 ,1 ,"In Progress" ,"NUST" ,"1 Month" ,"Mini Game"),

(1 ,2 ,"Done", "NESCOM" ,"2 Months" ,"Car Tracking System"),

(2 ,2 ,"Done" ,"NUST" ,"1 Month" ,"Banking System"),

(5 ,2 ,NULL ,"NUST" ,"2 Months" ,"Hospital Management System"),

(1 ,3 ,"Done" ,"NUST" ,"1 Week" ,"Discord Bot");

Create Table Job\_Application

(

CP\_ID int not null,

Application\_ID int not NULL,

J\_Status varchar(20),

Company varchar(60),

Deadline date,

Experience varchar(100),

CONSTRAINT pk\_job

PRIMARY KEY (Application\_ID,CP\_ID),

CONSTRAINT fk1\_Job

FOREIGN KEY (CP\_ID) references CareerPlanner (P\_ID)

);

CREATE TABLE Personal\_Activity

(

Activity\_ID int not NULL,

Act\_Name Varchar(30),

CONSTRAINT pk\_PersonalAct

PRIMARY KEY (Activity\_ID)

);

CREATE TABLE Performs

(

S\_CMS int not NULL,

A\_ID int not NULL,

A\_Status varchar(20),

Time\_Period varchar(20) ,

CONSTRAINT pk\_Tracks

PRIMARY KEY (A\_ID,S\_CMS),

CONSTRAINT fk1\_Tracks

FOREIGN KEY (S\_CMS) references Student (CMS),

CONSTRAINT fk2\_Tracks

FOREIGN KEY (A\_ID) references Personal\_Activity (Activity\_ID)

);

CREATE TABLE Habit

(

Habit\_ID int not NULL,

Habit\_Name Varchar(30),

CONSTRAINT pk\_habit

PRIMARY KEY (Habit\_ID)

);

CREATE TABLE Tracks

(

S\_CMS int not NULL,

H\_ID int not NULL,

Duration time,

Allocated\_Time time,

Time\_Remaining time,

CONSTRAINT pk\_Tracks

PRIMARY KEY (H\_ID,S\_CMS),

CONSTRAINT fk1\_Does

FOREIGN KEY (S\_CMS) references Student (CMS),

CONSTRAINT fk2\_Does

FOREIGN KEY (H\_ID) references Habit (Habit\_ID)

);

alter table Attempts\_Quiz add constraint QOM check (Obtained\_Marks <=(select Total\_Marks from Quiz q, Attempts\_Quiz a where q.Quiz\_ID = a.Q\_ID));

alter table Attempts\_Assignment add constraint AOM check (Obtained\_Marks <=(select Total\_Marks from Assignment aa, Attempts\_Assignment bb where aa.Assignment\_ID = bb.A\_ID));

alter table Attempts\_Exam add constraint EOM check (Obtained\_Marks <=(select Total\_Marks from Exam ee, Attempts\_Exam ff where ee.Exam\_ID = ff.E\_ID));