|  |
| --- |
|  |

**DDL AND INSERT STATEMENTS**

DROP TABLE Course\_Type;

DROP TABLE Room;

DROP TABLE Trainer;

DROP TABLE Allocated\_Student;

DROP TABLE Current\_Course;

DROP TABLE Allocation;

--Course\_Type Table--

CREATE TABLE Course\_Type (

Course\_Code VARCHAR(20) NOT NULL,

Course\_Title VARCHAR(200) NOT NULL,

Course\_Cost NUMERIC(20) NOT NULL,

CONSTRAINT Course\_Type\_Course\_Code\_PK PRIMARY KEY(Course\_Code));

--Room Table--

CREATE TABLE Room(

Room\_No VARCHAR(20) NOT NULL,

Room\_Type VARCHAR(70) NOT NULL,

CONSTRAINT Room\_Room\_No\_PK PRIMARY KEY(Room\_No)

);

--Trainer Table--

CREATE TABLE Trainer(

Trainer\_ID VARCHAR(20) NOT NULL,

Trainer\_Name VARCHAR(150) NOT NULL,

Course\_Title VARCHAR(300) NOT NULL,

CONSTRAINT Trainer\_Trainer\_ID\_PK PRIMARY KEY (Trainer\_ID)

);

--Allocated Student Table--

CREATE TABLE Allocated\_Student(

Student\_ID VARCHAR(20) NOT NULL,

Student\_Name VARCHAR(200) NOT NULL,

CONSTRAINT Allocated\_Student\_Student\_ID\_PK PRIMARY KEY(Student\_ID)

);

--Current\_Course Table--

CREATE TABLE Current\_Course(

Current\_Course\_ID VARCHAR(20) NOT NULL,

Course\_Code VARCHAR(20) NOT NULL,

Trainer\_ID VARCHAR(20) NOT NULL,

Room\_No VARCHAR(20) NOT NULL,

Start\_Date DATE NOT NULL,

End\_date DATE NOT NULL,

Period NUMERIC(20) NOT NULL,

Limit\_of\_Students NUMERIC(20) NOT NULL,

CONSTRAINT Current\_Course\_Current\_Course\_ID\_PK PRIMARY KEY (Current\_Course\_ID),

CONSTRAINT Current\_Course\_Course\_Code\_FK FOREIGN KEY (Course\_Code) REFERENCES Course\_Type(Course\_Code),

CONSTRAINT Current\_Course\_Trainer\_ID\_FK FOREIGN KEY (Trainer\_ID) REFERENCES Trainer(Trainer\_ID),

CONSTRAINT Current\_Course\_Room\_No\_FK FOREIGN KEY (Room\_No) REFERENCES Room(Room\_No)

);

--Allocation Table--

CREATE TABLE Allocation(

Allocation\_ID VARCHAR(20) NOT NULL,

Current\_Course\_ID VARCHAR(20) NOT NULL,

Student\_ID VARCHAR(20) NOT NULL,

CONSTRAINT Allocation\_Allocation\_ID\_PK PRIMARY KEY (Allocation\_ID),

CONSTRAINT Allocation\_Current\_Course\_ID\_FK FOREIGN KEY (Current\_Course\_ID) REFERENCES Current\_Course(Current\_Course\_ID),

CONSTRAINT Allocation\_Student\_ID\_FK FOREIGN KEY (Student\_ID) REFERENCES Allocated\_Student(Student\_ID)

);

--Insert Values into Course\_Type--

INSERT INTO Course\_Type VALUES ('PYT101','Introduction to Python','1500');

INSERT INTO Course\_Type VALUES ('ART101','Introduction to Artificial Intelligence','1500');

INSERT INTO Course\_Type VALUES ('BUS101','Introduction to Business Analysis','1500');

INSERT INTO Course\_Type VALUES ('MMO202','Mathematical Modelling In SAS','2000');

INSERT INTO Course\_Type VALUES ('ADA202','Advanced Data Analysis','2000');

INSERT INTO Course\_Type VALUES ('MAL202','Machine Learning','2000');

INSERT INTO Course\_Type VALUES ('EPR202','Essentials In Project Managment','2000');

INSERT INTO Course\_Type VALUES ('ACC202','Advanced Cloud Computing','2000');

INSERT INTO Course\_Type VALUES ('CYB101','Introduction to Cybersecurity','1500');

INSERT INTO Course\_Type VALUES ('DAM101','Data Mining','2000');

--Insert Values into Room Table--

INSERT INTO Room VALUES ('R202','COMPUTING LAB');

INSERT INTO Room VALUES ('R204','CONFERENCE HALL');

INSERT INTO Room VALUES ('R206','LECTURE HALL B');

INSERT INTO Room VALUES ('R208','MEETING ROOM 2');

INSERT INTO Room VALUES ('R210','LECTURE HALL A');

INSERT INTO Room VALUES ('R212','MEETING ROOM 1');

--Insert Values into Trainer Table--

INSERT INTO Trainer VALUES ('T001', 'JOEL HENS','Introduction to Python');

INSERT INTO Trainer VALUES ('T002', 'BRADLEY OWENS','Introduction to Artificial Intelligence');

INSERT INTO Trainer VALUES ('T003', 'NEILS BOHR','Introduction to Business Analysis');

INSERT INTO Trainer VALUES ('T004', 'AMANDA NOEL','Mathematical Modelling In SAS');

INSERT INTO Trainer VALUES ('T005', 'ZAC ORGI', 'Advanced Data Analysis');

INSERT INTO Trainer VALUES ('T006', 'EVANS FRANK','Machine Learning');

INSERT INTO Trainer VALUES ('T007', 'JAMES GARDNER','Essentials In Project Management');

INSERT INTO Trainer VALUES ('T008', 'LOI HUI','Advanced Cloud Computing');

INSERT INTO Trainer VALUES ('T009', 'KIGAN HENRY','Introduction to Cybersecurity');

INSERT INTO Trainer VALUES ('T010', 'JOSEPH HILLS','Data Minig');

--Insert Values into Allocated\_Student Table--

INSERT INTO Allocated\_Student VALUES ('103122','Amanda Giza');

INSERT INTO Allocated\_Student VALUES ('103124','Lola Klitz');

INSERT INTO Allocated\_Student VALUES ('103126','Milli Mall');

INSERT INTO Allocated\_Student VALUES ('103128','Derrick Oppong');

INSERT INTO Allocated\_Student VALUES ('103130','John Bull');

INSERT INTO Allocated\_Student VALUES ('103132','Han Marfo');

INSERT INTO Allocated\_Student VALUES ('103134','Milli Mall');

INSERT INTO Allocated\_Student VALUES ('103136','Burnli Mills');

INSERT INTO Allocated\_Student VALUES ('103138','Baah Eben');

INSERT INTO Allocated\_Student VALUES ('103140','Ebert Right');

INSERT INTO Allocated\_Student VALUES ('103150','Antwi Danso');

INSERT INTO Allocated\_Student VALUES ('103152','Hyda Sam');

INSERT INTO Allocated\_Student VALUES ('103154','Afua Lily');

INSERT INTO Allocated\_Student VALUES ('103158','Anas Anthony');

INSERT INTO Allocated\_Student VALUES ('103144','Jonas Kyle');

INSERT INTO Allocated\_Student VALUES ('103160','Jimy Chow');

INSERT INTO Allocated\_Student VALUES ('103162', 'Levi Dankyi');

INSERT INTO Allocated\_Student VALUES ('103164','Nathan Kwow');

INSERT INTO Allocated\_Student VALUES ('103168','Eturu Prisca');

INSERT INTO Allocated\_Student VALUES ('103170','Beden Joe');

INSERT INTO Allocated\_Student VALUES ('103172','Loel Donda');

INSERT INTO Allocated\_Student VALUES ('103174','Megan Menu');

INSERT INTO Allocated\_Student VALUES ('103176','Muni Maya');

INSERT INTO Allocated\_Student VALUES ('103178','Abass Lekwi');

INSERT INTO Allocated\_Student VALUES ('103180','Sam Dugan');

INSERT INTO Allocated\_Student VALUES ('103182','Willie Hem');

INSERT INTO Allocated\_Student VALUES ('103184','Alan Altman');

INSERT INTO Allocated\_Student VALUES ('103186','Desire Wan');

INSERT INTO Allocated\_Student VALUES ('103188','Suit Mann');

INSERT INTO Allocated\_Student VALUES ('103190','Alex Rodd');

--Insert into Current\_Course\_Table--

INSERT INTO Current\_Course VALUES ('AC01','PYT101','T001','R202','10-Jun-23','10-Jun-23','5','10');

INSERT INTO Current\_Course VALUES ('AC02','ART101','T002','R204','10-Jun-23','13-Jun-23','3','20');

INSERT INTO Current\_Course VALUES ('AC03','BUS101','T003','R206','15-Jul-23','18-Jul-23','3','20');

INSERT INTO Current\_Course VALUES ('AC04','MMO202','T004','R208','20-Jun-23','25-Jun-23','5','20');

INSERT INTO Current\_Course VALUES ('AC05','ADA202','T005','R210','03-Aug-23','6-Aug-23','3','10');

INSERT INTO Current\_Course VALUES ('AC06','MAL202','T006','R212','30-Dec-23','2-Jan-24','3','10');

INSERT INTO Current\_Course VALUES ('AC07','EPR202','T007','R202','04-Apr-23','9-Apr-23','5','10');

INSERT INTO Current\_Course VALUES ('AC08','ACC202','T008','R204','10-Apr-23','15-Apr-23','5','20');

INSERT INTO Current\_Course VALUES ('AC09','CYB101','T009','R206','07-May-23','12-May-23','5','10');

INSERT INTO Current\_Course VALUES ('AC10','DAM101','T010','R208','02-Oct-23','5-Oct-24','3','10');

INSERT INTO Current\_Course VALUES ('AC11','PYT101','T001','R202','02-Feb-23','7-Feb-24','5','10');

INSERT INTO Current\_Course VALUES ('AC12','ART101','T002','R204','05-Mar-23','08-Mar-24','3','10');

INSERT INTO Current\_Course VALUES ('AC13','MMO202','T004','R208','08-Jan-23','11-Jan-24','3','10');

INSERT INTO Current\_Course VALUES ('AC14','MAL202','T006','R212','05-Feb-23','08-Feb-24','3','10');

INSERT INTO Current\_Course VALUES ('AC15','EPR202','T007','R202','04-Aug-24','09-Aug-24','5','20');

INSERT INTO Current\_Course VALUES ('AC16','ACC202','T008','R204','23-May-23','28-May-24','5','20');

INSERT INTO Current\_Course VALUES ('AC17','DAM101','T010','R208','17-Nov-23','22-Nov-24','5','10');

INSERT INTO Current\_Course VALUES ('AC18','BUS101','T003','R206','14-May-23','19-May-24','5','10');

--Insert into Allocation Table--

INSERT INTO Allocation VALUES ('AL0','AC01','103122');

INSERT INTO Allocation VALUES ('AL1','AC02','103122');

INSERT INTO Allocation VALUES ('AL2','AC01','103124');

INSERT INTO Allocation VALUES ('AL3','AC02','103124');

INSERT INTO Allocation VALUES ('AL4','AC03','103126');

INSERT INTO Allocation VALUES ('AL5','AC04','103126');

INSERT INTO Allocation VALUES ('AL6','AC03','103128');

INSERT INTO Allocation VALUES ('AL7','AC04','103128');

INSERT INTO Allocation VALUES ('AL8','AC05','103130');

INSERT INTO Allocation VALUES ('AL9','AC06','103130');

INSERT INTO Allocation VALUES ('AL10','AC05','103132');

INSERT INTO Allocation VALUES ('AL11','AC06','103132');

INSERT INTO Allocation VALUES ('AL12','AC07','103134');

INSERT INTO Allocation VALUES ('AL13','AC08','103134');

INSERT INTO Allocation VALUES ('AL14','AC07','103136');

INSERT INTO Allocation VALUES ('AL15','AC08','103136');

INSERT INTO Allocation VALUES ('AL16','AC09','103138');

INSERT INTO Allocation VALUES ('AL17','AC10','103138');

INSERT INTO Allocation VALUES ('AL18','AC11','103140');

INSERT INTO Allocation VALUES ('AL19','AC10','103140');

INSERT INTO Allocation VALUES ('AL20','AC11','103128');

INSERT INTO Allocation VALUES ('AL21','AC12','103144');

INSERT INTO Allocation VALUES ('AL22','AC13','103150');

INSERT INTO Allocation VALUES ('AL23','AC14','103154');

INSERT INTO Allocation VALUES ('AL24','AC15','103170');

INSERT INTO Allocation VALUES ('AL25','AC16','103158');

INSERT INTO Allocation VALUES ('AL26','AC17','103182');

INSERT INTO Allocation VALUES ('AL27','AC18','103164');

**QUESTION 1:**

SELECT Course\_Code, Start\_date

FROM Current\_Course

ORDER BY Start\_Date;

A screenshot of a phone

Description automatically generated

A screenshot of a phone

Description automatically generated

**QUESTION 2:**

Select \*

FROM Course\_Type

WHERE Course\_Title LIKE '%Introduction%';

A screen shot of a computer

Description automatically generated

**QUESTION 3:**

--Most Expensive--

SELECT \* FROM Course\_Type WHERE Course\_Cost = (SELECT MAX(Course\_Cost) FROM (Course\_Type));

A screen shot of a computer

Description automatically generated

---Least Expensive--

SELECT \* FROM Course\_Type WHERE Course\_Cost = (SELECT MIN(Course\_Cost) FROM (Course\_Type));

A screen shot of a computer

Description automatically generated

**QUESTION 4:**

SELECT a.Current\_Course\_ID as "Currently Allocated Courses",

ct.Course\_Title as "Course Title",

COUNT(\*)"Number of Students"

FROM Allocation a,Current\_Course cc,Course\_Type ct

WHERE a.Current\_Course\_ID = cc.Current\_Course\_ID

AND cc.Course\_Code = ct.Course\_Code

GROUP BY a.Current\_Course\_ID,cc.Course\_Code,ct.Course\_Title

ORDER BY a.Current\_Course\_ID;

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**QUESTION 5:**

SELECT cc.Course\_Code,ct.Course\_Title,cc.Start\_Date

FROM Current\_Course cc, Course\_Type ct

WHERE cc.Course\_Code = ct.Course\_Code AND Start\_Date BETWEEN '01-APR-23' AND '31-DEC-23';

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

**QUESTION 6:**

SELECT course\_Code as "courses Code", count(Current\_Course\_ID) as "Number of Current Courses Running"

FROM Current\_Course

WHERE start\_date BETWEEN '01-Jan-2023' AND '31-Dec-2023'

GROUP BY course\_Code;

A screenshot of a computer

Description automatically generated

**QUESTION 7:**

SELECT r.Room\_No, r.Room\_Type, t.Trainer\_Name as Trainer, c.Course\_Code, c.Start\_Date, c.End\_date

FROM Room r

LEFT JOIN Current\_Course c on r.Room\_No = c.Room\_No

LEFT JOIN Trainer t on t.Trainer\_ID = c.Trainer\_ID;

A table with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

**QUESTION 8:**

SELECT a.Student\_ID, a.Student\_Name, c.Course\_Code, c.Start\_Date

FROM Allocated\_Student a

LEFT JOIN Allocation b on a.Student\_ID = b.Student\_ID

LEFT JOIN Current\_Course c ON c.Current\_Course\_ID = b.Current\_Course\_ID

WHERE c.start\_date = '20-JUN-23';

A screenshot of a computer

Description automatically generated

**QUESTION 9:**

SELECT

cc.Course\_Code as "COURSE CODE",

c.Start\_Date as "Run Date",

cc.Course\_Cost,

Count(ca.Student\_ID) as "No of Students",

(cc.Course\_Cost \* Count(ca.Student\_ID)) as "Revenue"

FROM Course\_Type cc

LEFT JOIN Current\_Course c on cc.Course\_Code = c.Course\_Code

LEFT JOIN Allocation ca on c.Current\_Course\_ID = ca.Current\_Course\_ID

GROUP BY cc.Course\_Code, c.Start\_Date, cc.Course\_Cost;

A table with numbers and numbers

Description automatically generated

A screenshot of a computer

Description automatically generated