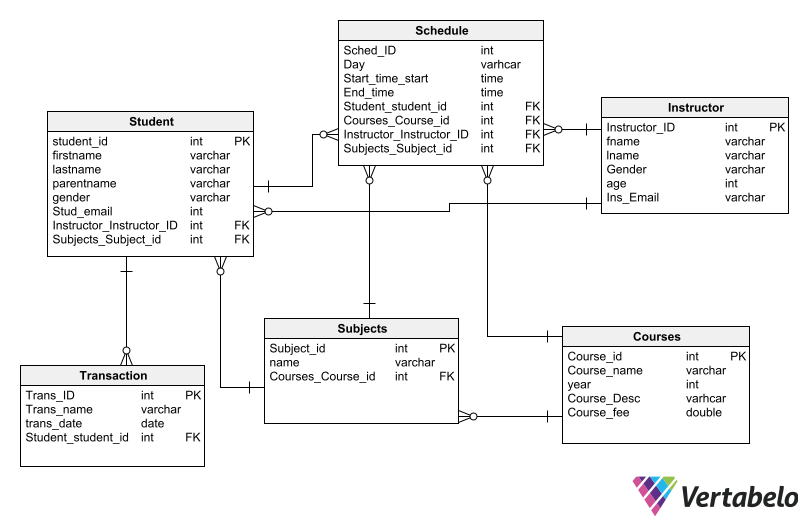
**School Management System**

A School Management System is an information management system for educational institutions to manage all student's data. School Management System allows complete control of the different activities that includes class, courses, subjects, etc.

**ERD Diagram:**

**Using vertabelo I have created a Relation Schema:**



**Tables and Data inserted:**

CREATE TABLE Courses (

Course\_id int NOT NULL DEFAULT 100,

Course\_name varchar NOT NULL DEFAULT 100,

year int NOT NULL,

Course\_Desc varhcar NOT NULL DEFAULT 100,

Course\_fee double NOT NULL,

CONSTRAINT Courses\_pk PRIMARY KEY (Course\_id)

);

-- Table: Instructor

CREATE TABLE Instructor (

Instructor\_ID int NOT NULL,

fname varchar NOT NULL,

lname varchar NOT NULL,

Gender varchar NOT NULL,

age int NOT NULL,

Ins\_Email varchar NOT NULL,

CONSTRAINT Instructor\_pk PRIMARY KEY (Instructor\_ID)

);

-- Table: Schedule

CREATE TABLE Schedule (

Sched\_ID int NOT NULL,

Day varhcar NOT NULL,

Start\_time\_start time NOT NULL,

End\_time time NOT NULL,

Student\_student\_id int NOT NULL,

Courses\_Course\_id int NOT NULL,

Instructor\_Instructor\_ID int NOT NULL,

Subjects\_Subject\_id int NOT NULL

);

-- Table: Student

CREATE TABLE Student (

student\_id int NOT NULL,

firstname varchar NOT NULL,

lastname varchar NOT NULL,

parentname varchar NOT NULL,

gender varchar NOT NULL,

Stud\_email int NOT NULL,

Instructor\_Instructor\_ID int NOT NULL,

Subjects\_Subject\_id int NOT NULL,

CONSTRAINT Student\_pk PRIMARY KEY (student\_id)

);

-- Table: Subjects

CREATE TABLE Subjects (

Subject\_id int NOT NULL,

name varchar NOT NULL,

Courses\_Course\_id int NOT NULL,

CONSTRAINT Subjects\_pk PRIMARY KEY (Subject\_id)

);

-- Table: Transaction

CREATE TABLE Transaction (

Trans\_ID int NOT NULL,

Trans\_name varchar NOT NULL,

trans\_date date NOT NULL,

Student\_student\_id int NOT NULL,

CONSTRAINT Transaction\_pk PRIMARY KEY (Trans\_ID)

);

-- foreign keys

-- Reference: Schedule\_Courses (table: Schedule)

ALTER TABLE Schedule ADD CONSTRAINT Schedule\_Courses FOREIGN KEY Schedule\_Courses (Courses\_Course\_id)

REFERENCES Courses (Course\_id);

-- Reference: Schedule\_Instructor (table: Schedule)

ALTER TABLE Schedule ADD CONSTRAINT Schedule\_Instructor FOREIGN KEY Schedule\_Instructor (Instructor\_Instructor\_ID)

REFERENCES Instructor (Instructor\_ID);

-- Reference: Schedule\_Student (table: Schedule)

ALTER TABLE Schedule ADD CONSTRAINT Schedule\_Student FOREIGN KEY Schedule\_Student (Student\_student\_id)

REFERENCES Student (student\_id);

-- Reference: Schedule\_Subjects (table: Schedule)

ALTER TABLE Schedule ADD CONSTRAINT Schedule\_Subjects FOREIGN KEY Schedule\_Subjects (Subjects\_Subject\_id)

REFERENCES Subjects (Subject\_id);

-- Reference: Student\_Instructor (table: Student)

ALTER TABLE Student ADD CONSTRAINT Student\_Instructor FOREIGN KEY Student\_Instructor (Instructor\_Instructor\_ID)

REFERENCES Instructor (Instructor\_ID);

-- Reference: Student\_Subjects (table: Student)

ALTER TABLE Student ADD CONSTRAINT Student\_Subjects FOREIGN KEY Student\_Subjects (Subjects\_Subject\_id)

REFERENCES Subjects (Subject\_id);

-- Reference: Subjects\_Courses (table: Subjects)

ALTER TABLE Subjects ADD CONSTRAINT Subjects\_Courses FOREIGN KEY Subjects\_Courses (Courses\_Course\_id)

REFERENCES Courses (Course\_id);

-- Reference: Transaction\_Student (table: Transaction)

ALTER TABLE Transaction ADD CONSTRAINT Transaction\_Student FOREIGN KEY Transaction\_Student (Student\_student\_id)

REFERENCES Student (student\_id);

**Data Inserted into Tables:**

**Courses Data**

insert into courses values(1,"Cloud",2021,"Related to AWS",1300);

insert into courses values(2,"Database",2021,"Related to Database Management System",1000);

insert into courses values(3,"DataScience",2021,"Related to Data Analysis",1500);

**Instructor Data:**

insert into instructor values(11,"George","Kal","male",40,"george@gmail.com");

insert into instructor values(12,"maali","joe","male",45,"mallijoe@gmail.com");

insert into instructor values(13,"karry","james","male",50,"james@gmail.com");

**Schedule Data:**

insert into schedule values(1,"Monday", '12:00:00', '13:00:00',1,1,11,1);

insert into schedule values(2,"Tuesday", '12:00:00', '13:00:00',1,2,13,3);

insert into schedule values(3,"Wednesday", '12:00:00', '13:00:00',1,3,12,2);

insert into schedule values(1,"Thursday", '12:00:00', '13:00:00',1,1,11,1);

insert into schedule values(1,"Friday", '12:00:00', '13:00:00',1,1,11,1);

**Student Data**

insert into student values(1,"saikrishna","dubak","Shiva","male","saikrishna@gmail.com",11,1);

insert into student values(2,"teja","Ravi","Notani","male","ravi@gmail.com",12,3);

insert into student values(3,"Manoj","Jakana","Natsha","male","manoj@gmail.com",11,2);

insert into student values(4,"Ramya","Lekanav","Hari","female","Ramya@gmail.com",13,3);

**Subjects Data:**

insert into subjects values(1,"Cloud Computing",1);

insert into subjects values(2,"AWS",1);

insert into subjects values(3,"Data Analysis",3);

insert into subjects values(4,"DBMS",2);

**Transaction Data:**

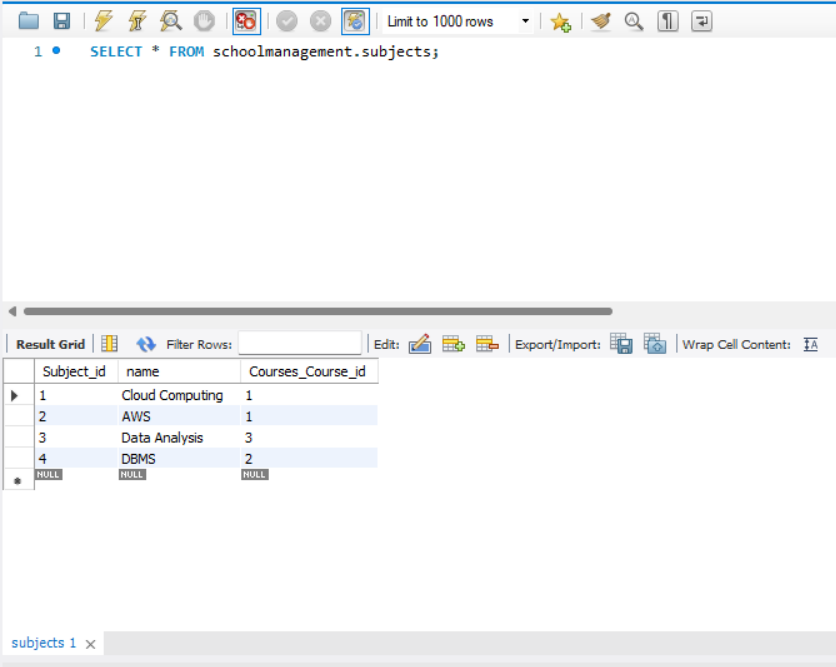
insert into transaction values(1,"Course\_fee",17/12/2021,1);

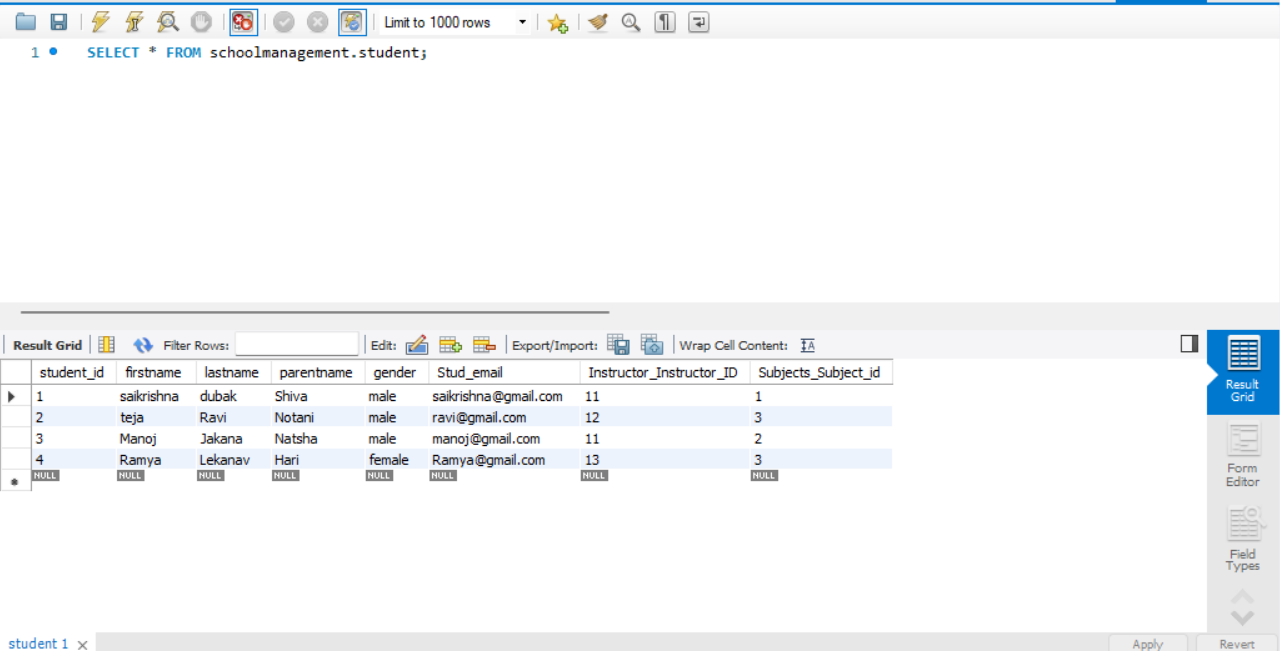
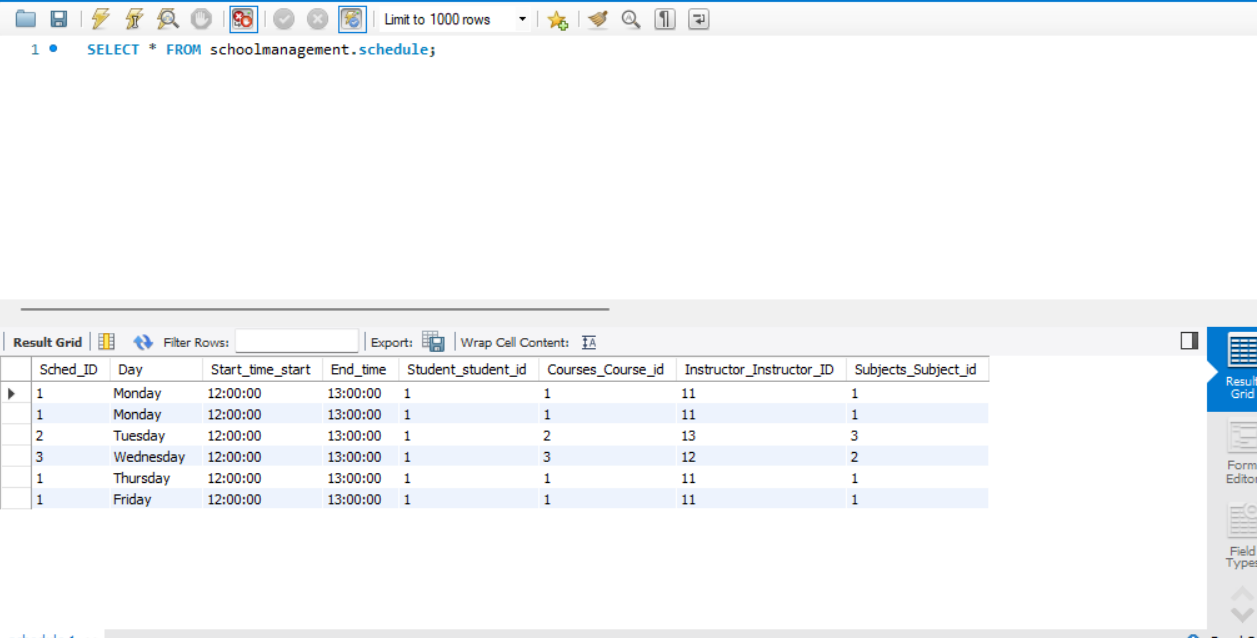
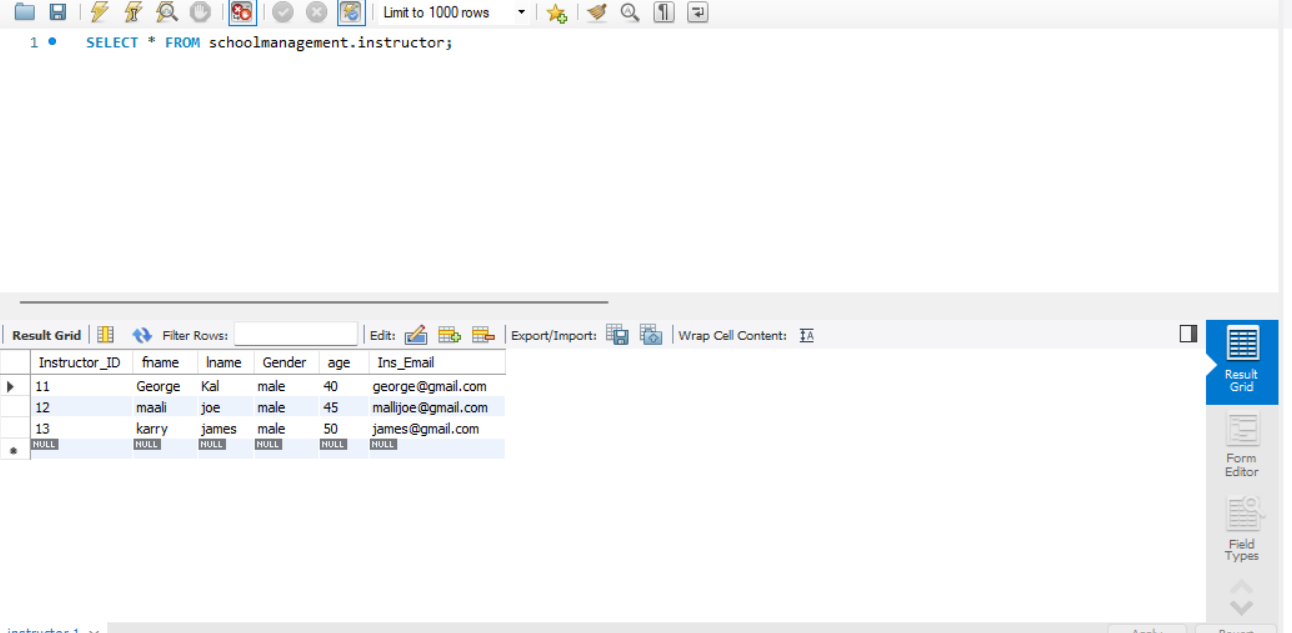
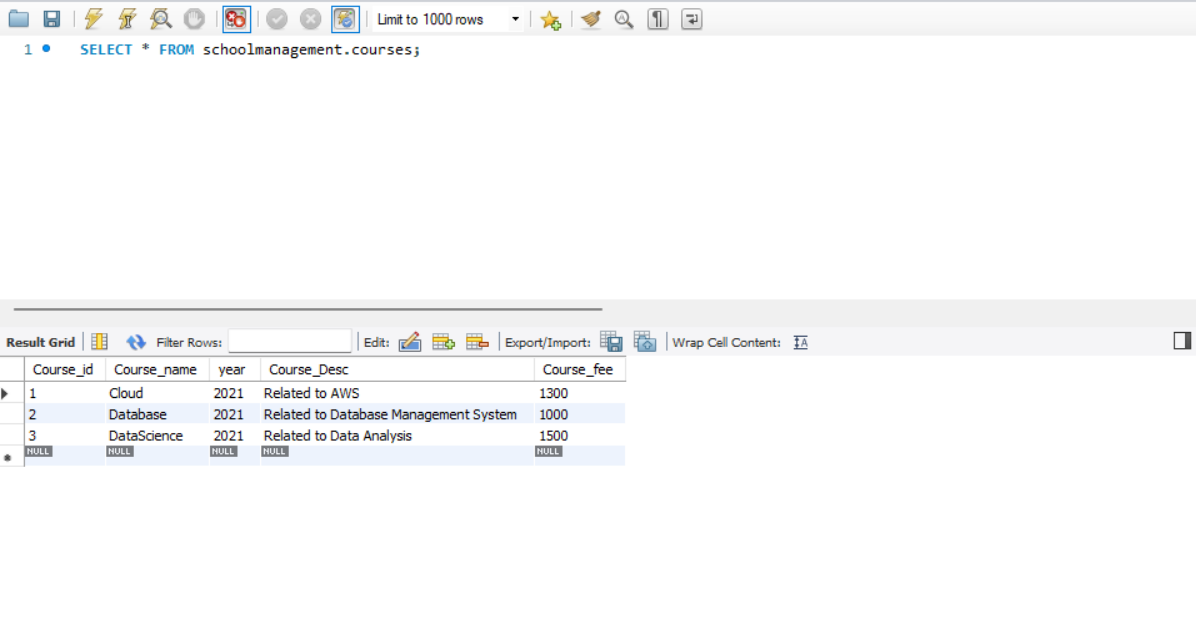
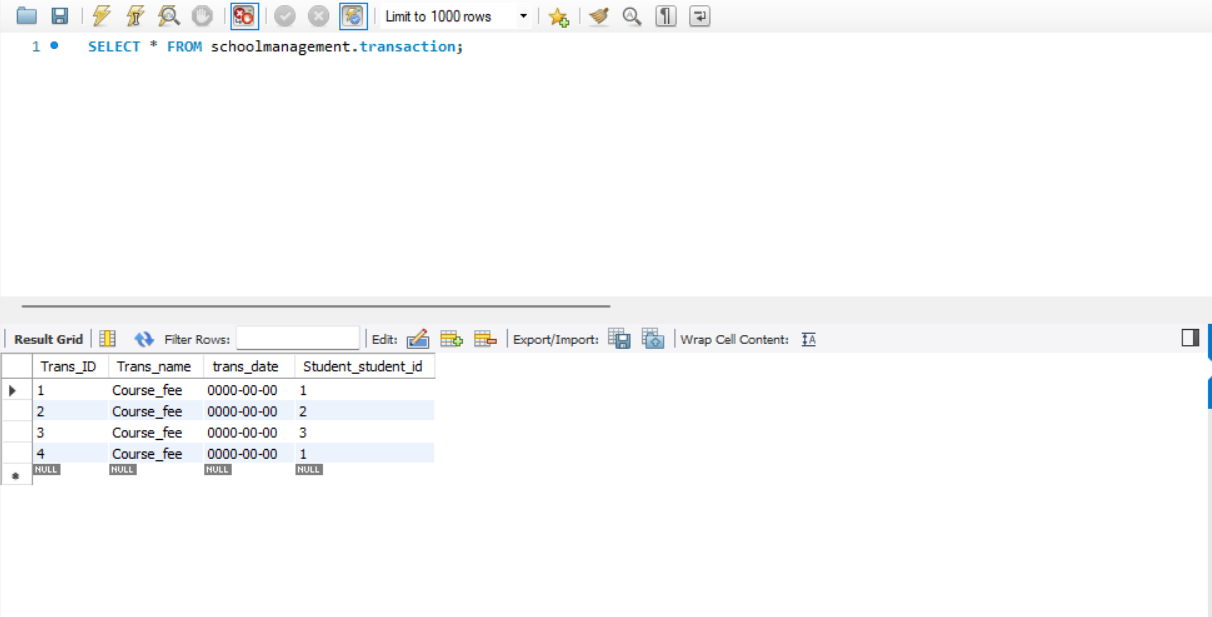
insert into transaction values(2,"Course\_fee",15/08/2021,2);

insert into transaction values(3,"Course\_fee",07/04/2021,3);

insert into transaction values(4,"Course\_fee",13/12/2021,1);

**Screen Shots of All Tables and Data:**

****

****

**Python Programming:**

import mysql.connector

from mysql.connector import errorcode

# Connect to the MySQL database

try:

   mydb = mysql.connector.connect(

      user="root",

      password="root",

      host="localhost",

      database="schoolmanagement")

except mysql.connector.Error as err:

   if err.errno == errorcode.ER\_ACCESS\_DENIED\_ERROR:

      print("Invalid credentials")

   elif err.errno == errorcode.ER\_BAD\_DB\_ERROR:

      print("Database not found")

   else:

      print("Cannot connect to database:", err)

else:

    cursor = mydb.cursor()

# Retreive all tables Infomration using the below code

    q1 = ("SELECT \* FROM student")

    cursor.execute(q1)

    print("------Student Table-------\n")

    for row in cursor.fetchall():

        print(row,"\n")

    q2 = ("SELECT \* FROM subjects")

    cursor.execute(q2)

    print("------Subjects Table ------------\n")

    for row in cursor.fetchall():

        print(row,"\n")

    q3 = ("SELECT \* FROM courses")

    cursor.execute(q3)

    print("-----Courses Table--------\n")

    for row in cursor.fetchall():

        print(row,"\n")

    q4 = ("SELECT \* FROM schedule")

    cursor.execute(q4)

    print("-----Schedule Table---------\n")

    for row in cursor.fetchall():

        print(row,"\n")

    q5 = ("SELECT \* FROM transaction")

    cursor.execute(q5)

    print("-----insurance Table---------\n")

    for row in cursor.fetchall():

        print(row,"\n")

    q6 = ("SELECT \* FROM instructor")

    cursor.execute(q6)

    print("-----Instructor  Table--------\n")

    for row in cursor.fetchall():

        print(row,"\n")

**Output:**

****