# **EdSt.Functions\_DB\_setup**

**Code:**

**import** mysql.connector **as** sql  
  
  
**def** connect(host: str = **'localhost'**, user: str = **'root'**, pswrd: str = **'99%Ch@c@late'**, database: str = **'EdStudio'**):  
 **if** database == **''**:  
 con = sql.connect(host=host,  
 user=user,  
 password=pswrd)  
 **else**:  
 con = sql.connect(host=host,  
 user=user,  
 password=pswrd,  
 database=database)  
 **if not** con.is\_connected():  
 print(**"Connection Error!"**)  
 exit(0)  
 **else**:  
 **return** con  
  
  
**def** createDatabase(DBname: str, pswrd: str):  
 con = connect(database=**''**, pswrd=pswrd)  
 cur = con.cursor()  
 q\_1 = **'DROP DATABASE IF EXISTS {}'**.format(DBname)  
 cur.execute(q\_1)  
 q2 = **'CREATE DATABASE IF NOT EXISTS {}'**.format(DBname)  
 cur.execute(q2)  
 con.commit()  
 con.close()  
  
  
**def** createParentTable(TbName: str, pswrd: str, Fields\_Datatype\_Constraints: tuple):  
 con = connect(pswrd=pswrd)  
 cur = con.cursor()  
  
 q\_1 = **'CREATE TABLE {}({})'**.format(TbName, Fields\_Datatype\_Constraints[0])  
 cur.execute(q\_1)  
  
 **for** i **in** Fields\_Datatype\_Constraints[1:len(Fields\_Datatype\_Constraints)]:  
 q\_2 = **'ALTER TABLE {} ADD COLUMN {}'**.format(TbName, i)  
 cur.execute(q\_2)  
  
 con.commit()  
 con.close()  
  
  
**def** createChildTable(TbName: str, pswrd: str, Field\_Datatype\_Constraints: tuple, ForeignKey\_Requirements: tuple):  
 con = connect(pswrd=pswrd)  
 cur = con.cursor()  
 q = **'CREATE TABLE {}({}'**.format(TbName, Field\_Datatype\_Constraints[0])  
 **for** column **in** Field\_Datatype\_Constraints[1:]:  
 q = q + **', {}'**.format(column)  
 **try**:  
 **for** f\_key **in** ForeignKey\_Requirements:  
 q = q + **', FOREIGN KEY({})**

**REFERENCES {}({}) ON DELETE CASCADE**

**ON UPDATE CASCADE'**.format(f\_key[0], f\_key[1], f\_key[2])  
 q = q + **')'** q = str(q)  
 cur.execute(q)  
 **finally**:  
 con.commit()  
 con.close()  
  
  
**def** InsertTbValues(tbname: str, pswrd: str, records: tuple):  
 counter = 1  
 con = connect(pswrd=pswrd)  
 cur = con.cursor()  
  
 **if** len(records) == 1:  
 q1 = **'INSERT INTO {} VALUES('**.format(tbname)  
 **for** i **in** range(len(records)):  
 **for** j **in** range(len(records[i])):  
 **if** i == 0:  
 **if** records[i][j] == **'NULL'**:  
 q1 = q1 + **'NULL'  
 elif** records[i][j] == **'default'**:  
 q1 = q1 + **'default'  
 elif** type(records[i][j]) == str:  
 q1 = q1 + **"'{}'"**.format(records[i][j])  
 **elif** type(records[i][j])==int **or** type(records[i][j])==float:  
 q1 = q1 + **'{}'**.format(records[i][j])  
 **else**:  
 **if** records[i][j] == **'NULL'**:  
 q1 = q1 + **',NULL'  
 elif** records[i][j] == **'default'**:  
 q1 = q1 + **',default'  
 elif** type(records[i][j]) == str:  
 q1 = q1 + **",'{}'"**.format(records[i][j])  
 **elif** type(records[i][j])==int **or** type(records[i][j])==float:  
 q1 = q1 + **',{}'**.format(records[i][j])  
 q1 = q1 + **')'  
  
 if** len(records) > 1:  
 q1 = **'INSERT INTO {} VALUES('**.format(tbname)  
 **for** i **in** range(len(records)):  
 **for** j **in** range(len(records[i])):  
 **if** j == 0:  
 **if** records[i][j] == **'NULL'**:  
 q1 = q1 + **'NULL'  
 elif** records[i][j] == **'default'**:  
 q1 = q1 + **'default'  
 elif** type(records[i][j]) == str:  
 q1 = q1 + **"'{}'"**.format(records[i][j])  
 **elif** type(records[i][j]) == int **or** type(records[i][j]) == float:  
 q1 = q1 + **'{}'**.format(records[i][j])  
 **else**:  
 **if** records[i][j] == **'NULL'**:  
 q1 = q1 + **',NULL'  
 elif** records[i][j] == **'default'**:  
 q1 = q1 + **',default'  
 elif** type(records[i][j]) == str:  
 q1 = q1 + **",'{}'"**.format(records[i][j])  
 **elif** type(records[i][j]) == int **or** type(records[i][j]) == float:  
 q1 = q1 + **',{}'**.format(records[i][j])  
 counter += 1  
 **if** counter <= len(records):  
 q1 = q1 + **'),('** q1 = q1 + **')'** cur.execute(q1)  
 con.commit()  
 con.close()

# **EdSt.main\_DB\_setup**

**Code:**

**import** Functions\_DB\_setup **as** db\_s  
  
pswrd = input(**"Enter the database password: "**)  
  
*# Creating EdStudio Database*db\_s.createDatabase(**'EdStudio'**, pswrd)  
  
*# Creating Admin Table*db\_s.createParentTable(**'Admin'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Password varchar(20) default("admin")'**))  
  
*# Creating Student Table*db\_s.createParentTable(**'Student'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Name varchar(80) not null'**,  
 **'Photo blob default("https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1")'**,  
 **'Gender char(1)'**,  
 **'DateOfBirth date not null'**,  
 **'BloodGroup char(3)'**,  
 **'PhoneNo varchar(10) not null'**,  
 **'Address blob not null'**,  
 **'DateOfAdmission date not null'**,  
 **'Password varchar(20) default("edstudio")'**))  
  
*# Creating Guardian Table*db\_s.createParentTable(**'Guardian'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Name varchar(80) not null'**,  
 **'Photo blob default("https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1")'**,  
 **'PhoneNo varchar(10) not null unique'**,  
 **'Address blob not null'**,  
 **'EmailAddress varchar(80) unique'**))  
  
*# Creating StudentGuardianRealtionship Table*db\_s.createParentTable(**'Relationship'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Type varchar(45) not null unique'**))  
  
*# Creatiing Teacher Table*db\_s.createParentTable(**'Teacher'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Name varchar(80) not null'**,  
 **'Photo blob default("https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1")'**,  
 **'PhoneNo varchar(10) not null'**,  
 **'DateOfBirth date not null'**,  
 **"Coordinator char(1) not null default('F')"**,  
 **'EdQualifications varchar(255)'**,  
 **"Status char(8) not null default('EMPLOYEE')"**,  
 **'Password varchar(20) default(NULL)'**))  
  
*# Creating Subject Table*db\_s.createParentTable(**'Subject'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'SubName varchar(25) not null unique'**))  
  
*# Creating Grade Table*db\_s.createParentTable(**'Grade'**, pswrd,  
 (**'Id int not null unique primary key'**,))  
  
*# Creating ExamGroup Table*db\_s.createParentTable(**'ExamGroup'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'Type varchar(60) not null unique'**))  
  
*# Creating Class Table*db\_s.createChildTable(**'Class'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'GradeId int not null'**,  
 **'Division char(1) not null'**),  
 ((**'GradeId'**, **'Grade'**, **'Id'**),))  
  
*# Creating Exam Table*db\_s.createChildTable(**'Exam'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'GroupId int not null'**,  
 **'SubjectId int not null'**,  
 **'GradeId int not null'**,  
 **'Date date not null'**,  
 **'Maxmark decimal(5,2) unsigned'**),  
 ((**'GroupId'**, **'ExamGroup'**, **'Id'**),  
 (**'SubjectId'**, **'Subject'**, **'Id'**),  
 (**'GradeId'**, **'Grade'**, **'Id'**)))  
  
*# Creating StudentGuardianRelationship Table*db\_s.createChildTable(**'StudGuardRelate'**, pswrd,  
 (**'StudentId int not null'**,  
 **'GuardianId int not null'**,  
 **'RelationshipId int'**),  
 ((**'StudentId'**, **'Student'**, **'Id'**),  
 (**'GuardianId'**, **'Guardian'**, **'Id'**),  
 (**'RelationshipId'**, **'Relationship'**, **'Id'**)))  
  
*# Creating TeacherSubject Table*db\_s.createChildTable(**'TeacherSubject'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'TeacherId int not null'**,  
 **'SubjectId int not null'**),  
 ((**'TeacherId'**, **'Teacher'**, **'Id'**),  
 (**'SubjectId'**, **'Subject'**, **'Id'**)))  
  
*# Creating StudentClass Table*db\_s.createChildTable(**'StudentClass'**, pswrd,  
 (**'StudentId int not null'**,  
 **'ClassId int not null'**),  
 ((**'StudentId'**, **'Student'**, **'Id'**),  
 (**'ClassId'**, **'Class'**, **'Id'**)))  
  
*# Creating TeacherSubjectClass Table*db\_s.createChildTable(**'TeacherSubjectClass'**, pswrd,  
 (**'TeacherSubjectId int not null'**,  
 **'ClassId int not null'**),  
 ((**'TeacherSubjectId'**, **'TeacherSubject'**, **'Id'**),  
 (**'ClassId'**, **'Class'**, **'Id'**)))  
  
*# Creating Marks Table*db\_s.createChildTable(**'Marks'**, pswrd,  
 (**'Id int unique not null primary key'**,  
 **'StudentId int not null'**,  
 **'ExamId int not null'**,  
 **'MarksScored decimal(5,2)'**),  
 ((**'StudentId'**, **'Student'**, **'Id'**),  
 (**'ExamId'**, **'Exam'**, **'Id'**)))  
  
*# Creating Assignment Table*db\_s.createChildTable(**'Assignment'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'TeacherId int not null'**,  
 **'GradeId int not null'**,  
 **'SubjectId int not null'**,  
 **'Topic varchar(45) not null'**,  
 **'StartDate date not null'**,  
 **'DueDate date not null'**,  
 **'Attachment blob'**),  
 ((**'TeacherId'**, **'Teacher'**, **'Id'**),  
 (**'SubjectId'**, **'Subject'**, **'Id'**)))  
  
*# Creating Announcement Table*db\_s.createChildTable(**'Announcement'**, pswrd,  
 (**'Id int not null unique primary key'**,  
 **'GradeId int not null'**,  
 **'TeacherId int not null'**,  
 **'AnnouncementDate date not null'**,  
 **'SubjectPreview varchar(255) default(null)'**,  
 **'Attachment blob default(null)'**),  
 ((**'TeacherId'**, **'Teacher'**, **'Id'**),  
 (**'GradeId'**, **'Grade'**, **'Id'**)))  
  
*# Admin Table Values*db\_s.InsertTbValues(**'Admin'**, pswrd,  
 ((10, **'admin'**),  
 (11, **'admin'**)))  
  
*# Student Table Values  
# 1-10*db\_s.InsertTbValues(**'Student'**, pswrd,  
 ((11001, **'SName01'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2004-01-01'**, **'O+'**, 9876543201, **'Address1'**, **'2018-01-01'**, **'edstudio'**),  
 (11002, **'SName02'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2005-01-01'**, **'O+'**, 9876543202, **'Address2'**, **'2019-01-01'**, **'edstudio'**),  
 (11003, **'SName03'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2006-01-01'**, **'O+'**, 9876543203, **'Address3'**, **'2020-01-01'**, **'edstudio'**),  
 (11004, **'SName04'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2007-01-01'**, **'O+'**, 9876543204, **'Address4'**, **'2021-01-01'**, **'edstudio'**),  
 (11005, **'SName05'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2008-01-01'**, **'O+'**, 9876543205, **'Address5'**, **'2022-01-01'**, **'edstudio'**),  
 (11006, **'SName06'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2004-01-01'**, **'O+'**, 9876543206, **'Address6'**, **'2018-01-01'**, **'edstudio'**),  
 (11007, **'SName07'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2005-01-01'**, **'O+'**, 9876543207, **'Address7'**, **'2019-01-01'**, **'edstudio'**),  
 (11008, **'SName08'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2006-01-01'**, **'O+'**, 9876543208, **'Address8'**, **'2020-01-01'**, **'edstudio'**),  
 (11009, **'SName09'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2007-01-01'**, **'O+'**, 9876543209, **'Address9'**, **'2021-01-01'**, **'edstudio'**),  
 (11010, **'SName10'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2008-01-01'**, **'O+'**, 9876543210, **'Address10'**, **'2022-01-01'**, **'edstudio'**)))  
*# 11-20*db\_s.InsertTbValues(**'Student'**, pswrd,  
 ((11011, **'SName11'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2004-01-01'**, **'O+'**, 9876543211, **'Address11'**, **'2018-01-01'**, **'edstudio'**),  
 (11012, **'SName12'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2005-01-01'**, **'O+'**, 9876543212, **'Address12'**, **'2019-01-01'**, **'edstudio'**),  
 (11013, **'SName13'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2006-01-01'**, **'O+'**, 9876543213, **'Address13'**, **'2020-01-01'**, **'edstudio'**),  
 (11014, **'SName14'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2007-01-01'**, **'O+'**, 9876543214, **'Address14'**, **'2021-01-01'**, **'edstudio'**),  
 (11015, **'SName15'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2008-01-01'**, **'O+'**, 9876543215, **'Address15'**, **'2022-01-01'**, **'edstudio'**),  
 (11016, **'SName16'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2004-01-01'**, **'O+'**, 9876543216, **'Address16'**, **'2018-01-01'**, **'edstudio'**),  
 (11017, **'SName17'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2005-01-01'**, **'O+'**, 9876543217, **'Address17'**, **'2019-01-01'**, **'edstudio'**),  
 (11018, **'SName18'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2006-01-01'**, **'O+'**, 9876543218, **'Address18'**, **'2020-01-01'**, **'edstudio'**),  
 (11019, **'SName19'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2007-01-01'**, **'O+'**, 9876543219, **'Address19'**, **'2021-01-01'**, **'edstudio'**),  
 (11020, **'SName20'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2008-01-01'**, **'O+'**, 9876543220, **'Address20'**, **'2022-01-01'**, **'edstudio'**)))  
*# 21-30*db\_s.InsertTbValues(**'Student'**, pswrd,  
 ((11021, **'SName21'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2004-01-01'**, **'O+'**, 9876543221, **'Address21'**, **'2018-01-01'**, **'edstudio'**),  
 (11022, **'SName22'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2005-01-01'**, **'O+'**, 9876543222, **'Address22'**, **'2019-01-01'**, **'edstudio'**),  
 (11023, **'SName23'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2006-01-01'**, **'O+'**, 9876543223, **'Address23'**, **'2020-01-01'**, **'edstudio'**),  
 (11024, **'SName24'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2007-01-01'**, **'O+'**, 9876543224, **'Address24'**, **'2021-01-01'**, **'edstudio'**),  
 (11025, **'SName25'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2008-01-01'**, **'O+'**, 9876543225, **'Address25'**, **'2022-01-01'**, **'edstudio'**),  
 (11026, **'SName26'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2004-01-01'**, **'O+'**, 9876543226, **'Address26'**, **'2018-01-01'**, **'edstudio'**),  
 (11027, **'SName27'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2005-01-01'**, **'O+'**, 9876543227, **'Address27'**, **'2019-01-01'**, **'edstudio'**),  
 (11028, **'SName28'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2006-01-01'**, **'O+'**, 9876543228, **'Address28'**, **'2020-01-01'**, **'edstudio'**),  
 (11029, **'SName29'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2007-01-01'**, **'O+'**, 9876543229, **'Address29'**, **'2021-01-01'**, **'edstudio'**),  
 (11030, **'SName30'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2008-01-01'**, **'O+'**, 9876543230, **'Address30'**, **'2022-01-01'**, **'edstudio'**)))  
*# 31-40*db\_s.InsertTbValues(**'Student'**, pswrd,  
 ((11031, **'SName31'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2004-01-01'**, **'O+'**, 9876543231, **'Address31'**, **'2018-01-01'**, **'edstudio'**),  
 (11032, **'SName32'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2005-01-01'**, **'O+'**, 9876543232, **'Address32'**, **'2019-01-01'**, **'edstudio'**),  
 (11033, **'SName33'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2006-01-01'**, **'O+'**, 9876543233, **'Address33'**, **'2020-01-01'**, **'edstudio'**),  
 (11034, **'SName34'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2007-01-01'**, **'O+'**, 9876543234, **'Address34'**, **'2021-01-01'**, **'edstudio'**),  
 (11035, **'SName35'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2008-01-01'**, **'O+'**, 9876543235, **'Address35'**, **'2022-01-01'**, **'edstudio'**),  
 (11036, **'SName36'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2004-01-01'**, **'O+'**, 9876543236, **'Address36'**, **'2018-01-01'**, **'edstudio'**),  
 (11037, **'SName37'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2005-01-01'**, **'O+'**, 9876543237, **'Address37'**, **'2019-01-01'**, **'edstudio'**),  
 (11038, **'SName38'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2006-01-01'**, **'O+'**, 9876543238, **'Address38'**, **'2020-01-01'**, **'edstudio'**),  
 (11039, **'SName39'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'F'**, **'2007-01-01'**, **'O+'**, 9876543239, **'Address39'**, **'2021-01-01'**, **'edstudio'**),  
 (11040, **'SName40'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 **'M'**, **'2008-01-01'**, **'O+'**, 9876543240, **'Address40'**, **'2022-01-01'**, **'edstudio'**)))  
  
  
*# Guardian Table Values  
# 1-10*db\_s.InsertTbValues(**'Guardian'**, pswrd,  
 ((10001, **'GName1'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450001, **'Address1'**, **'gname1@gmail.com'**),  
 (10002, **'GName2'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450002, **'Address2'**, **'gname2@gmail.com'**),  
 (10003, **'GName3'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450003, **'Address3'**, **'gname3@gmail.com'**),  
 (10004, **'GName4'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450004, **'Address4'**, **'gname4@gmail.com'**),  
 (10005, **'GName5'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450005, **'Address5'**, **'gname5@gmail.com'**),  
 (10006, **'GName6'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450006, **'Address6'**, **'gname6@gmail.com'**),  
 (10007, **'GName7'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450007, **'Address7'**, **'gname7@gmail.com'**),  
 (10008, **'GName8'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450008, **'Address8'**, **'gname8@gmail.com'**),  
 (10009, **'GName9'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450009, **'Address9'**, **'gname9@gmail.com'**),  
 (10010, **'GName10'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450010, **'Address10'**, **'gname10@gmail.com'**)))  
  
*# 11-20*db\_s.InsertTbValues(**'Guardian'**, pswrd,  
 ((10011, **'GName11'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450011, **'Address1'**, **'gname11@gmail.com'**),  
 (10012, **'GName12'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450012, **'Address2'**, **'gname12@gmail.com'**),  
 (10013, **'GName13'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450013, **'Address3'**, **'gname13@gmail.com'**),  
 (10014, **'GName14'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450014, **'Address4'**, **'gname14@gmail.com'**),  
 (10015, **'GName15'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450015, **'Address5'**, **'gname15@gmail.com'**),  
 (10016, **'GName16'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450016, **'Address6'**, **'gname16@gmail.com'**),  
 (10017, **'GName17'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450017, **'Address7'**, **'gname17@gmail.com'**),  
 (10018, **'GName18'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450018, **'Address8'**, **'gname18@gmail.com'**),  
 (10019, **'GName19'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450019, **'Address9'**, **'gname19@gmail.com'**),  
 (10020, **'GName20'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9123450020, **'Address10'**, **'gname20@gmail.com'**)))  
  
*# GuardianRelationship Table Values*db\_s.InsertTbValues(**'Relationship'**, pswrd,  
 ((10, **'Father'**),  
 (11, **'Mother'**),  
 (12, **'Sibling'**),  
 (13, **'Grand Parent'**),  
 (14, **'Guardian'**),  
 (15, **'Local Guardian'**),  
 (16, **'Relative'**),  
 (17, **'Other'**)))  
  
*# Teacher Table Values  
# 1-10*db\_s.InsertTbValues(**'Teacher'**, pswrd,  
 ((101, **'TName1'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450001, **'1992-01-01'**, **'F'**, **'B.Ed.'**, **'Employee'**, **'edstudio'**),  
 (102, **'TName2'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450002, **'1991-01-01'**, **'F'**, **'B.Ed., B.Sc.'**, **'Contract'**, **'edstudio'**),  
 (103, **'TName3'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450003, **'1990-01-01'**, **'F'**, **'B.Ed.'**, **'Contract'**, **'edstudio'**),  
 (104, **'TName4'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450004, **'1989-01-01'**, **'F'**, **'B.Ed., B.Sc.'**, **'Employee'**, **'edstudio'**),  
 (105, **'TName5'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450005, **'1988-01-01'**, **'F'**, **'B.Ed.'**, **'Contract'**, **'edstudio'**),  
 (106, **'TName6'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450006, **'1987-01-01'**, **'F'**, **'B.Ed., B.Sc.'**, **'Employee'**, **'edstudio'**),  
 (107, **'TName7'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450007, **'1986-01-01'**, **'T'**, **'B.Ed.'**, **'Contract'**, **'edstudio'**),  
 (108, **'TName8'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450008, **'1985-01-01'**, **'T'**, **'B.Ed., B.Sc.'**, **'Employee'**, **'edstudio'**),  
 (109, **'TName9'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450009, **'1984-01-01'**, **'T'**, **'B.Ed.'**, **'Contract'**, **'edstudio'**),  
 (110, **'TName10'**,  
 **'https://i0.wp.com/www.dc-hauswartungen.ch/wp-content/uploads/2018/01/dummy\_profile.png?ssl=1'**,  
 9213450010, **'1983-01-01'**, **'T'**, **'B.Ed., B.Sc.'**, **'Employee'**, **'edstudio'**),  
 (12, **'Admin'**,  
 **''**, 1000000000, **'1900-01-01'**, **'T'**, **'-'**, **'Employee'**, **'admin'**)))  
  
*# Subject Table Values*db\_s.InsertTbValues(**'Subject'**, pswrd,  
 ((1, **'Mathematics 1'**),  
 (2, **'Science'**),  
 (3, **'Social Science'**),  
 (4, **'English 1'**),  
 (5, **'Physics'**),  
 (6, **'Chemistry'**),  
 (7, **'Mathematics 2'**),  
 (8, **'English 2'**),  
 (9, **'Computer Science'**),  
 (10, **'Biology'**)))  
  
*# Grade Table Values*db\_s.InsertTbValues(**'Grade'**, pswrd,  
 ((9,),  
 (10,),  
 (11,),  
 (12,)))  
  
*# ExamGroup Table Values*db\_s.InsertTbValues(**'ExamGroup'**, pswrd,  
 ((100, **'Weekly Test'**),  
 (101, **'Unit Test 1'**),  
 (102, **'Unit Test 2'**),  
 (103, **'Quarterly Examination'**),  
 (104, **'Half-Yearly Examination'**),  
 (105, **'Annual Examination'**),  
 (106, **'Pre-Board 1'**),  
 (107, **'Pre-Board 2'**)))  
  
*# Class Table Values*db\_s.InsertTbValues(**'Class'**, pswrd,  
 ((1, 9, **'A'**),  
 (2, 9, **'B'**),  
 (3, 10, **'A'**),  
 (4, 10, **'B'**),  
 (5, 11, **'A'**),  
 (6, 11, **'B'**),  
 (7, 12, **'A'**),  
 (8, 12, **'B'**)))  
  
*# Exam Table Values*db\_s.InsertTbValues(**'Exam'**, pswrd,  
 ((101, 101, 5, 12, **'2022-01-01'**, 50),  
 (102, 101, 6, 12, **'2022-01-02'**, 50),  
 (103, 101, 7, 12, **'2022-01-03'**, 50),  
 (104, 101, 8, 12, **'2022-01-04'**, 50),  
 (105, 101, 9, 12, **'2022-01-05'**, 50),  
 (106, 101, 10, 12, **'2022-01-06'**, 50)))  
  
*# StudentGuardianRelationship Table Values*db\_s.InsertTbValues(**'StudGuardRelate'**, pswrd,  
 ((11001, 10001, 10),  
 (11002, 10001, 10),  
 (11003, 10002, 11),  
 (11004, 10002, 11),  
 (11005, 10003, 12),  
 (11006, 10003, 12),  
 (11007, 10004, 13),  
 (11008, 10004, 13),  
 (11009, 10005, 14),  
 (11010, 10005, 14),  
 (11011, 10006, 15),  
 (11012, 10006, 15),  
 (11013, 10007, 16),  
 (11014, 10007, 16),  
 (11015, 10008, 10),  
 (11016, 10008, 10),  
 (11017, 10009, 11),  
 (11018, 10009, 11),  
 (11019, 10010, 12),  
 (11020, 10010, 12),  
 (11021, 10011, 13),  
 (11022, 10011, 13),  
 (11023, 10012, 14),  
 (11024, 10012, 14),  
 (11025, 10013, 15),  
 (11026, 10013, 15),  
 (11027, 10014, 16),  
 (11028, 10014, 16),  
 (11029, 10015, 10),  
 (11030, 10015, 10),  
 (11031, 10016, 11),  
 (11032, 10016, 11),  
 (11033, 10017, 12),  
 (11034, 10017, 12),  
 (11035, 10018, 13),  
 (11036, 10018, 13),  
 (11037, 10019, 14),  
 (11038, 10019, 14),  
 (11039, 10020, 15),  
 (11040, 10020, 15)))  
  
*# TeacherSubject Table Values*db\_s.InsertTbValues(**'TeacherSubject'**, pswrd,  
 ((1, 101, 1),  
 (2, 102, 2),  
 (3, 103, 3),  
 (4, 104, 4),  
 (5, 105, 5),  
 (6, 106, 6),  
 (7, 107, 7),  
 (8, 108, 8),  
 (9, 109, 9),  
 (10, 110, 10),  
 (11, 104, 3),  
 (12, 12, 1),  
 (13, 12, 2),  
 (14, 12, 3),  
 (15, 12, 4),  
 (16, 12, 5),  
 (17, 12, 6),  
 (18, 12, 7),  
 (19, 12, 8),  
 (20, 12, 9),  
 (21, 12, 10)))  
  
*# StudentClass Table Values  
# Class 12 A & B*db\_s.InsertTbValues(**'StudentClass'**, pswrd,  
 ((11001, 7),  
 (11036, 7),  
 (11016, 7),  
 (11011, 7),  
 (11021, 7),  
 (11006, 8),  
 (11026, 8),  
 (11031, 8),  
 (11002, 8),  
 (11007, 8)))  
*# Class 11 A & B*db\_s.InsertTbValues(**'StudentClass'**, pswrd,  
 ((11012, 5),  
 (11017, 5),  
 (11022, 5),  
 (11027, 5),  
 (11032, 5),  
 (11037, 6),  
 (11003, 6),  
 (11008, 6),  
 (11013, 6),  
 (11018, 6)))  
*# Class 10 A & B*db\_s.InsertTbValues(**'StudentClass'**, pswrd,  
 ((11028, 3),  
 (11023, 3),  
 (11033, 3),  
 (11038, 3),  
 (11004, 3),  
 (11009, 4),  
 (11014, 4),  
 (11019, 4),  
 (11029, 4),  
 (11034, 4)))  
*# Class 9 A & B*db\_s.InsertTbValues(**'StudentClass'**, pswrd,  
 ((11024, 1),  
 (11039, 1),  
 (11005, 1),  
 (11010, 1),  
 (11015, 1),  
 (11020, 2),  
 (11025, 2),  
 (11030, 2),  
 (11035, 2),  
 (11040, 2)))  
  
*# TeacherSubjectClass Table Values*db\_s.InsertTbValues(**'TeacherSubjectClass'**, pswrd,  
 ((1, 1), (2, 1), (3, 1), (4, 1),  
 (1, 2), (2, 2), (3, 2), (4, 2),  
 (1, 3), (2, 3), (3, 3), (4, 3),  
 (1, 4), (2, 4), (3, 4), (4, 4),  
 (5, 5), (6, 5), (7, 5), (8, 5), (9, 5),  
 (5, 6), (6, 6), (7, 6), (8, 6), (10, 6),  
 (5, 7), (6, 7), (7, 7), (8, 7), (9, 7),  
 (5, 8), (6, 8), (7, 8), (8, 8), (10, 8),))  
  
*# Marks Table Values  
# For class 12 A*db\_s.InsertTbValues(**'Marks'**, pswrd,  
 ((11, 11001, 101, 26),  
 (12, 11001, 102, 27),  
 (13, 11001, 103, 28),  
 (14, 11001, 104, 29),  
 (15, 11001, 105, 30),  
 (16, 11036, 101, 31),  
 (17, 11036, 102, 32),  
 (18, 11036, 103, 33),  
 (19, 11036, 104, 34),  
 (20, 11036, 105, 35),  
 (21, 11016, 101, 36),  
 (22, 11016, 102, 37),  
 (23, 11016, 103, 38),  
 (24, 11016, 104, 39),  
 (25, 11016, 105, 40),  
 (26, 11011, 101, 41),  
 (27, 11011, 102, 42),  
 (28, 11011, 103, 43),  
 (29, 11011, 104, 44),  
 (30, 11011, 105, 45),  
 (31, 11021, 101, 46),  
 (32, 11021, 102, 47),  
 (33, 11021, 103, 48),  
 (34, 11021, 104, 49),  
 (35, 11021, 105, 50)))  
*# For class 12 B*db\_s.InsertTbValues(**'Marks'**, pswrd,  
 ((36, 11006, 101, 46),  
 (37, 11006, 102, 47),  
 (38, 11006, 103, 48),  
 (39, 11006, 104, 49),  
 (40, 11006, 106, 40),  
 (41, 11026, 101, 41),  
 (42, 11026, 102, 42),  
 (43, 11026, 103, 43),  
 (44, 11026, 104, 44),  
 (45, 11026, 106, 45),  
 (46, 11031, 101, 46),  
 (47, 11031, 102, 47),  
 (48, 11031, 103, 48),  
 (49, 11031, 104, 49),  
 (50, 11031, 106, 40),  
 (51, 11002, 101, 41),  
 (52, 11002, 102, 42),  
 (53, 11002, 103, 43),  
 (54, 11002, 104, 44),  
 (55, 11002, 106, 45),  
 (56, 11007, 101, 56),  
 (57, 11007, 102, 57),  
 (58, 11007, 103, 58),  
 (59, 11007, 104, 59),  
 (60, 11007, 106, 50),))  
  
*# Assignment Table Values*db\_s.InsertTbValues(**'Assignment'**, pswrd,  
 ((1, 101, 9, 1, **'Linear Equations WS'**, **'2022-07-01'**, **'2022-08-01'**,  
 **'https://data.templateroller.com/pdf\_docs\_html/167/1671/167137/algebra-2-wkst-3-5-3-7-linear-equations-in-slope-intercept-form-worksheet-with-answers\_print\_big.png'**),  
 (2, 106, 11, 6, **'Nomenclature of Organic Compounds WS'**, **'2022-07-02'**, **'2022-08-01'**,  
 **'https://i.pinimg.com/236x/9f/d9/2a/9fd92a1ac9d20a03a650fef71252d169--ap-chemistry-level-.jpg'**),  
 (3, 110, 12, 10, **'Principles of Inheritance and Variation WS'**, **'2022-07-03'**, **'2022-08-02'**,  
 **'https://i0.wp.com/www.worksheeto.com/postpic/2012/04/ap-biology-chapter-16-guided-reading-assignment-answers\_206593.png'**)))  
  
*# Announcement Table Values*db\_s.InsertTbValues(**'Announcement'**, pswrd,  
 ((1, 11, 109, **'2022-07-20'**,  
 **'A trek will be organised for all students of grade 11. \nInterested students may ask the CT for more details.'**,  
 **'NULL'**),  
 (2, 12, 110, **'2022-07-19'**,  
 **'A trek will be organised for all students of grade 12. \nInterested students may ask the CT for more details.'**,  
 **'NULL'**)))

# **EdSt.Functions\_CUI**

**Code:**

**from** EdSt.Functions\_DB\_setup **import** connect  
  
psword = **"99%Ch@c@late"***# SIGN IN***def** SignIn(userId: int, password):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 cur.execute(**"SELECT Id, Password FROM Student"**)  
 sID = cur.fetchall()  
 **if** (userId, password) **in** sID:  
 con.close()  
 **return "Student.valid"** cur.execute(**"SELECT Id, Password FROM Teacher"**)  
 tID = cur.fetchall()  
 **if** (userId, password) **in** tID:  
 con.close()  
 **return "Teacher.valid"** cur.execute(**"SELECT Id, Password FROM Admin"**)  
 aID = cur.fetchall()  
 **if** (userId, password) **in** aID:  
 con.close()  
 **return "Admin.valid"  
  
 if** ((userId, password) **not in** sID) **and** ((userId, password) **not in** tID):  
 con.close()  
 **return "Error!"** con.close()  
  
  
*# WELCOME BACK STUDENT***def** message\_hello\_student(StudentUser):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 cur.execute(**"SELECT name from student where id={}"**.format(StudentUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 name = i[0]  
 print(**"\nWelcome"**, name + **"!"**)  
 con.close()  
  
  
*# VIEW STUDENT PROFILE***def** view\_studentProfile\_student(StudentUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query = **'''SELECT student.id, student.name,grade.id,class.division,  
 student.photo,student.gender,  
 student.dateofbirth,guardian.name,student.bloodgroup,student.phoneno,student.address,  
 student.dateofadmission,guardian.name  
 from student   
 inner join studentclass on studentclass.studentid=student.id  
 inner join class on class.id=studentclass.classid  
 inner join grade on grade.id=class.gradeid  
 inner join studguardrelate on studguardrelate.studentid=student.id  
 inner join guardian on guardian.id=studguardrelate.guardianid  
 where student.id={}'''** cur.execute(query.format(StudentUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 print(**"\nAdmission ID"**, i[0], sep=**": "**)  
 print(**"Name"**, i[1], sep=**": "**)  
 print(**"Grade"**, i[2], sep=**": "**)  
 print(**"Division"**, i[3], sep=**": "**)  
 print(**"Photo"**, i[4], sep=**": "**)  
 print(**"Gender"**, i[5], sep=**": "**)  
 print(**"Date of Birth"**, i[6], sep=**": "**)  
 print(**"Guardian Name"**, i[7], sep=**": "**)  
 print(**"Blood group"**, i[8], sep=**": "**)  
 print(**"Phone no."**, i[9], sep=**": "**)  
 print(**"Address"**, i[10], sep=**": "**)  
 print(**"Date of admission"**, i[11], sep=**": "**, end=**"\n\n"**)  
 con.close()  
  
  
*# UPDATE USER PROFILE***def** edit\_profile\_all(UserType: str, UserId: int, Field: str, UpdatedValue):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **if** UserType == **'Student.valid'**:  
 qs = **'UPDATE Student SET {}="{}" WHERE Id={}'**.format(Field, UpdatedValue, UserId)  
 cur.execute(qs)  
 cur.execute(**'COMMIT'**)  
 cur.execute(**"SELECT \* FROM Student WHERE Id={}"**.format(UserId))  
 sData = cur.fetchone()  
 **return** sData  
 **elif** UserType == **'Teacher.valid'**:  
 qt = **'UPDATE Teacher Set {}="{}" WHERE Id={}'**.format(Field, UpdatedValue, UserId)  
 cur.execute(qt)  
 cur.execute(**'COMMIT'**)  
 cur.execute(**"SELECT \* FROM Teacher WHERE Id={}"**.format(UserId))  
 tData = cur.fetchone()  
 con.close()  
 **return** tData  
 **elif** UserType == **'Admin.valid'**:  
 qa = **'UPDATE Admin Set {}="{}" WHERE Id={}'**.format(Field, UpdatedValue, UserId)  
 cur.execute(qa)  
 cur.execute(**'COMMIT'**)  
 cur.execute(**"SELECT \* FROM Admin WHERE Id={}"**.format(UserId))  
 aData = cur.fetchone()  
 con.close()  
 **return** aData  
 **else**:  
 con.close()  
 **return 'Error!'***# VIEW ANNOUCEMENT STUDENT***def** view\_announcement\_student(StudentUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query1 = **'''SELECT grade.id from grade  
 inner join class on class.gradeid=grade.id  
 inner join studentclass on studentclass.classid=class.id  
 inner join student on student.id=studentclass.studentid  
 where student.id={}'''** cur.execute(query1.format(StudentUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 grade\_id = i[0]  
 query = **'''SELECT Teacher.Name, Announcement.SubjectPreview,   
 Announcement.AnnouncementDate,   
 Announcement.Attachment  
 FROM Announcement  
 INNER JOIN Teacher ON Teacher.Id = Announcement.TeacherId  
 WHERE announcement.gradeid={}  
 ORDER BY Announcement.GradeId;'''** cur.execute(query.format(grade\_id))  
 d = tuple(cur.fetchall())  
 header = ((**'Teacher'**, **'SubjectPreview'**, **'Date'**, **'Attachment'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW ASSIGNMENT STUDENT***def** view\_assignment\_student(StudentUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 List = []  
 query1 = **'''select class.gradeid,student.id  
 from student  
 inner join studentclass on studentclass.studentid=student.id  
 inner join class on class.id=studentclass.classid  
 where student.id={}'''** cur.execute(query1.format(StudentUser))  
 d = cur.fetchall()  
 List.append(d[0][0])  
 **for** i **in** List:  
 class\_int = int(i)  
 query2 = **'''SELECT Subject.SubName, Teacher.Name, Assignment.Topic,   
 Assignment.StartDate, Assignment.DueDate, Assignment.Attachment  
 FROM Assignment  
 INNER JOIN Teacher ON Teacher.Id = Assignment.TeacherId  
 INNER JOIN Subject ON Subject.Id = Assignment.SubjectId  
 WHERE Assignment.gradeid={}'''** cur.execute(query2.format(class\_int))  
 d = tuple(cur.fetchall())  
 header = ((**'Subject'**, **'Teacher'**, **'Topic'**, **'StartDate'**, **'DueDate'**, **'Attachment'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW TEST RESULT STUDENT***def** view\_testResult\_student(StudentUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 print(**'''1.UNIT TEST 1  
2.QUARTERLY EXAMINATION  
3.HALF YEARLY EXAMINATION  
4.UNIT TEST 2  
5.ANNUAL EXAMINATION'''**)  
 exam = int(input(**'Select exam: '**))  
 query = **'''SELECT subject.subname,  
 marks.marksscored  
 from student  
 inner join marks on marks.studentid=student.id  
 inner join exam on exam.id=marks.examid  
 inner join subject on subject.id=exam.subjectid  
 inner join examgroup on examgroup.id=exam.groupid  
 where student.id={} && examgroup.id=101'''  
 if** exam == 1:  
 cur.execute(query.format(StudentUser, 101))  
 d = tuple(cur.fetchall())  
 header = ((**'SUBJECT'**, **'MARKS'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 **if** exam == 2:  
 cur.execute(query.format(StudentUser, 103))  
 d = tuple(cur.fetchall())  
 header = ((**'SUBJECT'**, **'MARKS'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 **if** exam == 3:  
 cur.execute(query.format(StudentUser, 104))  
 d = tuple(cur.fetchall())  
 header = ((**'SUBJECT'**, **'MARKS'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 **if** exam == 4:  
 cur.execute(query.format(StudentUser, 102))  
 d = tuple(cur.fetchall())  
 header = ((**'SUBJECT'**, **'MARKS'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 **if** exam == 5:  
 cur.execute(query.format(StudentUser, 105))  
 d = tuple(cur.fetchall())  
 header = ((**'SUBJECT'**, **'MARKS'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# WELCOME BACK TEACHER***def** message\_hello\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 cur.execute(**"SELECT name from teacher where id={}"**.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 name = i[0]  
 print(**"\nWelcome"**, name + **"!"**)  
 con.close()  
  
  
*# VIEW TEACHER PROFILE***def** view\_teacherProfile\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query = **'''SELECT Teacher.id, Teacher.name,subject.SubName,  
 Teacher.photo,Teacher.coordinator,  
 Teacher.dateofbirth,Teacher.EdQualifications,  
 Teacher.phoneno,Teacher.status  
 from teacher  
 inner join teachersubject on teachersubject.teacherid=Teacher.id  
 inner join subject on subject.id=teachersubject.Subjectid  
 where Teacher.id={}'''**.format(TeacherUser)  
 cur.execute(query)  
 d = cur.fetchall()  
 **for** i **in** d:  
 print(**"\nID"**, i[0], sep=**": "**)  
 print(**"Name"**, i[1], sep=**": "**)  
 print(**"Subject:"**, i[2], sep=**": "**)  
 print(**"Photo"**, i[3], sep=**": "**)  
 print(**"Coordinator"**, i[4], sep=**": "**)  
 print(**"D.O.B"**, i[5], sep=**": "**)  
 print(**"EdQualifications"**, i[6], sep=**": "**)  
 print(**"Contact number"**, i[7], sep=**": "**)  
 print(**"Status"**, i[8], sep=**": "**, end=**'\n\n'**)  
 con.close()  
  
  
*# IS COORDINATOR CHECK***def** check\_coordinator\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query2 = **'''SELECT coordinator from teacher where Id={}'''** cur.execute(query2.format(TeacherUser))  
 d = cur.fetchall()  
 **for** j **in** d:  
 coordinator = j[0]  
 **if** coordinator == **'T'**:  
 con.close()  
 **return True** con.close()  
 **return False***# VIEW ANNOUNCEMENT TEACHER***def** view\_announcements\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query1 = **'''SELECT grade.id from grade  
 inner join class on class.gradeid=grade.id  
 inner join teachersubjectclass on teachersubjectclass.classid=class.id  
 inner join teachersubject on teachersubject.subjectid=teachersubjectclass.teachersubjectid  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 where Teacher.id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 **if** d == []:  
 print(**"Annnouncements Data Not Found."**, end=**"\n\n"**)  
 **return None  
 for** i **in** d:  
 grade\_id = i[0]  
 query2 = **'''SELECT Announcement.Id, Teacher.Name, Announcement.SubjectPreview,   
 Announcement.AnnouncementDate,   
 Announcement.Attachment  
 FROM Announcement  
 INNER JOIN Teacher ON Teacher.Id = Announcement.TeacherId  
 WHERE announcement.gradeid={}  
 ORDER BY Announcement.GradeId;'''** cur.execute(query2.format(grade\_id))  
 d = tuple(cur.fetchall())  
 header = ((**'S.No'**, **'Teacher'**, **'SubjectPreview'**, **'Date'**, **'Attachment'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD ANNOUNCEMENT TEACHER***def** add\_announcement\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **'''SELECT grade.id from grade  
 inner join class on class.gradeid=grade.id  
 inner join teachersubjectclass on teachersubjectclass.classid=class.id  
 inner join teachersubject on teachersubject.subjectid=teachersubjectclass.teachersubjectid  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 where Teacher.id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 grade\_id = i[0]  
 query2 = **'''SELECT COUNT(\*) from announcement'''** cur.execute(query2)  
 d = cur.fetchall()  
 **for** i **in** d:  
 length = i[0]  
 sn\_no = length + 1  
 query3 = **'''SELECT CURDATE()'''** cur.execute(query3)  
 d = cur.fetchall()  
 **for** i **in** d:  
 date = i[0]  
 sub\_preview = input(**"Enter subject:"**)  
 attachment = input(**"Enter attachment:"**)  
 query4 = **'''INSERT INTO announcement VALUES({},{},{},'{}','{}','{}')'''** cur.execute(query4.format(sn\_no, grade\_id, TeacherUser, date, sub\_preview, attachment))  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# DELETE ANNOUNCEMENT TEACHER***def** del\_announcement\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **'''SELECT grade.Id from grade  
 inner join class on class.gradeid=grade.Id  
 inner join teachersubjectclass on teachersubjectclass.classid=class.Id  
 inner join teachersubject on teachersubject.subjectid=teachersubjectclass.teachersubjectid  
 inner join teacher on Teacher.Id=teachersubject.teacherid  
 where Teacher.Id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 grade\_id = i[0]  
 query2 = **'''SELECT coordinator from teacher where Id={}'''** cur.execute(query2.format(TeacherUser))  
 d = cur.fetchall()  
 **for** j **in** d:  
 coordinator = j[0]  
 **if** coordinator == **'T'**:  
 Id = int(input(**"Enter Announcement ID to be deleted:"**))  
 query3 = **'''SELECT gradeid from announcement where Id={}'''** cur.execute(query3.format(Id))  
 d = cur.fetchall()  
 **for** k **in** d:  
 g = k[0]  
 **if** grade\_id == g:  
 query4 = **'''DELETE FROM announcement WHERE Id={}'''** cur.execute(query4.format(Id))  
 con.commit()  
 print(**"Deletion complete"**, end=**"\n\n"**)  
 **else**:  
 print(**"Unable to carry out deletion"**, end=**"\n\n"**)  
 **else**:  
 print(**"Unable to carry out deletion"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW ASSIGNMENT TEACHER***def** view\_assignment\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query = **'''SELECT subject.id from subject  
 inner join teachersubject on teachersubject.subjectid=subject.id  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 WHERE Teacher.id={}'''** cur.execute(query.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 subject\_id = i[0]  
 query2 = **'''SELECT Assignment.Id, Subject.SubName, Teacher.Name, Assignment.Topic,   
 Assignment.StartDate, Assignment.DueDate, Assignment.Attachment  
 FROM Assignment  
 INNER JOIN Teacher ON Teacher.Id = Assignment.TeacherId  
 INNER JOIN Subject ON Subject.Id = Assignment.SubjectId  
 WHERE Assignment.subjectid={}'''** cur.execute(query2.format(subject\_id))  
 d = tuple(cur.fetchall())  
 header = ((**'S.No.''Subject'**, **'Teacher'**, **'Topic'**, **'StartDate'**, **'DueDate'**, **'Attachment'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD ASSIGNMENT TEACHER***def** add\_assignment\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **'''SELECT subject.id from subject  
 inner join teachersubject on teachersubject.subjectid=subject.id  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 WHERE Teacher.id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 subject\_id = i[0]  
 query2 = **'''SELECT MAX(id) from assignment'''** cur.execute(query2)  
 d = cur.fetchall()  
 **for** i **in** d:  
 length = i[0]  
 sn\_no = length + 1  
 grade = int(input(**"Enter grade:"**))  
 topic = input(**"Enter topic:"**)  
 start\_date = input(**"Enter start date:"**)  
 due\_date = input(**"Enter due date:"**)  
 attachment = input(**"Enter attachment:"**)  
 query3 = **'''INSERT INTO assignment values({},{},{},{},'{}','{}','{}','{}')'''** cur.execute(query3.format(sn\_no, TeacherUser, grade, subject\_id, topic, start\_date, due\_date, attachment))  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# DELETE ASSIGNMENT TEACHER***def** del\_assignment\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **'''SELECT subject.id from subject  
 inner join teachersubject on teachersubject.subjectid=subject.id  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 WHERE Teacher.id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 **for** i **in** d:  
 subject\_id = i[0]  
 cur.execute(**"SELECT COUNT(\*) FROM assignment"**)  
 d = cur.fetchall()  
 **for** i **in** d:  
 p = i[0]  
 assignment\_id = int(input(**"Enter assignment no:"**))  
 **if** assignment\_id <= p:  
 query = **'''delete from assignment where id={} && subjectid={}'''** cur.execute(query.format(assignment\_id, subject\_id))  
 con.commit()  
 print(**"Successfully deleted!"**, end=**"\n\n"**)  
 **else**:  
 print(**"Invalid assignment number"**, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD TEST RESULT TEACHER***def** add\_testResult\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **"select subjectid from teachersubject where teacherid={}"**.format(TeacherUser)  
 cur.execute(query1)  
 [subject\_id] = cur.fetchall()  
 **for** i **in** subject\_id:  
 query2 = **"select id, subname from subject where id={}"**.format(i)  
 cur.execute(query2)  
 [(Id, Subjects)] = cur.fetchall()  
 print(**"Subjects:"**)  
 print(str(Id) + **'.'**, Subjects)  
 subject = int(input(**"Enter subject ID: "**))  
 print(**'''\nExaminations:  
 1. Annual Examination   
 2. Half-Yearly Examination   
 3. Pre-Board 1   
 4. Pre-Board 2   
 5. Quarterly Examination   
 6. Unit Test 1   
 7. Unit Test 2   
 8. Weekly Test'''**)  
 examgroup = int(input(**"Select exam name:"**))  
 **if** examgroup == 1:  
 examgroup = 105  
 **elif** examgroup == 2:  
 examgroup = 104  
 **elif** examgroup == 3:  
 examgroup = 106  
 **elif** examgroup == 4:  
 examgroup = 107  
 **elif** examgroup == 5:  
 examgroup = 103  
 **elif** examgroup == 6:  
 examgroup = 101  
 **elif** examgroup == 7:  
 examgroup = 102  
 **elif** examgroup == 8:  
 examgroup = 100  
 student\_id = int(input(**"Enter Student ID: "**))  
 query3 = **"""Select gradeid from class  
 Inner join studentclass on studentclass.classid=class.id  
 Where studentclass.studentid={};"""**.format(student\_id)  
 cur.execute(query3)  
 [(grade,)] = cur.fetchall()  
 query4 = **"SELECT id FROM exam where gradeid={} && subjectid={} && groupid={};"**.format(grade, subject, examgroup)  
 cur.execute(query4)  
 [(exam\_id,)] = cur.fetchall()  
 marks = int(input(**"Enter marks scored: "**))  
 query5 = **"SELECT max(id) from marks;"** cur.execute(query5)  
 [(s\_no,)] = cur.fetchall()  
 s\_no = int(s\_no) + 1  
 query6 = **"INSERT INTO marks values({},{},{},{});"**.format(s\_no, student\_id, exam\_id, marks)  
 cur.execute(query6)  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW TEST RESULT TEACHER***def** view\_testResult\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query1 = **"select subjectid from teachersubject where teacherid={}"**.format(TeacherUser)  
 cur.execute(query1)  
 [subject\_id] = cur.fetchall()  
 **for** i **in** subject\_id:  
 query2 = **"select id, subname from subject where id={}"**.format(i)  
 cur.execute(query2)  
 [(Id, Subjects)] = cur.fetchall()  
 print(**"Subjects:"**)  
 print(str(Id) + **'.'**, Subjects)  
 subject = int(input(**"Enter subject ID: "**))  
 print(**'''\nExaminations:  
 1. Annual Examination  
 2. Half-Yearly Examination  
 3. Pre-Board 1  
 4. Pre-Board 2  
 5. Quarterly Examination  
 6. Unit Test 1  
 7. Unit Test 2  
 8. Weekly Test'''**)  
 examgroup = int(input(**"Select exam name:"**))  
 **if** examgroup == 1:  
 examgroup = 105  
 **elif** examgroup == 2:  
 examgroup = 104  
 **elif** examgroup == 3:  
 examgroup = 106  
 **elif** examgroup == 4:  
 examgroup = 107  
 **elif** examgroup == 5:  
 examgroup = 103  
 **elif** examgroup == 6:  
 examgroup = 101  
 **elif** examgroup == 7:  
 examgroup = 102  
 **elif** examgroup == 8:  
 examgroup = 100  
 query = **'''SELECT student.id, student.name, exam.gradeid, examgroup.type, subject.subname,   
 marks.marksscored, exam.maxmark FROM exam INNER JOIN examgroup ON examgroup.id = exam.groupid   
 INNER JOIN subject ON subject.id=exam.subjectid   
 INNER JOIN marks ON marks.examid = exam.id   
 INNER JOIN student ON student.id = marks.studentid   
 WHERE exam.subjectid={} && marks.examid={}  
 ORDER BY student.name, exam.gradeid, examgroup.type, subject.subname asc;'''**.format(subject, examgroup)  
  
 cur.execute(query)  
 header = ((**'Student ID'**, **'Student Name'**, **'Class'**, **'Exam Name'**, **'Subject'**, **'Marks'**, **'Total Marks'**),)  
 d = tuple(cur.fetchall())  
 num\_of\_rows = cur.rowcount  
 **if** num\_of\_rows == 0:  
 print(**"Data Not Found.\n"**)  
 **else**:  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**'\n\n'**)  
 con.close()  
  
  
*# VIEW STUDENT DETAILS TEACHER***def** view\_studentDetails\_teacher(TeacherUser: int):  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 counter = 0  
 query1 = **'''SELECT grade.id,class.division  
 FROM grade  
 inner join class on class.gradeid=grade.id  
 inner join teachersubjectclass on teachersubjectclass.classid=class.id  
 inner join teachersubject on teachersubject.subjectid=teachersubjectclass.teachersubjectid  
 inner join teacher on Teacher.id=teachersubject.teacherid  
 where Teacher.id={}'''** cur.execute(query1.format(TeacherUser))  
 d = cur.fetchall()  
 class\_ = int(input(**"Enter class:"**))  
 division\_ = input(**"Enter division"**)  
 division\_upper = division\_.upper()  
 **for** i **in** d:  
 **if** i[0] == class\_ **and** i[1] == division\_upper:  
 query = **'''SELECT student.id, student.name,grade.id,class.division,  
 student.photo,student.gender,  
 student.dateofbirth,student.bloodgroup,student.phoneno,student.address,  
 student.dateofadmission,student.password  
 from student   
 inner join studentclass on studentclass.studentid=student.id  
 inner join class on class.id=studentclass.classid  
 inner join grade on grade.id=class.gradeid  
 WHERE grade.id={} && class.division='{}'  
 order by student.id'''** cur.execute(query.format(class\_, division\_upper))  
 header = ((**'ID'**, **'Name'**, **'Class'**, **'section'**, **'Photo'**, **'Gender'**, **'D.O.B'**, **'Blood Group'**, **'Ph.No'**, **'Address'**,  
 **'Date of admission'**, **'Password'**),)  
 d = tuple(cur.fetchall())  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 counter += 1  
 **else**:  
 **pass  
 if** counter == 0:  
 print(**"INVALID CLASS/DIVISION"**, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD TEST RESULT ADMIN***def** add\_testResult\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 print(**'''Subjects:  
 0. MATHEMATICS 1  
 1. SCIENCE  
 2. SOCIAL SCIENCE  
 3. ENGLISH 1  
 4. PHYSICS  
 5. CHEMISTRY  
 6. MATHEMATICS 2  
 7. ENGLISH 2  
 8. COMPUTER SCIENCE  
 9. BIOLOGY'''**)  
 subject = int(input(**"Enter subject: "**)) + 1  
 print(**'''\nExaminations:  
 1. Annual Examination   
 2. Half-Yearly Examination   
 3. Pre-Board 1   
 4. Pre-Board 2   
 5. Quarterly Examination   
 6. Unit Test 1   
 7. Unit Test 2   
 8. Weekly Test'''**)  
 examgroup = int(input(**"Select exam name:"**))  
 **if** examgroup == 1:  
 examgroup = 105  
 **elif** examgroup == 2:  
 examgroup = 104  
 **elif** examgroup == 3:  
 examgroup = 106  
 **elif** examgroup == 4:  
 examgroup = 107  
 **elif** examgroup == 5:  
 examgroup = 103  
 **elif** examgroup == 6:  
 examgroup = 101  
 **elif** examgroup == 7:  
 examgroup = 102  
 **elif** examgroup == 8:  
 examgroup = 100  
 student\_id = int(input(**"Enter Student ID: "**))  
 q1 = **"""Select gradeid from class  
 Inner join studentclass on studentclass.classid=class.id  
 Where studentclass.studentid={};"""**.format(student\_id)  
 cur.execute(q1)  
 grade = cur.fetchall()  
 q2 = **"SELECT id FROM exam where gradeid={} && subjectid={} && groupid={};"**.format(grade,  
 subject, examgroup)  
 cur.execute(q2)  
 exam\_id = int(cur.fetchall())  
 marks = int(input(**"Enter marks scored: "**))  
 q3 = **"SELECT max(id) from marks;"** cur.execute(q3)  
 s\_no = cur.fetchall() + 1  
 cur.execute(**"INSERT INTO marks values({},{},{},{})"**.format(s\_no,  
 student\_id,  
 exam\_id,  
 marks))  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW ANNOUNCEMENTS ADMIN***def** view\_announcements\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query = **'''SELECT Announcement.Id, Teacher.Name, Announcement.SubjectPreview,   
 Announcement.AnnouncementDate,   
 Announcement.Attachment  
 FROM Announcement  
 INNER JOIN Teacher ON Teacher.Id = Announcement.TeacherId  
 ORDER BY Announcement.GradeId;'''** cur.execute(query)  
 d = tuple(cur.fetchall())  
 header = ((**'Sn.No'**, **'Teacher'**, **'SubjectPreview'**, **'Date'**, **'Attachment'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEWING TEACHER DETAILS ADMIN***def** view\_teacherDetails\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 cur.execute(**"Select \* from teacher"**)  
 header = ((**"ID"**, **"Name"**, **"Photo"**, **"Ph.No"**, **"D.O.B"**, **"Coordinator"**, **"Ed Qualifications"**, **"Status"**, **"Password"**),)  
 d = tuple(cur.fetchall())  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# ADDING TEACHER DETAILS ADMIN***def** add\_teacherDetails\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 teacher\_id = int(input(**"Enter teacher id:"**))  
 name = input(**"Enter teacher name:"**)  
 photo = input(**"Enter photo jpg extension:"**)  
 ph\_no = int(input(**"Enter phone number:"**))  
 date\_of\_birth = input(**"Enter date of birth:"**)  
 coordinator = input(**"Enter coordinator:"**)  
 qualifications = input(**"Enter ed qualifications:"**)  
 status = input(**"Enter status:"**)  
 password = input(**"Enter password:"**)  
 query = **"INSERT INTO teacher values({},'{}','{}',{},'{}','{}','{}','{}','{}')"** cur.execute(query.format(teacher\_id, name, photo, ph\_no,  
 date\_of\_birth, coordinator, qualifications,  
 status, password))  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# EDIT TEACHER PROFILE ADMIN***def** edit\_teacherProfile():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 teacher\_id = int(input(**"Enter teacher id to update:"**))  
 print(**'''What would you like to update:  
 1. Name  
 2. Photo  
 3.Phone Number  
 4. Date Of Birth  
 5. Is a Co-ordinator?  
 6. Educational Qualifications  
 7. Status  
 8. Password'''**)  
 n = int(input(**"Enter selection:"**))  
 **if** n == 1:  
 name = input(**"Enter teacher name:"**)  
 cur.execute(**"UPDATE teacher SET name='{}' WHERE id={}"**.format(name, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 2:  
 photo = input(**"Enter photo jpg extension:"**)  
 cur.execute(**"UPDATE teacher SET photo='{}' WHERE id={}"**.format(photo, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 3:  
 ph\_no = input(**"Enter phone number:"**)  
 cur.execute(**"UPDATE teacher SET phoneno='{}' WHERE id={}"**.format(ph\_no, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 4:  
 date\_of\_birth = input(**"Enter date of birth:"**)  
 cur.execute(**"UPDATE teacher SET DateOfBirth='{}' WHERE id={}"**.format(date\_of\_birth, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 5:  
 coordinator = input(**"Enter coordinator:"**)  
 cur.execute(**"UPDATE teacher SET coordinator='{}' WHERE id={}"**.format(coordinator, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 6:  
 EdQualifications = int(input(**"Enter EdQualifications::"**))  
 cur.execute(**"UPDATE teacher SET EdQualifications={} WHERE id={}"**.format(EdQualifications, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 7:  
 status = input(**"Enter status:"**)  
 cur.execute(**"UPDATE teacher SET status'{}' WHERE id={}"**.format(status, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 8:  
 password = input(**"Enter password:"**)  
 cur.execute(**"UPDATE teacher SET password='{}' WHERE id={}"**.format(password, teacher\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 con.close()  
  
  
*# DELETE TEACHER PROFILE ADMIN***def** del\_teacherProfile():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 teacher\_id = int(input(**"Enter Teacher ID:"**))  
 confirm = input(**"CONFIRM DELETION(Y/N):"**).upper()  
 **if** confirm == **'Y'**:  
 cur.execute(**"DELETE from teacher where id={}"**.format(teacher\_id))  
 print(**"Deletion successful"**, end=**"\n\n"**)  
 **elif** confirm == **'N'**:  
 print(**"Deletion halted"**, end=**"\n\n"**)  
 **else**:  
 print(**"INVALID OPTION"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW STUDENT DETAILS ADMIN***def** view\_studentDetails\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query = **'''SELECT student.id, student.name,grade.id,class.division,  
 student.photo,student.gender,  
 student.dateofbirth,student.bloodgroup,student.phoneno,student.address,  
 student.dateofadmission,student.password  
 from student   
 inner join studentclass on studentclass.studentid=student.id  
 inner join class on class.id=studentclass.classid  
 inner join grade on grade.id=class.gradeid  
 order by student.id'''** cur.execute(query)  
 header = ((**'ID'**, **'Name'**, **'Class'**, **'section'**, **'Photo'**, **'Gender'**, **'D.O.B'**, **'Blood Group'**, **'Ph.No'**, **'Address'**,  
 **'Date of admission'**, **'Password'**),)  
 d = tuple(cur.fetchall())  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD STUDENT DETAILS ADMIN***def** add\_studentDetails\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 student\_id = int(input(**"Enter student id:"**))  
 name = input(**"Enter student name:"**)  
 class\_ = int(input(**"Enter class: "**))  
 division = input(**"Enter division: "**)  
 photo = input(**"Enter photo (.jpg extension):"**)  
 gender = input(**"Enter gender:(M/F):"**)  
 date\_of\_birth = input(**"Enter date of birth:"**)  
 blood\_group = input(**"Enter blood group:"**)  
 ph\_no = int(input(**"Enter phone number:"**))  
 address = input(**"Enter address:"**)  
 date\_of\_admission = input(**"Enter date of admission:"**)  
 password = input(**"Enter password:"**)  
 q1 = **"INSERT INTO student values({},'{}','{}','{}','{}','{}',{},'{}','{}','{}')"** cur.execute(q1.format(student\_id, name, photo,  
 gender, date\_of\_birth,  
 blood\_group, ph\_no, address,  
 date\_of\_admission, password))  
 con.commit()  
 q2 = **"SELECT Id FROM class where GradeId={} && Division='{}'"**.format(class\_, division)  
 cur.execute(q2)  
 class\_id = cur.fetchall()  
 q3 = **"INSERT INTO studentclass VALUES({}, {})"**.format(student\_id, class\_id)  
 cur.execute(q3)  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# EDIT STUDENT PROFILE ADMIN***def** edit\_studentProfile\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 student\_id = int(input(**"Enter student id:"**))  
 print(**'''What would you like to update?  
 1. Name  
 2. Photo  
 3. Gender  
 4. Date Of Birth  
 5. Blood Group  
 6. Phone Number  
 7. Address  
 8. Password'''**)  
 n = int(input(**"Enter selection: "**))  
 **if** n == 1:  
 name = input(**"Enter student name:"**)  
 cur.execute(**"UPDATE student SET name='{}' WHERE id={}"**.format(name, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 2:  
 photo = input(**"Enter photo jpg extension:"**)  
 cur.execute(**"UPDATE student SET photo='{}' WHERE id={}"**.format(photo, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 3:  
 gender = input(**"Enter gender(M/F):"**)  
 cur.execute(**"UPDATE student SET gender='{}' WHERE id={}"**.format(gender, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 4:  
 date\_of\_birth = input(**"Enter date of birth:"**)  
 cur.execute(**"UPDATE student SET DateOfBirth='{}' WHERE id={}"**.format(date\_of\_birth, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 5:  
 blood\_group = input(**"Enter bloodgroup:"**)  
 cur.execute(**"UPDATE student SET bloodGroup='{}' WHERE id={}"**.format(blood\_group, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 6:  
 ph\_no = int(input(**"Enter phone number:"**))  
 cur.execute(**"UPDATE student SET phoneno={} WHERE id={}"**.format(ph\_no, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 7:  
 address = input(**"Enter address:"**)  
 cur.execute(**"UPDATE student SET address='{}' WHERE id={}"**.format(address, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 **if** n == 8:  
 password = input(**"Enter password:"**)  
 cur.execute(**"UPDATE student SET password='{}' WHERE id={}"**.format(password, student\_id))  
 con.commit()  
 print(**"Successfully edited"**, end=**"\n\n"**)  
 con.close()  
  
  
*# DELETE STUDENT PROFILE ADMIN***def** del\_studentProfile\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 student\_id = int(input(**"Enter student ID:"**))  
 confirm = input(**"CONFIRM DELETION(Y/N):"**).upper()  
 **if** confirm == **'Y'**:  
 cur.execute(**"DELETE from student where id={}"**.format(student\_id))  
 print(**"Deletion successful"**, end=**"\n\n"**)  
 **elif** confirm == **'N'**:  
 print(**"Deletion halted"**, end=**"\n\n"**)  
 **else**:  
 print(**"INVALID OPTION"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW TEST RESULT ADMIN***def** view\_testResult\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 print(**'''Subjects:  
 0. MATHEMATICS 1  
 1. SCIENCE  
 2. SOCIAL SCIENCE  
 3. ENGLISH 1  
 4. PHYSICS  
 5. CHEMISTRY  
 6. MATHEMATICS 2  
 7. ENGLISH 2  
 8. COMPUTER SCIENCE  
 9. BIOLOGY'''**)  
 subject = int(input(**"Enter subject: "**)) + 1  
 print(**'''\nExaminations:  
 1. Annual Examination   
 2. Half-Yearly Examination   
 3. Pre-Board 1   
 4. Pre-Board 2   
 5. Quarterly Examination   
 6. Unit Test 1   
 7. Unit Test 2   
 8. Weekly Test'''**)  
 examgroup = int(input(**"Select exam name:"**))  
 **import** tabulate  
 **if** examgroup == 1:  
 examgroup = 105  
 **elif** examgroup == 2:  
 examgroup = 104  
 **elif** examgroup == 3:  
 examgroup = 106  
 **elif** examgroup == 4:  
 examgroup = 107  
 **elif** examgroup == 5:  
 examgroup = 103  
 **elif** examgroup == 6:  
 examgroup = 101  
 **elif** examgroup == 7:  
 examgroup = 102  
 **elif** examgroup == 8:  
 examgroup = 100  
 query = **'''SELECT student.id, student.name, exam.gradeid, examgroup.type, subject.subname,   
 marks.marksscored, exam.maxmark FROM exam INNER JOIN examgroup ON examgroup.id = exam.groupid   
 INNER JOIN subject ON subject.id=exam.subjectid   
 INNER JOIN marks ON marks.examid = exam.id   
 INNER JOIN student ON student.id = marks.studentid   
 WHERE exam.subjectid={} && marks.examid={}  
 ORDER BY student.name, exam.gradeid, examgroup.type, subject.subname asc;'''**.format(subject, examgroup)  
  
 cur.execute(query)  
 header = ((**'Student ID'**, **'Student Name'**, **'Class'**, **'Exam Name'**, **'Subject'**, **'Marks'**, **'Total Marks'**),)  
 d = tuple(cur.fetchall())  
 num\_of\_rows = cur.rowcount  
 **if** num\_of\_rows == 0:  
 print(**"Data Not Found.\n"**)  
 **else**:  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**'fancy\_grid'**)  
 print(table, end=**'\n\n'**)  
 con.close()  
  
  
*# VIEW CLASSES ADMIN***def** view\_classes\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query = **'''SELECT grade.id,class.division  
 from grade  
 inner join class on class.gradeid=grade.id'''** cur.execute(query)  
 d = tuple(cur.fetchall())  
 header = ((**'Class'**, **'Division'**),)  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 con.close()  
  
  
*# ADD CLASS ADMIN***def** add\_class\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 query1 = **'''Select count(\*) from class'''** cur.execute(query1)  
 d = cur.fetchall()  
 Id = d + 1  
 grade\_id = int(input(**"Enter Grade: "**))  
 division = input(**"Enter Division:"**)  
 query = **'''INSERT INTO class values({},{},'{}')'''** cur.execute(query.format(Id, grade\_id, division))  
 con.commit()  
 print(**"Successfully added"**, end=**"\n\n"**)  
 con.close()  
  
  
*# VIEW ASSIGNMENT ADMIN***def** view\_assignment\_admin():  
 con = connect(pswrd=psword)  
 cur = con.cursor()  
 **import** tabulate  
 query = **'''SELECT Assignment.Id, Subject.SubName, Teacher.Name, Assignment.Topic, Assignment.StartDate,   
 Assignment.DueDate, Assignment.Attachment   
 FROM Assignment   
 INNER JOIN Teacher ON Teacher.Id = Assignment.TeacherId   
 INNER JOIN Subject ON Subject.Id = Assignment.SubjectId   
 ORDER BY Assignment.GradeId; '''** cur.execute(query)  
 header = ((**"Sn.No."**, **"Grade"**, **"Subject"**, **"Teacher"**, **"Topic"**, **"StartDate"**, **"DueDate"**, **"Attachment"**),)  
 d = tuple(cur.fetchall())  
 retrieve = header + d  
 table = tabulate.tabulate(retrieve, tablefmt=**"fancy\_grid"**)  
 print(table, end=**"\n\n"**)  
 con.close()

# **EdSt.main\_CUI**

**Code:**

**import** mysql.connector **as** conn  
**import** EdSt.Function\_CUI **as** cui  
  
*# main program loop***while True**:  
 user = int(input(**"User ID: "**)) *# User Id* Password = input(**"Password: "**) *# Password* SignIn = cui.SignIn(user, Password) *# Signing in. Checking user access.  
  
 # Student User* **if** SignIn == **'Student.valid'**:  
 cui.message\_hello\_student(user)  
 **while True**:  
 *# Dashboard Menu* print(**"""\n\tDashboard  
 1. Profile  
 2. Announcement  
 3. Assignment  
 4. Marks  
 5. Log Out"""**)  
 **try**:  
 student\_menu = int(input(**"Enter Menu Selection: "**)) *# Dashboard menu* **try**:  
 *# if Student User chooses 'Profile'* **if** student\_menu == 1:  
 *# Displaying student profile info* s\_Profile = cui.view\_studentProfile\_student(user)  
  
 **while True**:  
 *# Profile Menu* print(**"1. Change Password?\n"  
 "2. Back to Dashboard"**)  
 student\_profile\_menu = int(input(**"Enter menu Selection: "**))  
  
 *# if Student User chooses 'Change Password?'* **if** student\_profile\_menu == 1:  
 old\_password = input(**"Enter old passward: "**) *# Accept old password  
  
 # Compare user entered password with original password (Check access)* **if** old\_password == Password:  
 new\_password = input(**"Enter new password: "**)  
 new\_password\_confirm = input(**"Confirm new password: "**)  
 **if** new\_password == new\_password\_confirm:  
 cui.edit\_profile\_all(SignIn, user, **'Password'**, new\_password\_confirm)  
 print(**"Password succefully changed!"**)  
 **else**:  
 print(**"Error! The passwords do not match!"**)  
 **else**:  
 print(**"Error! Incorrect password!"**)  
 **elif** student\_profile\_menu == 2:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** student\_menu == 2:  
 print(**"Announcements"**)  
 cui.view\_announcement\_student(user)  
 **elif** student\_menu == 3:  
 print(**"Assignments"**)  
 cui.view\_assignment\_student(user)  
 **elif** student\_menu == 4:  
 print(**"Marks"**)  
 cui.view\_testResult\_student(user)  
 **elif** student\_menu == 5:  
 print(**"\nLogging Out..."**)  
 print(**"Logged Out!\n"**)  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **except** conn.errors.DataError:  
 print(**"Error! Invalid input!"**)  
 **except** ValueError:  
 print(**"Invalid input. Try again."**)  
  
 **elif** SignIn == **'Teacher.valid'**:  
 cui.message\_hello\_teacher(user)  
 **while True**:  
 print(**"""  
 \tDashboard  
 1. Profile  
 2. Announcement  
 3. Assignment  
 4. Marks  
 5. Student Details  
 6. Log Out"""**)  
 **try**:  
 teacher\_menu = int(input(**"Enter Menu Selection: "**)) *# Dashboard menu* **try**:  
 **if** teacher\_menu == 1:  
 *# Displaying teacher profile info* t\_Profile = cui.view\_teacherProfile\_teacher(user)  
 print(t\_Profile)  
  
 **while True**:  
 *# Profile Menu* print(**"1. Change Password?\n"  
 "2. Back to Dashboard"**)  
 teacher\_profile\_menu = int(input(**"Enter Menu Selection: "**))  
  
 *# if Teacher User chooses 'Change Password?'* **if** teacher\_profile\_menu == 1:  
 old\_password = input(**"Enter old passward: "**) *# Accept old password  
  
 # Compare user entered password with original password (Check access)* **if** old\_password == Password:  
 new\_password = input(**"Enter new password: "**)  
 new\_password\_confirm = input(**"Confirm new password: "**)  
 **if** new\_password == new\_password\_confirm:  
 cui.edit\_profile\_all(SignIn, user, **'Password'**, new\_password\_confirm)  
 print(**"Password succefully changed!"**)  
 **else**:  
 print(**"Error! The passwords do not match!"**)  
  
 **else**:  
 print(**"Error! Incorrect password!"**)  
 **elif** teacher\_profile\_menu == 2:  
 **break  
  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** teacher\_menu == 2:  
 coordinator = cui.check\_coordinator\_teacher(user)  
  
 *# if Teacher is Coordinator* **if** coordinator **is True**:  
 **while True**:  
 teacher\_menu\_2 = **"""  
 Announcements  
 1) View announcements  
 2) Add announcement  
 3) Delete announcement  
 4) Return to Dashboard"""** print(teacher\_menu\_2)  
 teacher\_announcements\_menu = int(input(**"Enter selection: "**))  
 **if** teacher\_announcements\_menu == 1:  
 print(**"Announcements"**)  
 cui.view\_announcements\_teacher(user)  
 **elif** teacher\_announcements\_menu == 2:  
 cui.add\_announcement\_teacher(user)  
 **elif** teacher\_announcements\_menu == 3:  
 cui.del\_announcement\_teacher(user)  
 **elif** teacher\_announcements\_menu == 4:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **else**:  
 cui.view\_announcements\_teacher(user)  
  
 **elif** teacher\_menu == 3:  
 **while True**:  
 teacher\_menu\_3 = **"""  
 Assignments  
 1) View assignments  
 2) Add assignment  
 3) Delete assignment  
 4) Return to Dashboard"""** print(teacher\_menu\_3)  
 teacher\_assignments\_menu = int(input(**"Enter selection: "**))  
 **if** teacher\_assignments\_menu == 1:  
 print(**"Assignments"**)  
 cui.view\_assignment\_teacher(user)  
 **elif** teacher\_assignments\_menu == 2:  
 cui.add\_assignment\_teacher(user)  
 **elif** teacher\_assignments\_menu == 3:  
 cui.del\_assignment\_teacher(user)  
 **elif** teacher\_assignments\_menu == 4:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** teacher\_menu == 4:  
 **while True**:  
 teacher\_menu\_4 = **"""  
 Marks  
 1) View student marks  
 2) Add student marks  
 3) Return to Dashboard"""** print(teacher\_menu\_4)  
 teacher\_marks\_menu = int(input(**"Enter selection: "**))  
 **if** teacher\_marks\_menu == 1:  
 print(**"Marks"**)  
 cui.view\_testResult\_teacher(user)  
 **elif** teacher\_marks\_menu == 2:  
 cui.add\_testResult\_teacher(user)  
 **elif** teacher\_marks\_menu == 3:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** teacher\_menu == 5:  
 cui.view\_studentDetails\_teacher(user)  
 **elif** teacher\_menu == 6:  
 print(**"\nLogging Out..."**)  
 print(**"Logged Out!\n"**)  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **except** conn.errors.DataError:  
 print(**"Error! Invalid input!"**)  
 **except** ValueError:  
 print(**"Invalid input. Try again."**)  
  
 **elif** SignIn == **'Admin.valid'**:  
 **while True**:  
 print(**"""  
 \tDASHBOARD  
 1. Announcement  
 2. Assignment  
 3. Teacher Details  
 4. Student Details  
 5. Marks  
 6. Classes  
 7. Change Password  
 8. Log Out  
 """**)  
 **try**:  
 admin\_menu = int(input(**"Enter Menu Selection: "**))  
 **try**:  
 **if** admin\_menu == 1:  
 **while True**:  
 admin\_menu\_1 = **"""  
 Announcements  
 1) View announcements  
 2) Add announcement  
 3) Delete announcement  
 4) Return to Dashboard"""** print(admin\_menu\_1)  
 admin\_announcements\_menu = int(input(**"Enter selection: "**))  
 **if** admin\_announcements\_menu == 1:  
 cui.view\_announcements\_admin()  
 **elif** admin\_announcements\_menu == 2:  
 cui.add\_announcement\_teacher(12)  
 **elif** admin\_announcements\_menu == 3:  
 cui.del\_announcement\_teacher(12)  
 **elif** admin\_announcements\_menu == 4:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** admin\_menu == 2:  
 **while True**:  
 admin\_menu\_2 = **"""  
 Assignments  
 1) View assignments  
 2) Add assignment  
 3) Delete assignment  
 4) Return to Dashboard"""** print(admin\_menu\_2)  
 admin\_assignments\_menu = int(input(**"Enter selection: "**))  
 **if** admin\_assignments\_menu == 1:  
 cui.view\_assignment\_admin()  
 **elif** admin\_assignments\_menu == 2:  
 cui.add\_assignment\_teacher(12)  
 **elif** admin\_assignments\_menu == 3:  
 cui.del\_assignment\_teacher(12)  
 **elif** admin\_assignments\_menu == 4:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** admin\_menu == 3:  
 **while True**:  
 admin\_menu\_3 = **"""  
 Teacher Details  
 1) View Teacher Details  
 2) Add Teacher  
 3) Edit Teacher details  
 4) Remove Teacher  
 5) Return to Dashboard"""** print(admin\_menu\_3)  
 admin\_teacherDetails\_menu = int(input(**"Enter selection: "**))  
 **if** admin\_teacherDetails\_menu == 1:  
 cui.view\_teacherDetails\_admin()  
 **elif** admin\_teacherDetails\_menu == 2:  
 cui.add\_teacherDetails\_admin()  
 **elif** admin\_teacherDetails\_menu == 3:  
 cui.edit\_teacherProfile()  
 **elif** admin\_teacherDetails\_menu == 4:  
 cui.del\_teacherProfile()  
 **elif** admin\_teacherDetails\_menu == 5:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** admin\_menu == 4:  
 **while True**:  
 admin\_menu\_4 = **"""  
 Student Details  
 1) View Student Details  
 2) Add Student  
 3) Edit Student details  
 4) Remove Student  
 5) Return to Dashboard"""** print(admin\_menu\_4)  
 admin\_studentDetails\_menu = int(input(**"Enter selection: "**))  
 **if** admin\_studentDetails\_menu == 1:  
 cui.view\_studentDetails\_admin()  
 **elif** admin\_studentDetails\_menu == 2:  
 cui.add\_studentDetails\_admin()  
 **elif** admin\_studentDetails\_menu == 3:  
 cui.edit\_studentProfile\_admin()  
 **elif** admin\_studentDetails\_menu == 4:  
 cui.del\_studentProfile\_admin()  
 **elif** admin\_studentDetails\_menu == 5:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** admin\_menu == 5:  
 cui.view\_testResult\_admin()  
 **elif** admin\_menu == 6:  
 **while True**:  
 admin\_menu\_6 = **"""  
 Classes  
 1) View Classes  
 2) Add a Class  
 3) Remove Class  
 4) Return to Dashboard"""** print(admin\_menu\_6)  
 admin\_classes\_menu = int(input(**"Enter selection: "**))  
 **if** admin\_classes\_menu == 1:  
 cui.view\_classes\_admin()  
 **elif** admin\_classes\_menu == 2:  
 cui.add\_class\_admin()  
 **elif** admin\_classes\_menu == 3:  
 print(**"Delete Classes"**)  
 **elif** admin\_classes\_menu == 4:  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **elif** admin\_menu == 7:  
 old\_password = input(**"Enter old passward: "**) *# Accept old password  
  
 # Compare user entered password with original password (Check access)* **if** old\_password == Password:  
 new\_password = input(**"Enter new password: "**)  
 new\_password\_confirm = input(**"Confirm new password: "**)  
 **if** new\_password == new\_password\_confirm:  
 cui.edit\_profile\_all(SignIn, user, **'Password'**, new\_password)  
 print(**"Password succefully changed!"**)  
 **else**:  
 print(**"Error! The passwords do not match!"**)  
  
 **else**:  
 print(**"Error! Incorrect password!"**)  
 **elif** admin\_menu == 8:  
 print(**"\nLogging Out..."**)  
 print(**"Logged Out!\n"**)  
 **break  
 else**:  
 print(**"Invalid selection. Try again."**)  
 **except** conn.errors.DataError:  
 print(**"Error! Invalid input!"**)  
 **except** ValueError:  
 print(**"Invalid input. Try again."**)  
  
 **elif** SignIn == **'Error!'**:  
 print(**'Error! Exit program?\n(n/y)'**)  
 Exit = input()  
 **if** Exit.lower() **in** (**'y'**, **'yes'**):  
 exit(0)  
  
 **else**:  
 print(**"Error!"**)  
 print(**'Exit the program?\n(n/y)'**)  
 Exit = input()  
 **if** Exit.lower() **in** (**'y'**, **'yes'**):  
 exit(0)  
  
 print(**'Exit the program?\n(n/y)'**)  
 Exit = input()  
 **if** Exit.lower() **in** (**'y'**, **'yes'**):  
 exit(0)