Tables (11)

Name	Type	Schema
admin		CREATE TABLE admin (id INTEGER PRIMARY KEY AUTOINCREMENT, username TEXT, password TEXT, fullName TEXT, position TEXT)
id	INTEGER	"id" INTEGER
username	TEXT	"username" TEXT
password	TEXT	"password" TEXT
fullName	TEXT	"fullName" TEXT
position	TEXT	"position" TEXT
assignment		CREATE TABLE "assignment" ("id" INTEGER NOT NULL, "dateOfAssignment" TEXT, "dateOfSubmission" TEXT, "dateOfDeadline" TEXT, "packageID" INTEGER UNIQUE, "personID" INT UNIQUE, PRIMARY KEY("id" AUTOINCREMENT), FOREIGN KEY("personID") REFERENCES "person"("id"), FOREIGN KEY("packageID") REFERENCES "package"("id"))
id	INTEGER	"id" INTEGER NOT NULL
dateOfAssignment	TEXT	"dateOfAssignment" TEXT
dateOfSubmission	TEXT	"dateOfSubmission" TEXT
dateOfDeadline	TEXT	"dateOfDeadline" TEXT
packageID	INTEGER	"packageID" INTEGER UNIQUE
personID	INT	"personID" INT UNIQUE
college		CREATE TABLE "college" ("id" INTEGER, "collegeName" TEXT NOT NULL, PRIMARY KEY("id" AUTOINCREMENT))
id	INTEGER	"id" INTEGER
collegeName	TEXT	"collegeName" TEXT NOT NULL
current_session		CREATE TABLE "current_session" ("id" INTEGER NOT NULL, "sessionName" TEXT NOT NULL, PRIMARY KEY("id" AUTOINCREMENT))
id	INTEGER	"id" INTEGER NOT NULL
sessionName	TEXT	"sessionName" TEXT NOT NULL
department		CREATE TABLE "department" ("id" INTEGER NOT NULL, "departmentName" TEXT, PRIMARY KEY("id" AUTOINCREMENT))
id	INTEGER	"id" INTEGER NOT NULL

Name	Type	Schema
departmentName	TEXT	"departmentName" TEXT
exam		`CREATE TABLE "exam" ( "id" INTEGER NOT NULL, "subjectID" INTEGER NOT NULL, "examType" TEXT NOT NULL DEFAULT 'Regular' CHECK("examType" IN ('Regular', 'Back')), "date" TEXT, FOREIGN KEY("subjectID") REFERENCES "subject"("id"), PRIMARY KEY("id" AUTOINCREMENT) )`
id	INTEGER	"id" INTEGER NOT NULL
subjectID	INTEGER	"subjectID" INTEGER NOT NULL
examType	TEXT	"examType" TEXT NOT NULL DEFAULT 'Regular' CHECK("examType" IN ('Regular', 'Back'))
date	TEXT	"date" TEXT
package		`CREATE TABLE "package" ("id" INTEGER NOT NULL, "packageCode" TEXT UNIQUE, "noOfCopies" INT, "codeStart" TEXT, "codeEnd" TEXT, "examID" INT, "status" TEXT DEFAULT 'Not Assigned' CHECK("status" IN ('Not Assigned', 'Pending', 'Submitted')), FOREIGN KEY("examID") REFERENCES "exam"("id"), PRIMARY KEY("id" AUTOINCREMENT))`
id	INTEGER	"id" INTEGER NOT NULL
packageCode	TEXT	"packageCode" TEXT UNIQUE
noOfCopies	INT	"noOfCopies" INT
codeStart	TEXT	"codeStart" TEXT
codeEnd	TEXT	"codeEnd" TEXT
examID	INT	"examID" INT
status	TEXT	"status" TEXT DEFAULT 'Not Assigned' CHECK("status" IN ('Not Assigned', 'Pending', 'Submitted'))
person		`CREATE TABLE "person" ( "id" INTEGER NOT NULL, "fullName" TEXT NOT NULL, "contact" TEXT NOT NULL, "email" TEXT, "collegeID" INTEGER NOT NULL, PRIMARY KEY("id" AUTOINCREMENT), FOREIGN KEY("collegeID") REFERENCES "college"("id") )`
id	INTEGER	"id" INTEGER NOT NULL
fullName	TEXT	"fullName" TEXT NOT NULL
contact	TEXT	"contact" TEXT NOT NULL
email	TEXT	"email" TEXT

Name	Type	Schema
collegeID	INTEGER	"collegeID" INTEGER NOT NULL
program		`CREATE TABLE program ( id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, programName TEXT NOT NULL, academicDegree TEXT CHECK(academicDegree IN ('Bachelors', 'Masters')) NOT NULL, departmentID INT NOT NULL, FOREIGN KEY(departmentID) REFERENCES department(id) )`
id	INTEGER	"id" INTEGER NOT NULL
programName	TEXT	"programName" TEXT NOT NULL
academicDegree	TEXT	"academicDegree" TEXT NOT NULL CHECK("academicDegree" IN ('Bachelors', 'Masters'))
departmentID	INT	"departmentID" INT NOT NULL
sqlite_sequence		`CREATE TABLE sqlite_sequence(name,seq)`
name		"name"
seq		"seq"
subject		`CREATE TABLE "subject" ("id" INTEGER NOT NULL, "courseCode" TEXT NOT NULL, "year" TEXT CHECK("year" IN ('I', 'II', 'III', 'IV')), "part" TEXT CHECK("year" IN ('I', 'II')), "programID" INT NOT NULL, "subjectName" TEXT, FOREIGN KEY("programID") REFERENCES "program"("id"), PRIMARY KEY("id" AUTOINCREMENT))`
id	INTEGER	"id" INTEGER NOT NULL
courseCode	TEXT	"courseCode" TEXT NOT NULL
year	TEXT	"year" TEXT CHECK("year" IN ('I', 'II', 'III', 'IV'))
part	TEXT	"part" TEXT CHECK("year" IN ('I', 'II'))
programID	INT	"programID" INT NOT NULL
subjectName	TEXT	"subjectName" TEXT

Indices (0)

Name	Type	Schema

Views (0)

Name	Type	Schema

Triggers (0)

Name	Type	Schema