

بہ نام خدا کہ یکتا

گزارش تمرین چہارم درس پایگاہ دادہ پیشرفتہ

نام و نام خانوادگی :

مبین رمضانے خرم آبادک 40311415016

نام استاد :

آقاک دکتر آرمین رشنو

(۱) یک پایگاه داده به نام **LorestanUniv** ایجاد کنید.

```
test> use LorestanUniv;
switched to db LorestanUniv
LorestanUniv>
```

به صورت خودکار در زمان ایجاد دیتابیس، به آن متصل می‌شود.

(۲) جداول **student**، **lecturer** و **course** را در پایگاه داده براساس فیلدهای مشخص شده در جداول فوق ایجاد کنید.

ایجاد جدول **student**:

```
db.createCollection("student", {validator: {$jsonSchema: {bsonType: "object",required: ["STID", "FName", "LName", "Father", "Birth", "SerialID", "BornCity", "PostalCode", "CPhone", "HPhone", "Department", "Major", "Married", "ID"],properties: {STID: {bsonType: "string",pattern: "^[0-9]{3}114150[0-9]{2}$"},FName: {bsonType: "string",maxLength: 50},LName: {bsonType: "string",maxLength: 50},Father: {bsonType: "string",maxLength: 50},Birth: {bsonType: "string",pattern: "^[0-9]{4}/[0-9]{2}/[0-9]{2}$"},SerialID: {bsonType: "string",maxLength: 9},BornCity: {bsonType: "string",enum: ["Tehran", "Mashhad", "Isfahan", "Shiraz", "Tabriz", "Ahvaz", "Kermanshah", "Urmia", "Rasht", "Zahedan","Kerman", "Yazd", "Hamadan", "Arak", "Ardabil", "Bandar Abbas", "Khorramabad", "Bojnord", "Semnan","Qazvin", "Sanandaj", "Shahrekord", "Gorgan", "Ilam", "Zanjan"]},Address: {bsonType: "string",maxLength: 200},PostalCode: {bsonType: "string",pattern: "^[0-9]{10}$"},CPhone: {bsonType: "string",pattern: "^09[0-9]{9}$"},HPhone: {bsonType: "string",pattern: "^[0-9]{11}$"},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics","Agriculture and Natural Resources", "Veterinary Medicine"]},Major: {bsonType: "string",enum: ["Electrical Engineering-Electronics", "Electrical Engineering-Power","Computer Engineering", "Civil Engineering", "Mining Engineering","Mechanical Engineering", "Polymer Engineering"]},Married: {bsonType: "bool"},ID: {bsonType: "string",pattern: "^[0-9]{10}$"}}}});
```

```
LorestanUniv> db.createCollection("student", {validator: {$jsonSchema: {bsonType: "object",required: ["STID", "FName", "LName", "Father", "Birth", "SerialID", "BornCity", "PostalCode", "CPhone", "HPhone", "Department", "Major", "Married", "ID"],properties: {STID: {bsonType: "string",pattern: "^[0-9]{3}114150[0-9]{2}$"},FName: {bsonType: "string",maxLength: 50},LName: {bsonType: "string",maxLength: 50},Father: {bsonType: "string",maxLength: 50},Birth: {bsonType: "string",pattern: "^[0-9]{4}/[0-9]{2}/[0-9]{2}$"},SerialID: {bsonType: "string",maxLength: 9},BornCity: {bsonType: "string",enum: ["Tehran", "Mashhad", "Isfahan", "Shiraz", "Tabriz", "Ahvaz", "Kermanshah", "Urmia", "Rasht", "Zahedan","Kerman", "Yazd", "Hamadan", "Arak", "Ardabil", "Bandar Abbas", "Khorramabad", "Bojnord", "Semnan","Qazvin", "Sanandaj", "Shahrekord", "Gorgan", "Ilam", "Zanjan"]},Address: {bsonType: "string",maxLength: 200},PostalCode: {bsonType: "string",pattern: "^[0-9]{10}$"},CPhone: {bsonType: "string",pattern: "^09[0-9]{9}$"},HPhone: {bsonType: "string",pattern: "^[0-9]{11}$"},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics", "Agriculture and Natural Resources", "Veterinary Medicine"]},Major: {bsonType: "string",enum: ["Electrical Engineering-Electronics", "Electrical Engineering-Power", "Computer Engineering", "Civil Engineering", "Mining Engineering", "Mechanical Engineering", "Polymer Engineering"]},Married: {bsonType: "bool"},ID: {bsonType: "string",pattern: "^[0-9]{10}$"}}}});
{ ok: 1 }
LorestanUniv> show collections;
student
```

سپس برای ایجاد کلید اصلی جدول (STID) از دستور زیر استفاده میکنیم:

```
LorestanUniv> db.student.createIndex({ STID: 1 }, { unique: true });
STID_1
```

: ایجاد جدول lecturer

```
db.createCollection("lecturer", {validator: {$jsonSchema: {bsonType: "object",required: ["LID", "FName", "LName", "ID", "Department", "Major", "Birth", "BornCity", "PostalCode", "CPhone", "HPhone"],properties: {LID: {bsonType: "string",pattern: "^([0-9]{6})$"},FName: {bsonType: "string",maxLength: 50},LName: {bsonType: "string",maxLength: 50},ID: {bsonType: "string",pattern: "^([0-9]{10})$"},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics", "Agriculture and Natural Resources", "Veterinary Medicine"]},Major: {bsonType: "string",enum: ["Electrical Engineering-Electronics", "Electrical Engineering-Power", "Computer Engineering", "Civil Engineering", "Mining Engineering", "Mechanical Engineering", "Polymer Engineering"]},Birth: {bsonType: "string",pattern: "^([0-9]{4})/([0-9]{2})/([0-9]{2})$"},BornCity: {bsonType: "string",enum: ["Tehran", "Mashhad", "Isfahan", "Shiraz", "Tabriz", "Ahvaz", "Kermanshah", "Urmia", "Rasht", "Zahedan", "Kerman", "Yazd", "Hamadan", "Arak", "Ardabil", "Bandar Abbas", "Khorramabad", "Bojnord", "Semnan", "Qazvin", "Sanandaj", "Shahrekord", "Gorgan", "Ilam", "Zanjan"]},Address: {bsonType: "string",maxLength: 200},PostalCode: {bsonType: "string",pattern: "^([0-9]{10})$"},CPhone: {bsonType: "string",pattern: "^09([0-9]{9})$"},HPhone: {bsonType: "string",pattern: "^([0-9]{11})$"}}}}});
```

```
LoRESTanUniv> db.createCollection("lecturer", {validator: {$jsonSchema: {bsonType: "object",required: ["LID", "FName", "LName", "ID", "Department", "Major", "Birth", "BornCity", "PostalCode", "CPhone", "HPhone"],properties: {LID: {bsonType: "string",pattern: "^([0-9]{6})$"},FName: {bsonType: "string",maxLength: 50},LName: {bsonType: "string",maxLength: 50},ID: {bsonType: "string",pattern: "^([0-9]{10})$"},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics", "Agriculture and Natural Resources", "Veterinary Medicine"]},Major: {bsonType: "string",enum: ["Electrical Engineering-Electronics", "Electrical Engineering-Power", "Computer Engineering", "Civil Engineering", "Mining Engineering", "Mechanical Engineering", "Polymer Engineering"]},Birth: {bsonType: "string",pattern: "^([0-9]{4})/([0-9]{2})/([0-9]{2})$"},BornCity: {bsonType: "string",enum: ["Tehran", "Mashhad", "Isfahan", "Shiraz", "Tabriz", "Ahvaz", "Kermanshah", "Urmia", "Rasht", "Zahedan", "Kerman", "Yazd", "Hamadan", "Arak", "Ardabil", "Bandar Abbas", "Khorramabad", "Bojnord", "Semnan", "Qazvin", "Sanandaj", "Shahrekord", "Gorgan", "Ilam", "Zanjan"]},Address: {bsonType: "string",maxLength: 200},PostalCode: {bsonType: "string",pattern: "^([0-9]{10})$"},CPhone: {bsonType: "string",pattern: "^09([0-9]{9})$"},HPhone: {bsonType: "string",pattern: "^([0-9]{11})$"}}}}});
{ ok: 1 }
LoRESTanUniv> show collections
lecturer
student
```

سپس برای ایجاد کلید اصلی جدول (LID) از دستور زیر استفاده میکنیم :

```
LoRESTanUniv> db.lecturer.createIndex({ LID: 1 }, { unique: true });
LID_1
```

: ایجاد جدول course

```
db.createCollection("course", {validator: {$jsonSchema: {bsonType: "object",required: ["CID", "CName", "Department", "Credit"],properties: {CID: {bsonType: "string",pattern: "^([0-9]{5})$"},CName: {bsonType: "string",maxLength: 25},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics", "Agriculture and Natural Resources", "Veterinary Medicine"]},Credit: {bsonType: "int",minimum: 1,maximum: 4}}}}});
```

```
LoRESTanUniv> db.createCollection("course", {validator: {$jsonSchema: {bsonType: "object",required: ["CID", "CName", "Department", "Credit"],properties: {CID: {bsonType: "string",pattern: "^([0-9]{5})$"},CName: {bsonType: "string",maxLength: 25},Department: {bsonType: "string",enum: ["Basic Sciences", "Engineering", "Literature and Humanities", "Economics", "Agriculture and Natural Resources", "Veterinary Medicine"]},Credit: {bsonType: "int",minimum: 1,maximum: 4}}}}});
{ ok: 1 }
LoRESTanUniv> show collections
course
lecturer
student
```


سپس برای ایجاد کلید اصلی جدول (CID) از دستور زیر استفاده میکنیم :

```
LoRESTanUniv> db.course.createIndex({ CID: 1 }, { unique: true });
CID_1
```

جدول انتخاب واحد CourseRegister: (دارای چند مرحله)

```
db.createCollection("courseRegister", {validator: {$jsonSchema: {bsonType: "object",required: ["SID", "CID"],properties: {SID: {bsonType: "string",pattern: "^([0-9]{3})114150([0-9]{2})$"},CID: {bsonType: "string",pattern: "^([0-9]{5})$"}}}}});
```

```
LoRESTanUniv> db.createCollection("courseRegister", {validator: {$jsonSchema: {bsonType: "object",required: ["SID", "CID"],properties: {SID: {bsonType: "string",pattern: "^([0-9]{3})114150([0-9]{2})$"},CID: {bsonType: "string",pattern: "^([0-9]{5})$"}}}}});
{ ok: 1 }
LoRESTanUniv> show collections
course
courseRegister
lecturer
student
```



سپس برای ایجاد کلیدهای اصلی جدول (STID و CID) از دستور زیر استفاده میکنیم:

```
LoRESTanUniv> db.courseRegister.createIndex({ SID: 1, CID: 1 }, { unique: true });
SID_1_CID_1
```

اما چون جدول ما شامل ستون های دیگر هم هست، آن ها را با توجه به کلیدهای اصلی و از طریق متد **aggregate** این کار را به صورت زیر انجام می دهیم:


```
db.courseRegister.aggregate([{$lookup: {from: "student",localField: "SID",foreignField: "STID",as: "studentDetails"}},{lookup: {from: "course",localField: "CID",foreignField: "CID",as: "courseDetails"}},{project: { _id: 0, FName: { $arrayElemAt: ["$studentDetails.FName", 0] },LName: { $arrayElemAt: ["$studentDetails.LName", 0] },CName: { $arrayElemAt: ["$courseDetails.CName", 0] },Department: { $arrayElemAt: ["$courseDetails.Department", 0] },Credit: { $arrayElemAt: ["$courseDetails.Credit", 0] }}}];
```

```
LoRESTanUniv> db.courseRegister.aggregate([{$lookup: {from: "student",localField: "SID",foreignField: "STID",as: "studentDetails"}},{lookup: {from: "course",localField: "CID",foreignField: "CID",as: "courseDetails"}},{project: { _id: 0, FName: { $arrayElemAt: ["$studentDetails.FName", 0] },LName: { $arrayElemAt: ["$studentDetails.LName", 0] },CName: { $arrayElemAt: ["$courseDetails.CName", 0] },Department: { $arrayElemAt: ["$courseDetails.Department", 0] },Credit: { $arrayElemAt: ["$courseDetails.Credit", 0] }}}];
```

جدول انتخاب واحد PresentedCourses: (دارای چند مرحله)

```
db.createCollection("presentedCourses", {validator: {$jsonSchema: {bsonType: "object",required: ["CID", "LID"],properties: {CID: {bsonType: "string",pattern: "^([0-9]{5})$"},LID: {bsonType: "string",pattern: "^([0-9]{6})$"}}}}});
```

```
LoRESTanUniv> db.createCollection("presentedCourses", {validator: {$jsonSchema: {bsonType: "object",required: ["CID", "LID"],properties: {CID: {bsonType: "string",pattern: "^([0-9]{5})$"},LID: {bsonType: "string",pattern: "^([0-9]{6})$"}}}}});
{ ok: 1 }
LoRESTanUniv> show collections
course
courseRegister
lecturer
presentedCourses
student
```



سپس برای ایجاد کلیدهای اصلی جدول (LID و CID) از دستور زیر استفاده میکنیم :

```
LoRESTanUniv> db.presentedCourses.createIndex({ CID: 1, LID: 1 }, { unique: true });
CID_1_LID_1
```

اما چون جدول ما شامل ستون های دیگر هم هست، آن ها را با توجه به کلیدهای اصلی و از طریق متد **aggregate** این کار را به صورت زیر انجام می دهیم :

```
db.presentedCourses.aggregate([{$lookup: {from: "course",localField: "CID",foreignField: "CID",as: "courseDetails"}},{$lookup: {from: "lecturer",localField: "LID",foreignField: "LID",as: "lecturerDetails"}},{$project: { _id: 0,CName: {$arrayElemAt: ["$courseDetails.CName", 0] },Department: {$arrayElemAt: ["$courseDetails.Department", 0] },Credit: {$arrayElemAt: ["$courseDetails.Credit", 0] },FName: {$arrayElemAt: ["$lecturerDetails.FName", 0] },LName: {$arrayElemAt: ["$lecturerDetails.LName", 0] }}}];
```

```
LoRESTanUniv> db.presentedCourses.aggregate([{$lookup: {from: "course",localField: "CID",foreignField: "CID",as: "courseDetails"}},{$lookup: {from: "lecturer",localField: "LID",foreignField: "LID",as: "lecturerDetails"}},{$project: { _id: 0,CName: {$arrayElemAt: ["$courseDetails.CName", 0] },Department: {$arrayElemAt: ["$courseDetails.Department", 0] },Credit: {$arrayElemAt: ["$courseDetails.Credit", 0] },FName: {$arrayElemAt: ["$lecturerDetails.FName", 0] },LName: {$arrayElemAt: ["$lecturerDetails.LName", 0] }}}];
```

۳) عمل درج رکورد در هر جدول را انجام دهید به گونه ای که اعتبارسنجی روی همه فیلدها وجود داشته باشد و مقادیر غیرمجاز را برای درج قبول نکند.

درج سه رکورد در جدول **student** :

```
db.student.insertMany([{$STID: "40311415016",FName: "Mobin",LName: "Ramezani",Father: "Shahram",Birth: "1381/04/11",SerialID: "12b145844",BornCity: "Khorramabad",Address: "lorestan-----123 St.",PostalCode: "6814617137",CPhone: "09121234567",HPhone: "02112345678",Department: "Engineering",Major: "Computer Engineering",Married: false,ID: "1234567890"},{$STID: "40311415071",FName: "Sara",LName: "Rezaei",Father: "Mohammad",Birth: "1382/07/20",SerialID: "987654321",BornCity: "Shiraz",Address: "shiraz-----456 St.",PostalCode: "9876543210",CPhone: "09123456789",HPhone: "03123456789",Department: "Economics",Major: "Polymer Engineering",Married: false,ID: "0987654321"},{$STID: "40311415052",FName: "Reza",LName: "Karimi",Father: "Ali",Birth: "1382/09/10",SerialID: "741852963",BornCity: "Mashhad",Address: "mashad-----789 St.",PostalCode: "7418529630",CPhone: "09187654321",HPhone: "05112345678",Department: "Basic Sciences",Major: "Electrical Engineering-Electronics",Married: true,ID: "1122334455"}]);
```

```
LoRESTanUniv> db.student.insertMany([{$STID: "40311415016",FName: "Mobin",LName: "Ramezani",Father: "Shahram",Birth: "1381/04/11",SerialID: "12b145844",BornCity: "Khorramabad",Address: "lorestan-----123 St.",PostalCode: "6814617137",CPhone: "09121234567",HPhone: "02112345678",Department: "Engineering",Major: "Computer Engineering",Married: false,ID: "1234567890"},{$STID: "40311415071",FName: "Sara",LName: "Rezaei",Father: "Mohammad",Birth: "1382/07/20",SerialID: "987654321",BornCity: "Shiraz",Address: "shiraz-----456 St.",PostalCode: "9876543210",CPhone: "09123456789",HPhone: "03123456789",Department: "Economics",Major: "Polymer Engineering",Married: false,ID: "0987654321"},{$STID: "40311415052",FName: "Reza",LName: "Karimi",Father: "Ali",Birth: "1382/09/10",SerialID: "741852963",BornCity: "Mashhad",Address: "mashad-----789 St.",PostalCode: "7418529630",CPhone: "09187654321",HPhone: "05112345678",Department: "Basic Sciences",Major: "Electrical Engineering-Electronics",Married: true,ID: "1122334455"}]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67484ff0a96ed4cf290d8190'),
    '1': ObjectId('67484ff0a96ed4cf290d8191'),
    '2': ObjectId('67484ff0a96ed4cf290d8192')
  }
}
```

مشاهده اطلاعات درج شده در جدول :

```

LorestanUniv> db.student.find().pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8190'),
    STID: '40311415016',
    FName: 'Mobin',
    LName: 'Ramezani',
    Father: 'Shahram',
    Birth: '1381/04/11',
    SerialID: '12b145844',
    BornCity: 'Khorramabad',
    Address: 'lorestan-----123 St.',
    PostalCode: '6814617137',
    CPhone: '09121234567',
    HPhone: '02112345678',
    Department: 'Engineering',
    Major: 'Computer Engineering',
    Married: false,
    ID: '1234567890'
  },
  {
    _id: ObjectId('67484ff0a96ed4cf290d8191'),
    STID: '40311415071',
    FName: 'Sara',
    LName: 'Rezaei',
    Father: 'Mohammad',
    Birth: '1382/07/20',
    SerialID: '987654321',
    BornCity: 'Shiraz',
    Address: 'shiraz-----456 St.',
    PostalCode: '9876543210',
    CPhone: '09123456789',
    HPhone: '03123456789',
    Department: 'Economics',
    Major: 'Polymer Engineering',
    Married: false,
    ID: '0987654321'
  },
  {
    _id: ObjectId('67484ff0a96ed4cf290d8192'),
    STID: '40311415052',
    FName: 'Reza',
    LName: 'Karimi',
    Father: 'Ali',
    Birth: '1382/09/10',
    SerialID: '741852963',
    BornCity: 'Mashhad',
    Address: 'mashad-----789 St.',
    PostalCode: '7418529630',
    CPhone: '09187654321',
    HPhone: '05112345678',
    Department: 'Basic Sciences',
    Major: 'Electrical Engineering-Electronics',
    Married: true,
    ID: '1122334455'
  }
]

```

درج سه رکورد در جدول Lecturer :

```

db.lecturer.insertMany([
  {LID: "123456",FName: "Reza",LName: "Hosseini",ID: "4785136548",Department: "Engineering",Major: "Computer Engineering",Birth: "1372/03/10",BornCity: "Ilam",Address: "ilam-----123 St.",PostalCode: "7257567890",CPhone: "09121234567",HPhone: "02112345678"},
  {LID: "654321",FName: "Mina",LName: "Rastegar",ID: "1548741215",Department: "Economics",Major: "Polymer Engineering",Birth: "1368/06/25",BornCity: "Tehran",Address: "tehran-----456 St.",PostalCode: "9876773120",CPhone: "09123456789",HPhone: "02187654321"},
  {LID: "789012",FName: "Ali",LName: "Shahbazi",ID: "6523526896",Department: "Basic Sciences",Major: "Electrical Engineering-Electronics",Birth: "1353/09/15",BornCity: "Isfahan",Address: "isfahan-----789 St.",PostalCode: "7418529630",CPhone: "09187654321",HPhone: "03112345678"}
]);

```

```

LorestanUniv> db.lecturer.insertMany([
  {LID: "123456", FName: "Reza", LName: "Hosseini", ID: "4785136548", Department: "Engineering", Major: "Computer Engineering", Birth: "1372/03/10", BornCity: "Ilam", Address: "ilam-----123 St.", PostalCode: "7257567890", CPhone: "09121234567", HPhone: "02112345678"},
  {LID: "654321", FName: "Mina", LName: "Rastegar", ID: "1548741215", Department: "Economics", Major: "Polymer Engineering", Birth: "1368/06/25", BornCity: "Tehran", Address: "tehran-----456 St.", PostalCode: "9876773120", CPhone: "09123456789", HPhone: "02187654321"},
  {LID: "789012", FName: "Ali", LName: "Shahbazi", ID: "6523526896", Department: "Basic Sciences", Major: "Electrical Engineering-Electronics", Birth: "1353/09/15", BornCity: "Isfahan", Address: "isfahan-----789 St.", PostalCode: "7418529630", CPhone: "09187654321", HPhone: "03112345678"}
]);

```

```

{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67485269a96ed4cf290d8193'),
    '1': ObjectId('67485269a96ed4cf290d8194'),
    '2': ObjectId('67485269a96ed4cf290d8195')
  }
}

```

مشاهده اطلاعات درج شده در جدول :

```

LorestanUniv> db.lecturer.find().pretty();
[
  {
    _id: ObjectId('67485269a96ed4cf290d8193'),
    LID: '123456',
    FName: 'Reza',
    LName: 'Hosseini',
    ID: '4785136548',
    Department: 'Engineering',
    Major: 'Computer Engineering',
    Birth: '1372/03/10',
    BornCity: 'Ilam',
    Address: 'ilam-----123 St.',
    PostalCode: '7257567890',
    CPhone: '09121234567',
    HPhone: '02112345678'
  },
  {
    _id: ObjectId('67485269a96ed4cf290d8194'),
    LID: '654321',
    FName: 'Mina',
    LName: 'Rastegar',
    ID: '1548741215',
    Department: 'Economics',
    Major: 'Polymer Engineering',
    Birth: '1368/06/25',
    BornCity: 'Tehran',
    Address: 'tehran-----456 St.',
    PostalCode: '9876773120',
    CPhone: '09123456789',
    HPhone: '02187654321'
  },
  {
    _id: ObjectId('67485269a96ed4cf290d8195'),
    LID: '789012',
    FName: 'Ali',
    LName: 'Shahbazi',
    ID: '6523526896',
    Department: 'Basic Sciences',
    Major: 'Electrical Engineering-Electronics',
    Birth: '1353/09/15',
    BornCity: 'Isfahan',
    Address: 'isfahan-----789 St.',
    PostalCode: '7418529630',
    CPhone: '09187654321',
    HPhone: '03112345678'
  }
]

```

درج سه رکورد در جدول **course** :

```
db.course.insertMany([{CID: "12345", CName: "Database", Department: "Engineering", Credit: 3},{CID: "23456", CName: "Microeconomic", Department: "Economics", Credit: 4},{CID: "34567", CName: "Advanced Physic", Department: "Basic Sciences", Credit: 2}]);
```

```
LoRESTanUniv> db.course.insertMany([{CID: "12345", CName: "Database", Department: "Engineering", Credit: 3},{CID: "23456", CName: "Microeconomic", Department: "Economics", Credit: 4},{CID: "34567", CName: "Advanced Physic", Department: "Basic Sciences", Credit: 2}]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67485f6ea96ed4cf290d8196'),
    '1': ObjectId('67485f6ea96ed4cf290d8197'),
    '2': ObjectId('67485f6ea96ed4cf290d8198')
  }
}
```

مشاهده اطلاعات درج شده در جدول :

```
LoRESTanUniv> db.course.find().pretty();
```

```
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8196'),
    CID: '12345',
    CName: 'Database',
    Department: 'Engineering',
    Credit: 3
  },
  {
    _id: ObjectId('67485f6ea96ed4cf290d8197'),
    CID: '23456',
    CName: 'Microeconomic',
    Department: 'Economics',
    Credit: 4
  },
  {
    _id: ObjectId('67485f6ea96ed4cf290d8198'),
    CID: '34567',
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 2
  }
]
```

درج سه رکورد در جدول **CourseRegister** :

```
db.CourseRegister.insertMany([{Credit: 3,Department: "Engineering",CName: "Database",FName: "Mobin",LName: "Ramezani",STID: "40311415016",CID: "12345"},{Credit: 4,Department: "Economics",CName: "Microeconomic",FName: "Sara",LName: "Rezaei",STID: "40311415071",CID: "23456"},{Credit: 2,Department: "Basic Sciences",CName: "Advanced Physic",FName: "Reza",LName: "Karimi",STID: "40311415052",CID: "34567"}]);
```

```
LoRESTanUniv> db.CourseRegister.insertMany([{Credit: 3,Department: "Engineering",CName: "Database",FName: "Mobin",LName: "Ramezani",STID: "40311415016",CID: "12345"},{Credit: 4,Department: "Economics",CName: "Microeconomic",FName: "Sara",LName: "Rezaei",STID: "40311415071",CID: "23456"},{Credit: 2,Department: "Basic Sciences",CName: "Advanced Physic",FName: "Reza",LName: "Karimi",STID: "40311415052",CID: "34567"}]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67486365a96ed4cf290d8199'),
    '1': ObjectId('67486365a96ed4cf290d819a'),
    '2': ObjectId('67486365a96ed4cf290d819b')
  }
}
```


مشاهده اطلاعات درج شده در جدول :

```
LoRESTanUniv> db.CourseRegister.find().pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d8199'),
    Credit: 3,
    Department: 'Engineering',
    CName: 'Database',
    FName: 'Mobin',
    LName: 'Ramezani',
    STID: '40311415016',
    CID: '12345'
  },
  {
    _id: ObjectId('67486365a96ed4cf290d819a'),
    Credit: 4,
    Department: 'Economics',
    CName: 'Microeconomic',
    FName: 'Sara',
    LName: 'Rezaei',
    STID: '40311415071',
    CID: '23456'
  },
  {
    _id: ObjectId('67486365a96ed4cf290d819b'),
    Credit: 2,
    Department: 'Basic Sciences',
    CName: 'Advanced Physic',
    FName: 'Reza',
    LName: 'Karimi',
    STID: '40311415052',
    CID: '34567'
  }
]
```

درج سه رکورد در جدول PresentedCourses :

```
db.PresentedCourses.insertMany([{CName: "Database",Department: "Engineering",Credit: 3,FName: "Reza",LName: "Hosseini",CID: "12345",LID: "123456"},{CName: "Microeconomic",Department: "Economics",Credit: 4,FName: "Mina",LName: "Rastegar",CID: "23456",LID: "654321"},{CName: "Advanced Physic",Department: "Basic Sciences",Credit: 2,FName: "Ali",LName: "Shahbazi",CID: "34567",LID: "789012"}]);
```

```
LoRESTanUniv> db.PresentedCourses.insertMany([{CName: "Database",Department: "Engineering",Credit: 3,FName: "Reza",LName: "Hosseini",CID: "12345",LID: "123456"},{CName: "Microeconomic",Department: "Economics",Credit: 4,FName: "Mina",LName: "Rastegar",CID: "23456",LID: "654321"},{CName: "Advanced Physic",Department: "Basic Sciences",Credit: 2,FName: "Ali",LName: "Shahbazi",CID: "34567",LID: "789012"}]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6748664aa96ed4cf290d819c'),
    '1': ObjectId('6748664aa96ed4cf290d819d'),
    '2': ObjectId('6748664aa96ed4cf290d819e')
  }
}
```

مشاهده اطلاعات درج شده در جدول :

```
LoRESTanUniv> db.PresentedCourses.find().pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819c'),
    CName: 'Database',
    Department: 'Engineering',
    Credit: 3,
    FName: 'Reza',
    LName: 'Hosseini',
    CID: '12345',
    LID: '123456'
  },
  {
    _id: ObjectId('6748664aa96ed4cf290d819d'),
    CName: 'Microeconomic',
    Department: 'Economics',
    Credit: 4,
    FName: 'Mina',
    LName: 'Rastegar',
    CID: '23456',
    LID: '654321'
  },
  {
    _id: ObjectId('6748664aa96ed4cf290d819e'),
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 2,
    FName: 'Ali',
    LName: 'Shahbazi',
    CID: '34567',
    LID: '789012'
  }
]
```

۴) عملیات حذف و بروزرسانی را نیز برای ۳ مورد از هر جدول تست کنید.

در جدول student :

آپدیت ها :

تغییر نام خانوادگی و آدرس دانشجوی با 'STID='40311415016'

```
db.student.updateOne({ STID: "40311415016" }, {$set: {LName: "Mohammadi", Address: "Lorestan-----New Address"}});
```

```
LorestanUniv> db.student.updateOne({ STID: "40311415016" }, {$set: {LName: "Mohammadi", Address: "Lorestan-----New Address"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LorestanUniv> db.student.find({ STID: "40311415016" }).pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8190'),
    STID: '40311415016',
    FName: 'Mobin',
    LName: 'Mohammadi',
    Father: 'Shahram',
    Birth: '1381/04/11',
    SerialID: '12b145844',
    BornCity: 'Khorramabad',
    Address: 'Lorestan-----New Address',
    PostalCode: '6814617137',
    CPhone: '09121234567',
    HPhone: '02112345678',
    Department: 'Engineering',
    Major: 'Computer Engineering',
    Married: false,
    ID: '1234567890'
  }
]
```

تغییر رشته و وضعیت تاهل دانشجوی با 'STID='40311415071'

```
db.student.updateOne({ STID: "40311415071" }, {$set: {Major: "Electrical Engineering-Power", Married: true}});
```

```
LorestanUniv> db.student.updateOne({ STID: "40311415071" }, {$set: {Major: "Electrical Engineering-Power", Married: true}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LorestanUniv> db.student.find({ STID: "40311415071" }).pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8191'),
    STID: '40311415071',
    FName: 'Sara',
    LName: 'Rezaei',
    Father: 'Mohammad',
    Birth: '1382/07/20',
    SerialID: '987654321',
    BornCity: 'Shiraz',
    Address: 'Shiraz-----456 St.',
    PostalCode: '9876543210',
    CPhone: '09123456789',
    HPhone: '03123456789',
    Department: 'Economics',
    Major: 'Electrical Engineering-Power',
    Married: true,
    ID: '0987654321'
  }
]
```

تغییر شماره تماس و کد پستی دانشجوی با 'STID='40311415052'

```
db.student.updateOne({ STID: "40311415052" }, {$set: {CPhone: "09121111111", PostalCode: "1111111111"}});
```

```
LoRESTanUniv> db.student.updateOne({ STID: "40311415052" }, {$set: {CPhone: "09121111111", PostalCode: "1111111111"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.student.find({ STID: "40311415052" }).pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8192'),
    STID: '40311415052',
    FName: 'Reza',
    LName: 'Karimi',
    Father: 'Ali',
    Birth: '1382/09/10',
    SerialID: '741852963',
    BornCity: 'Mashhad',
    Address: 'mashad-----789 St.',
    PostalCode: '1111111111',
    CPhone: '09121111111',
    HPhone: '05112345678',
    Department: 'Basic Sciences',
    Major: 'Electrical Engineering-Electronics',
    Married: true,
    ID: '1122334455'
  }
]
```

حذف ها :

حذف دانشجوی با 'STID='40311415016'

```
db.student.deleteOne({STID: "40311415016"});
```

```
LoRESTanUniv> db.student.deleteOne({STID: "40311415016"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.student.find().pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8191'),
    STID: '40311415071',
    FName: 'Sara',
    LName: 'Rezaei',
    Father: 'Mohammad',
    Birth: '1382/07/20',
    SerialID: '987654321',
    BornCity: 'Shiraz',
    Address: 'shiraz-----456 St.',
    PostalCode: '9876543210',
    CPhone: '09123456789',
    HPhone: '03123456789',
    Department: 'Economics',
    Major: 'Electrical Engineering-Power',
    Married: true,
    ID: '0987654321'
  },
  {
    _id: ObjectId('67484ff0a96ed4cf290d8192'),
    STID: '40311415052',
    FName: 'Reza',
    LName: 'Karimi',
    Father: 'Ali',
    Birth: '1382/09/10',
    SerialID: '741852963',
    BornCity: 'Mashhad',
    Address: 'mashad-----789 St.',
    PostalCode: '1111111111',
    CPhone: '09121111111',
    HPhone: '05112345678',
    Department: 'Basic Sciences',
    Major: 'Electrical Engineering-Electronics',
    Married: true,
    ID: '1122334455'
  }
]
```

حذف دانشجوی با 'Reza' fname =

```
db.student.deleteMany({FName: "Reza"});
```

```
LoRESTanUniv> db.student.deleteMany({FName: "Reza"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.student.find().pretty();
[
  {
    _id: ObjectId('67484ff0a96ed4cf290d8191'),
    STID: '40311415071',
    FName: 'Sara',
    LName: 'Rezaei',
    Father: 'Mohammad',
    Birth: '1382/07/20',
    SerialID: '987654321',
    BornCity: 'Shiraz',
    Address: 'shiraz-----456 St.',
    PostalCode: '9876543210',
    CPhone: '09123456789',
    HPhone: '03123456789',
    Department: 'Economics',
    Major: 'Electrical Engineering-Power',
    Married: true,
    ID: '0987654321'
  }
]
```

حذف دانشجوی با 'Shiraz' borncity =

```
db.student.deleteMany({BornCity: "Shiraz"});
```

```
LoRESTanUniv> db.student.deleteMany({BornCity: "Shiraz"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.student.find().pretty();
```

در جدول Lecturer :

اپدیت ها :

تغییر رشته و آدرس استاد با '123456' LID=

```
db.lecturer.updateOne({LID: "123456"}, {$set: {Major: "Civil Engineering", Address: "Ilam-----Updated Address"}});
```

```
LoRESTanUniv> db.lecturer.updateOne({LID: "123456"}, {$set: {Major: "Civil Engineering", Address: "Ilam-----Updated Address"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.lecturer.find({ LID: "123456" }).pretty();
[
  {
    _id: ObjectId('67485269a96ed4cf290d8193'),
    LID: '123456',
    FName: 'Reza',
    LName: 'Hosseini',
    ID: '4785136548',
    Department: 'Engineering',
    Major: 'Civil Engineering',
    Birth: '1372/03/10',
    BornCity: 'Ilam',
    Address: 'Ilam-----Updated Address',
    PostalCode: '7257567890',
    CPhone: '09121234567',
    HPhone: '02112345678'
  }
]
```

تغییر نام و شماره تماس استاد با 'LID='654321'

```
db.lecturer.updateOne({LID: "654321"}, {$set: { FName: "Maryam", CPhone: "09131111111"}});
```

```
 LorestanUniv> db.lecturer.updateOne({LID: "654321"}, {$set: { FName: "Maryam", CPhone: "09131111111"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
 LorestanUniv> db.lecturer.find({ LID: "654321" }).pretty();
[
  {
    _id: ObjectId('67485269a96ed4cf290d8194'),
    LID: '654321',
    FName: 'Maryam',
    LName: 'Rastegar',
    ID: '1548741215',
    Department: 'Economics',
    Major: 'Polymer Engineering',
    Birth: '1368/06/25',
    BornCity: 'Tehran',
    Address: 'tehran-----456 St.',
    PostalCode: '9876773120',
    CPhone: '09131111111',
    HPhone: '02187654321'
  }
]
```

تغییر محل تولد و کد پستی استاد با 'LID='789012'

```
db.lecturer.updateOne({LID: "789012"}, {$set: {BornCity: "Tabriz", PostalCode: "2222222222"}});
```

```
 LorestanUniv> db.lecturer.updateOne({LID: "789012"}, {$set: {BornCity: "Tabriz", PostalCode: "2222222222"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
 LorestanUniv> db.lecturer.find({ LID: "789012" }).pretty();
[
  {
    _id: ObjectId('67485269a96ed4cf290d8195'),
    LID: '789012',
    FName: 'Ali',
    LName: 'Shahbazi',
    ID: '6523526896',
    Department: 'Basic Sciences',
    Major: 'Electrical Engineering-Electronics',
    Birth: '1353/09/15',
    BornCity: 'Tabriz',
    Address: 'isfahan-----789 St.',
    PostalCode: '2222222222',
    CPhone: '09187654321',
    HPhone: '03112345678'
  }
]
```

حذف ها :

حذف استادی با 'LID='654321'

```
db.lecturer.deleteOne({LID: "654321"});
```

```
LoRESTanUniv> db.lecturer.deleteOne({LID: "654321"});  
{ acknowledged: true, deletedCount: 1 }  
LoRESTanUniv> db.lecturer.find({ LID: "654321" }).pretty();
```

حذف استادی با 'id = '6523526896'

```
db.lecturer.deleteOne({ID: "6523526896"});
```

```
LoRESTanUniv> db.lecturer.deleteOne({ID: "6523526896"});  
{ acknowledged: true, deletedCount: 1 }  
LoRESTanUniv> db.lecturer.find().pretty();  
[  
  {  
    _id: ObjectId('67485269a96ed4cf290d8193'),  
    LID: '123456',  
    FName: 'Reza',  
    LName: 'Hosseini',  
    ID: '4785136548',  
    Department: 'Engineering',  
    Major: 'Civil Engineering',  
    Birth: '1372/03/10',  
    BornCity: 'Ilam',  
    Address: 'Ilam-----Updated Address',  
    PostalCode: '7257567890',  
    CPhone: '09121234567',  
    HPhone: '02112345678'  
  }  
]
```

حذف استادی با 'lname = 'Hosseini'

```
db.lecturer.deleteMany({LName: "Hosseini"});
```

```
LoRESTanUniv> db.lecturer.deleteMany({LName: "Hosseini"});  
{ acknowledged: true, deletedCount: 1 }  
LoRESTanUniv> db.lecturer.find().pretty();
```

در جدول course :

آپدیت ها:

تغییر نام درس و تعداد واحد درس با CID='12345'

```
db.course.updateOne({CID: "12345"}, {$set: {CName: "Advanced Database", Credit: 4}});
```

```
LoRESTanUniv> db.course.updateOne({CID: "12345"}, {$set: {CName: "Advanced Database", Credit: 4}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.course.find({ CID: "12345" }).pretty();
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8196'),
    CID: '12345',
    CName: 'Advanced Database',
    Department: 'Engineering',
    Credit: 4
  }
]
```

تغییر دپارتمان درس با CID='23456'

```
db.course.updateOne({CID: "23456"}, {$set: {Department: "Basic Sciences"}});
```

```
LoRESTanUniv> db.course.updateOne({CID: "23456"}, {$set: {Department: "Basic Sciences"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.course.find({ CID: "23456" }).pretty();
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8197'),
    CID: '23456',
    CName: 'Microeconomic',
    Department: 'Basic Sciences',
    Credit: 4
  }
]
```

تغییر تعداد واحد درس با CID='34567'

```
db.course.updateOne({CID: "34567"}, {$set: {Credit: 3}});
```

```
LoRESTanUniv> db.course.updateOne({CID: "34567"}, {$set: {Credit: 3}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.course.find({ CID: "34567" }).pretty();
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8198'),
    CID: '34567',
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 3
  }
]
```

حذف ها :

حذف درسی با 'cid = '12345'

```
db.course.deleteOne({CID: "12345"});
```

```
LoRESTanUniv> db.course.deleteOne({CID: "12345"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.course.find().pretty();
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8197'),
    CID: '23456',
    CName: 'Microeconomic',
    Department: 'Basic Sciences',
    Credit: 4
  },
  {
    _id: ObjectId('67485f6ea96ed4cf290d8198'),
    CID: '34567',
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 3
  }
]
```

حذف درسی با 'cname = 'Microeconomic'

```
db.course.deleteOne({CName: "Microeconomic"});
```

```
LoRESTanUniv> db.course.deleteOne({CName: "Microeconomic"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.course.find().pretty();
[
  {
    _id: ObjectId('67485f6ea96ed4cf290d8198'),
    CID: '34567',
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 3
  }
]
```

حذف درسی با credit = 3

```
db.course.deleteMany({Credit: 3});
```

```
LoRESTanUniv> db.course.deleteMany({Credit: 3});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.course.find().pretty();
```


در جدول CourseRegister :

آپدیت ها :

تغییر تعداد واحد و نام درس برای ثبت نام با 'STID='40311415016' و 'CID='12345'

```
db.CourseRegister.updateOne({STID: "40311415016", CID: "12345"}, {$set: {Credit: 4, CName: "Database Advanced"}});
```

```
LoRESTanUniv> db.CourseRegister.updateOne({STID: "40311415016", CID: "12345"}, {$set: {Credit: 4, CName: "Database Advanced"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
LoRESTanUniv> db.CourseRegister.find({STID: "40311415016", CID: "12345"}).pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d8199'),
    Credit: 4,
    Department: 'Engineering',
    CName: 'Database Advanced',
    FName: 'Mobin',
    LName: 'Ramezani',
    STID: '40311415016',
    CID: '12345'
  }
]
```

تغییر دانشکده و نام دانشجو برای ثبت نام با 'STID='40311415071' و 'CID='23456'

```
db.CourseRegister.updateOne({STID: "40311415071", CID: "23456"}, {$set: {Department: "Basic Sciences", FName: "Sahar"}});
```

```
LoRESTanUniv> db.CourseRegister.updateOne({STID: "40311415071", CID: "23456"}, {$set: {Department: "Basic Sciences", FName: "Sahar"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.CourseRegister.find({STID: "40311415071", CID: "23456"}).pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d819a'),
    Credit: 4,
    Department: 'Basic Sciences',
    CName: 'Microeconomic',
    FName: 'Sahar',
    LName: 'Rezaei',
    STID: '40311415071',
    CID: '23456'
  }
]
```

تغییر نام خانوادگی دانشجو و نام درس با 'STID='40311415052' و 'CID='34567'

```
db.CourseRegister.updateOne({STID: "40311415052", CID: "34567"}, {$set: {LName: "Kiani", CName: "Physics - Quantum"}});
```

```
LoRESTanUniv> db.CourseRegister.updateOne({STID: "40311415052", CID: "34567"}, {$set: {LName: "Kiani", CName: "Physics - Quantum"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.CourseRegister.find({STID: "40311415052", CID: "34567"}).pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d819b'),
    Credit: 2,
    Department: 'Basic Sciences',
    CName: 'Physics - Quantum',
    FName: 'Reza',
    LName: 'Kiani',
    STID: '40311415052',
    CID: '34567'
  }
]
```

حذف ها :

حذف ثبت نام دانشجویی با 'stid = '40311415052'

```
db.CourseRegister.deleteMany({STID: "40311415052"});
```

```
 LorestanUniv> db.CourseRegister.deleteMany({STID: "40311415052"});
{ acknowledged: true, deletedCount: 1 }
 LorestanUniv> db.CourseRegister.find().pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d8199'),
    Credit: 4,
    Department: 'Engineering',
    CName: 'Database Advanced',
    FName: 'Mobin',
    LName: 'Ramezani',
    STID: '40311415016',
    CID: '12345'
  },
  {
    _id: ObjectId('67486365a96ed4cf290d819a'),
    Credit: 4,
    Department: 'Basic Sciences',
    CName: 'Microeconomic',
    FName: 'Sahar',
    LName: 'Rezaei',
    STID: '40311415071',
    CID: '23456'
  }
]
```

حذف ثبت نام دانشجویی با 'lname = 'Ramezani' و department = 'Engineering'

```
db.CourseRegister.deleteMany({Department: "Engineering", LName: "Ramezani"});
```

```
 LorestanUniv> db.CourseRegister.deleteMany({Department: "Engineering", LName: "Ramezani"});
{ acknowledged: true, deletedCount: 1 }
 LorestanUniv> db.CourseRegister.find().pretty();
[
  {
    _id: ObjectId('67486365a96ed4cf290d819a'),
    Credit: 4,
    Department: 'Basic Sciences',
    CName: 'Microeconomic',
    FName: 'Sahar',
    LName: 'Rezaei',
    STID: '40311415071',
    CID: '23456'
  }
]
```

حذف ثبت نام دانشجویی با 'fname = 'Sahar' و lname = 'Rezaei'

```
db.CourseRegister.deleteMany({FName: "Sahar", LName: "Rezaei"});
```

```
 LorestanUniv> db.CourseRegister.deleteMany({FName: "Sahar", LName: "Rezaei"});
{ acknowledged: true, deletedCount: 1 }
 LorestanUniv> db.CourseRegister.find().pretty();
```

در جدول PresentedCourses :

آپدیت ها :

تغییر نام درس و تعداد واحد برای درس ارائه شده با LID='123456' و CID='12345'

```
db.PresentedCourses.updateOne({LID: "123456", CID: "12345"}, {$set: {CName: "Advanced Databases", Credit: 4}});
```

```
LoRESTanUniv> db.PresentedCourses.updateOne({LID: "123456", CID: "12345"}, {$set: {CName: "Advanced Databases", Credit: 4}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.PresentedCourses.find({LID: "123456", CID: "12345"}).pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819c'),
    CName: 'Advanced Databases',
    Department: 'Engineering',
    Credit: 4,
    FName: 'Reza',
    LName: 'Hosseini',
    CID: '12345',
    LID: '123456'
  }
]
```

تغییر نام استاد و دپارتمان برای درس ارائه شده با LID='654321' و CID='23456'

```
db.PresentedCourses.updateOne({LID: "654321", CID: "23456"}, {$set: {FName: "Maryam", Department: "Basic Sciences"}});
```

```
LoRESTanUniv> db.PresentedCourses.updateOne({LID: "654321", CID: "23456"}, {$set: {FName: "Maryam", Department: "Basic Sciences"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.PresentedCourses.find({LID: "654321", CID: "23456"}).pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819d'),
    CName: 'Microeconomic',
    Department: 'Basic Sciences',
    Credit: 4,
    FName: 'Maryam',
    LName: 'Rastegar',
    CID: '23456',
    LID: '654321'
  }
]
```

تغییر نام درس و دپارتمان برای درس ارائه شده با LID='789012' و CID='34567'

```
UPDATE PresentedCourses SET CName = 'Quantum Physics', Department = 'Engineering' WHERE LID = '789012' AND CID = '34567';
```

```
LoRESTanUniv> db.PresentedCourses.updateOne({LID: "789012", CID: "34567"}, {$set: {CName: "Quantum Physics", Department: "Engineering"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
LoRESTanUniv> db.PresentedCourses.find({LID: "789012", CID: "34567"}).pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819e'),
    CName: 'Quantum Physics',
    Department: 'Engineering',
    Credit: 2,
    FName: 'Ali',
    LName: 'Shahbazi',
    CID: '34567',
    LID: '789012'
  }
]
```

حذف ها :

حذف درس ارائه شده با 'CID = '12345' و LID = '123456'

```
DELETE FROM PresentedCourses WHERE LID = '123456' AND CID = '12345';
```

```
LoRESTanUniv> db.PresentedCourses.deleteOne({LID: "123456", CID: "12345"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.PresentedCourses.find().pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819d'),
    CName: 'Microeconomic',
    Department: 'Basic Sciences',
    Credit: 4,
    FName: 'Maryam',
    LName: 'Rastegar',
    CID: '23456',
    LID: '654321'
  },
  {
    _id: ObjectId('6748664aa96ed4cf290d819e'),
    CName: 'Quantum Physics',
    Department: 'Engineering',
    Credit: 2,
    FName: 'Ali',
    LName: 'Shahbazi',
    CID: '34567',
    LID: '789012'
  }
]
```

حذف درس ارائه شده با 'cid = '23456' و lName = 'Rastegar'

```
db.PresentedCourses.deleteOne({LName: "Rastegar", CID: "23456"});
```

```
LoRESTanUniv> db.PresentedCourses.deleteOne({LName: "Rastegar", CID: "23456"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.PresentedCourses.find().pretty();
[
  {
    _id: ObjectId('6748664aa96ed4cf290d819e'),
    CName: 'Quantum Physics',
    Department: 'Engineering',
    Credit: 2,
    FName: 'Ali',
    LName: 'Shahbazi',
    CID: '34567',
    LID: '789012'
  }
]
```

حذف درس ارائه شده با 'cname = 'Quantum Physics'

```
DELETE FROM PresentedCourses WHERE cname = 'Quantum Physics';
```

```
LoRESTanUniv> db.PresentedCourses.deleteMany({CName: "Quantum Physics"});
{ acknowledged: true, deletedCount: 1 }
LoRESTanUniv> db.PresentedCourses.find().pretty();
```

۵) ۳ کاربر به این صورت ایجاد کنید : ۱- **Saman** با دسترسی کامل (ادمین). ۲- **Armin** با دسترسی فقط خواندنی.
۳- **Sara** با دسترسی نوشتن بر روی جدول **student** و خواندن از بقیه جدول ها. برای هر کاربر هر سه عملیات درج، حذف، بروز رسانی و خواندن را برای ۱ نمونه پرس و جو تست روی همه جداول تست کنید.

ابتدا وارد دیتابیس مدنظر می شود.

برای کاربر **Saman** :

ابتدا ایجاد کاربر و بررسی ایجاد آن :

```
db.createUser({user: "Saman", pwd: "1234", roles: [{role: "dbOwner", db: "LorestanUniv"}]});
```

```
db.getUsers();
```

```
LorestanUniv> db.createUser({user: "Saman", pwd: "1234", roles: [{role: "dbOwner", db: "LorestanUniv"}]});
{ ok: 1 }
LorestanUniv> db.getUsers();
{
  users: [
    {
      _id: 'LorestanUniv.Saman',
      userId: UUID('4ea889c1-06a5-4555-869a-ea5ddf6a7f4a'),
      user: 'Saman',
      db: 'LorestanUniv',
      roles: [ { role: 'dbOwner', db: 'LorestanUniv' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    }
  ],
  ok: 1
}
```

و بعد به کاربر **Saman** متصل می شویم :

```
db.auth("Saman", "1234");
```

```
db.runCommand({ connectionStatus: 1 });
```

```
LorestanUniv> db.auth("Saman", passwordPrompt());
Enter password
****{ ok: 1 }
LorestanUniv> db.runCommand({ connectionStatus: 1 })
{
  authInfo: {
    authenticatedUsers: [ { user: 'Saman', db: 'LorestanUniv' } ],
    authenticatedUserRoles: [ { role: 'dbOwner', db: 'LorestanUniv' } ]
  },
  ok: 1
}
```

سپس انجام دستورها برای saman :

درج :

```
db.student.insertOne({STID: "40311415066", FName: "Nima", LName: "Ahmadi", Father: "Hassan", Birth: "1381/04/15", SerialID: "123123123", BornCity: "Tehran", Address: "Tehran-----123 St.", PostalCode: "1231231234", CPhone: "09121111111", HPhone: "02122222222", Department: "Engineering", Major: "Computer Engineering", Married: false, ID: "3213213210" });
```

```
LoRESTanUniv> db.student.insertOne({STID: "40311415066", FName: "Nima", LName: "Ahmadi", Father: "Hassan", Birth: "1381/04/15", SerialID: "123123123", BornCity: "Tehran", Address: "Tehran-----123 St.", PostalCode: "1231231234", CPhone: "09121111111", HPhone: "02122222222", Department: "Engineering", Major: "Computer Engineering", Married: false, ID: "3213213210" });
{
  acknowledged: true,
  insertedId: ObjectId('674982fd9c6e5lad220d8190')
}
```

آپدیت :

```
db.lecturer.updateOne({ LID: "123456" }, {$set: {FName: "Ali"}});
```

```
LoRESTanUniv> db.lecturer.updateOne({ LID: "123456" }, {$set: {FName: "Ali"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

حذف :

```
db.course.deleteOne({CID: "12345"});
```

```
LoRESTanUniv> db.course.deleteOne({CID: "12345"});
{ acknowledged: true, deletedCount: 1 }
```

خواندن :

```
db.PresentedCourses.find().pretty();
```

```
LoRESTanUniv> db.PresentedCourses.find().pretty();
[
  {
    _id: ObjectId('6748ae0da96ed4cf290d81ab'),
    CName: 'Database',
    Department: 'Engineering',
    Credit: 3,
    FName: 'Reza',
    LName: 'Hosseini',
    CID: '12345',
    LID: '123456'
  },
  {
    _id: ObjectId('6748ae0da96ed4cf290d81ac'),
    CName: 'Microeconomic',
    Department: 'Economics',
    Credit: 4,
    FName: 'Nima',
    LName: 'Rastegar',
    CID: '123456',
    LID: '654321'
  },
  {
    _id: ObjectId('6748ae0da96ed4cf290d81ad'),
    CName: 'Advanced Physic',
    Department: 'Basic Sciences',
    Credit: 2,
    FName: 'Ali',
    LName: 'Shahbazi',
    CID: '34567',
    LID: '789012'
  }
]
```

برای کاربر Armin :

ابتدا ایجاد کاربر و بررسی ایجاد آن :

```
db.createUser({user: "Armin", pwd: "1234", roles: [{ role: "read", db: "LorestanUniv" }]});
```

```
db.getUsers();
```

```
LorestanUniv> db.createUser({user: "Armin", pwd: "1234", roles: [{ role: "read", db: "LorestanUniv" }]});
{ ok: 1 }
LorestanUniv> db.getUsers();
{
  users: [
    {
      _id: 'LorestanUniv.Armin',
      userId: UUID('4eda8f53-fe64-40f3-a78a-24f69b32445a'),
      user: 'Armin',
      db: 'LorestanUniv',
      roles: [ { role: 'read', db: 'LorestanUniv' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    },
    {
      _id: 'LorestanUniv.Saman',
      userId: UUID('1628f352-c9a5-4330-b84d-98b9998c238c'),
      user: 'Saman',
      db: 'LorestanUniv',
      roles: [ { role: 'dbOwner', db: 'LorestanUniv' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    }
  ],
  ok: 1
}
```

و بعد به کاربر Armin متصل می‌شویم :

```
db.auth("Armin", "1234");
```

```
db.runCommand({ connectionStatus: 1 });
```

سپس انجام دستورها برای armin :

درج (ارور دسترسی) :

```
db.student.insertOne({ STID: "40311415078", FName: "Ali", LName: "Karimi", Father: "Mohammad", Birth: "1380/02/15", SerialID: "456456456", BornCity: "Mashhad", Address: "Mashhad-----123 St.", PostalCode: "4564564567", CPhone: "0913222222", HPhone: "05133333333", Department: "Basic Sciences", Major: "Physics", Married: true, ID: "6546546541" });
```

```
LorestanUniv> db.student.insertOne({ STID: "40311415078", FName: "Ali", LName: "Karimi", Father: "Mohammad", Birth: "1380/02/15", SerialID: "456456456", BornCity: "Mashhad", Address: "Mashhad-----123 St.", PostalCode: "4564564567", CPhone: "0913222222", HPhone: "05133333333", Department: "Basic Sciences", Major: "Physics", Married: true, ID: "6546546541" });
Uncaught:
MongoServerError: Document failed validation
```

اپدیت (ارور دسترسی) :

```
db.lecturer.updateOne({LID: "123456"}, {$set: {FName: "Ali"}});
```

```
LorestanUniv> db.lecturer.updateOne({LID: "123456"}, {$set: {FName: "Ali"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
```

حذف (ارور دسترسی) :

```
db.course.deleteOne({CID: "12345"});
```

```
 LorestanUniv> db.course.deleteOne({CID: "12345"});  
{ acknowledged: true, deletedCount: 0 }
```

خواندن :

```
db.PresentedCourses.find();
```

```
 LorestanUniv> db.PresentedCourses.find();  
[  
  {  
    _id: ObjectId('674c9f52255d45c50c0d81a2'),  
    CName: 'Database',  
    Department: 'Engineering',  
    Credit: 3,  
    FName: 'Reza',  
    LName: 'Hosseini',  
    CID: '12345',  
    LID: '123456'  
  },  
  {  
    _id: ObjectId('674c9f52255d45c50c0d81a3'),  
    CName: 'Microeconomic',  
    Department: 'Economics',  
    Credit: 4,  
    FName: 'Mina',  
    LName: 'Rastegar',  
    CID: '23456',  
    LID: '654321'  
  },  
  {  
    _id: ObjectId('674c9f52255d45c50c0d81a4'),  
    CName: 'Advanced Physic',  
    Department: 'Basic Sciences',  
    Credit: 2,  
    FName: 'Ali',  
    LName: 'Shahbazi',  
    CID: '34567',  
    LID: '789012'  
  }  
]
```


برای کاربر Sara :

ابتدا ایجاد کاربر و بررسی ایجاد آن :

```
db.createUser({user: "Sara", pwd: "1234", roles: [{role: "readWrite", db: "LorestanUniv", collection: "student"}, {role: "read", db: "LorestanUniv"}]});
```

```
db.getUsers();
```

```
LorestanUniv> db.createUser({user: "Sara", pwd: "1234", roles: [{role: "readWrite", db: "LorestanUniv", collection: "student"}, {role: "read", db: "LorestanUniv"}]});
{ ok: 1 }
LorestanUniv> db.getUsers();
{
  users: [
    {
      _id: 'LorestanUniv.Armin',
      userId: UUID('aafdc2fd-b669-4691-a671-d167daaa81d0'),
      user: 'Armin',
      db: 'LorestanUniv',
      roles: [ { role: 'read', db: 'LorestanUniv' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    },
    {
      _id: 'LorestanUniv.Saman',
      userId: UUID('1628f352-c9a5-4330-b84d-98b9998c238c'),
      user: 'Saman',
      db: 'LorestanUniv',
      roles: [ { role: 'dbOwner', db: 'LorestanUniv' } ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    },
    {
      _id: 'LorestanUniv.Sara',
      userId: UUID('6fdbfb99-bff0-49ad-93d0-43c66c39b172'),
      user: 'Sara',
      db: 'LorestanUniv',
      roles: [
        { role: 'readWrite', db: 'LorestanUniv' },
        { role: 'read', db: 'LorestanUniv' }
      ],
      mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
    }
  ],
  ok: 1
}
```

و بعد به کاربر Sara متصل می‌شویم :

```
db.auth("Sara", "1234");
```

```
db.runCommand({ connectionStatus: 1 });
```

```
LorestanUniv> db.auth("Sara", passwordPrompt());
Enter password
****{ ok: 1 }
LorestanUniv> db.runCommand({ connectionStatus: 1 });
{
  authInfo: {
    authenticatedUsers: [ { user: 'Sara', db: 'LorestanUniv' } ],
    authenticatedUserRoles: [
      { role: 'read', db: 'LorestanUniv' },
      { role: 'readWrite', db: 'LorestanUniv' }
    ]
  },
  ok: 1
}
```

سپس انجام دستورها برای sara :

درج (ارور دسترسی برای غیر از student) :

```
db.student.insertOne({STID: "39911415056", FName: "Reza", LName: "Darvishi", Father: "MohammadAli", Birth: "1371/02/10", SerialID: "456456456", BornCity: "Mashhad", Address: "Mashhad-----123 St.", PostalCode: "4564564567", CPhone: "09132222222", HPhone: "05133333333", Department: "Engineering", Major: "Computer Engineering", Married: true, ID: "6546546541"});
```

```
LoRESTanUniv> db.student.insertOne({STID: "39911415056", FName: "Reza", LName: "Darvishi", Father: "MohammadAli", Birth: "1371/02/10", SerialID: "456456456", BornCity: "Mashhad", Address: "Mashhad-----123 St.", PostalCode: "4564564567", CPhone: "09132222222", HPhone: "05133333333", Department: "Engineering", Major: "Computer Engineering", Married: true, ID: "6546546541"});
{
  acknowledged: true,
  insertedId: ObjectId('674ca534219be753810d8190')
}
```

اپدیت (ارور دسترسی) :

```
db.student.updateOne({STID: "39911415056"}, {$set: {FName: "Ali"}});
```

```
LoRESTanUniv> db.student.updateOne({STID: "39911415056"}, {$set: {FName: "Ali"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}
```

حذف (ارور دسترسی) :

```
db.student.deleteOne({STID: "39911415056"});
```

```
LoRESTanUniv> db.student.deleteOne({STID: "39911415056"});
{ acknowledged: true, deletedCount: 0 }
```

۶) تراکنش زیر را پیاده سازی کنید : ۱- حذف رکورد یک درس از جدول **course** ۲- حذف همه رکوردهای مرتبط با رکورد مرحله ۱ از جدول انتخاب واحد ۳- حذف همه رکوردهای مرتبط با رکورد مرحله ۱ از جدول دروس ارائه شده. برای این تراکنش از دستورات **commit** ، استفاده کنید. اجرای همه مراحل تراکنش را با **commit** نشان دهید. لغو تراکنش را با دستورات **abort** و **rollback** انجام دهید و نشان دهید لغویکی از مراحل فوق چگونه منجر به لغو خودکار بقیه موارد می شود. همچنین از **savepoint** های مختلف استفاده کنید تا بعد از لغو به یک نقطه خاص برگردد و همه عملیات را لغو نکند.

```
const session = db.getMongo().startSession();
session.startTransaction();

try {
    print("Transaction started.....");
    const course_id = "34567";
    const delete_Course = db.course.deleteOne({CID: course_id}, {session});
    if (delete_Course.deletedCount === 0) {
        throw new Error("No record found in course collection.");
    }
    print("Deleted one record from course collection.");
    const delete_CourseRegister = db.CourseRegister.deleteMany({ CID: course_id }, { session });
    print("Deleted related records from CourseRegister collection.");
    const delete_PresentedCourses = db.PresentedCourses.deleteMany({ CID: course_id }, { session });
    print("Deleted related records from PresentedCourses collection.");
    session.commitTransaction();
    print("Transaction committed.");
} catch (error) {
    print("ERROR: " + error.message);
    session.abortTransaction();
    print("Transaction ABORTED.");
} finally {
    session.endSession();
    print("Session End.");
}
```

```

LorestanUniv> try {
...   print("Transaction started.....");
...   const course_id = "34567";
...   const delete_Course = db.course.deleteOne({CID: course_id}, {session});
...   if (delete_Course.deletedCount === 0) {
...     throw new Error("No record found in course collection.");
...   }
...   print("Deleted one record from course collection.");
...   const delete_CourseRegister = db.CourseRegister.deleteMany({ CID: course_id }, { session });
...   print("Deleted related records from CourseRegister collection.");
...   const delete_PresentedCourses = db.PresentedCourses.deleteMany({ CID: course_id }, { session });
...   print("Deleted related records from PresentedCourses collection.");
...   session.commitTransaction();
...   print("Transaction committed.");
... } catch (error) {
...   print("ERROR: " + error.message);
...   session.abortTransaction();
...   print("Transaction ABORTED.");
... } finally {
...   session.endSession();
...   print("Session End.");
... }
Transaction started.....
ERROR: r?.inTransaction is not a function
Transaction ABORTED.
Session End.

```