

1. Create a database "Student" with the following attributes Rollno, Name , Age, ContactNo, Email-Id, grade, hobby:
use Students

2. Insert 5 appropriate values according to the below queries.

```
db.students.insertMany([
  { "Rollno": 10, "Name": "John", "Age": 20, "ContactNo": "1234567890", "Email-Id":
"john@example.com", "grade": "A", "hobby": "Reading" },
  { "Rollno": 11, "Name": "Alice", "Age": 21, "ContactNo": "9876543210", "Email-Id":
"alice@example.com", "grade": "B", "hobby": "Painting" },
  { "Rollno": 12, "Name": "Bob", "Age": 22, "ContactNo": "2345678901", "Email-Id":
"bob@example.com", "grade": "C", "hobby": "Cooking" },
  { "Rollno": 13, "Name": "Eve", "Age": 23, "ContactNo": "3456789012", "Email-Id":
"eve@example.com", "grade": "A" },
  { "Rollno": 14, "Name": "Charlie", "Age": 24, "ContactNo": "4567890123", "Email-Id":
"charlie@example.com", "hobby": "Gardening" }
])
```

```
Atlas atlas-wanmtx-shard-0 [primary] Student> use Students
switched to db Students
Atlas atlas-wanmtx-shard-0 [primary] Students> show collections

Atlas atlas-wanmtx-shard-0 [primary] Students> db.students.insertMany([
...   { "Rollno": 10, "Name": "John", "Age": 20, "ContactNo": "1234567890", "Email-Id":
...   "john@example.com", "grade": "A", "hobby": "Reading" },
...   { "Rollno": 11, "Name": "Alice", "Age": 21, "ContactNo": "9876543210", "Email-Id":
...   "alice@example.com", "grade":
...   "B", "hobby": "Painting" },
...   { "Rollno": 12, "Name": "Bob", "Age": 22, "ContactNo": "2345678901", "Email-Id":
...   "bob@example.com", "grade": "C", "hobby": "Cooking" },
...   { "Rollno": 13, "Name": "Eve", "Age": 23, "ContactNo": "3456789012", "Email-Id":
...   "eve@example.com", "grade": "A"
...   },
...   { "Rollno": 14, "Name": "Charlie", "Age": 24, "ContactNo": "4567890123", "Email-Id":
...   "charlie@example.com", "hobby": "Gardening" }
... ])
{
  acknowledged: true,
  insertedIds: {
    0: ObjectId("661ce9dc76a00ff8cc51dae1"),
    1: ObjectId("661ce9dc76a00ff8cc51dae2"),
    2: ObjectId("661ce9dc76a00ff8cc51dae3"),
    3: ObjectId("661ce9dc76a00ff8cc51dae4"),
    4: ObjectId("661ce9dc76a00ff8cc51dae5")
  }
}
```

3. Write query to update Email-Id of a student with rollno 10.

```
db.students.updateOne(
  { "Rollno": 10 },
  { $set: { "Email-Id": "john.doe@example.com" } }
)
```

```
Atlas atlas-wanmtx-shard-0 [primary] Students> db.students.updateOne(
...   { "Rollno": 10 },
...   { $set: { "Email-Id": "john.doe@example.com" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

4. Replace the student name from “Alice” to “Alicee” of rollno 11

```
db.students.updateOne(
  { "Rollno": 11 },
  { $set: { "Name": "Alicee" } }
)
```

```
Atlas atlas-wanmtx-shard-0 [primary] Students> db.students.updateOne(
...   { "Rollno": 11 },
...   { $set: { "Name": "Alicee" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

5. Display Student Name and grade(Add if grade is not present)where the _id column is 1.

```
db.students.find({}, { "Name": 1, "grade": { $ifNull: ["$grade", "Not available"] }, "_id": 0 })
```

```
Atlas atlas-wanmtx-shard-0 [primary] Students> db.students.find({}, { "Name": 1, "grade":
{ $ifNull: ["$grade", "Not available"] }, "_id": 0 })
[
  { Name: 'John', grade: 'A' },
  { Name: 'Alicee', grade: 'B' },
  { Name: 'Bob', grade: 'C' },
  { Name: 'Eve', grade: 'A' },
  { Name: 'Charlie', grade: 'Not available' }
]
```

6. Update to add hobbies

```
db.students.updateMany(
  { "Name":
    "Eve" },
  { $set: { "hobby": "Dancing" }
})
```

)

```
Atlas atlas-wanmtx-shard-0 [primary] Students> db.students.updateMany(
...   { "Name": "Eve" },
...   { $set: { "hobby": "Dancing" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```