

SMARTBRIDGE EXTERNSHIP
MODERN APPLICATION DEVELOPMENT
Java Spring Boot
ASSIGNMENT- 2

Name : P. Chiranjeevi Sai Ram

Registration number : 20MIS7017

Campus : VITAP

1. CREATE, UPDATE , DELETE COMMANDS IN MySQL.

Create

```
MySQL v ⓘ
1 CREATE TABLE student (
2     Id INTEGER PRIMARY KEY,
3     name TEXT NOT NULL,
4     dept TEXT NOT NULL
5 );
6
7 -- insert
8 INSERT INTO student VALUES (1, 'john', 'cricket');
9 INSERT INTO student VALUES (2, 'virat', 'cricket');
10 INSERT INTO student VALUES (3, 'sindhu', 'batminton');
11
12 -- fetch
13 SELECT * FROM student;
```

 Run Save

Output

Id	name	dept
1	john	cricket
2	virat	cricket
3	sindhu	batmintion

[Execution complete with exit code 0]

Update

 MySQL ▾

```
1 CREATE TABLE student (  
2   Id INTEGER PRIMARY KEY,  
3   name TEXT NOT NULL,  
4   dept TEXT NOT NULL  
5 );  
6  
7 -- insert  
8 INSERT INTO student VALUES (1, 'john', 'cricket');  
9 INSERT INTO student VALUES (2, 'virat', 'cricket');  
10 INSERT INTO student VALUES (3, 'sindhu', 'batmintion');  
11  
12 -- fetch  
13 SELECT * FROM student;  
14  
15 -- update  
16 update student set dept="all" where Id=2;  
17 SELECT * FROM student;
```

 Run Save

Output

Id	name	dept
1	john	cricket
2	virat	cricket
3	sindhu	batmintion

Id	name	dept
1	john	cricket
2	virat	all
3	sindhu	batmintion

[Execution complete with exit code 0]

Delete



MySQL ▾



```
1 CREATE TABLE student (  
2   Id INTEGER PRIMARY KEY,  
3   name TEXT NOT NULL,  
4   dept TEXT NOT NULL  
5 );  
6  
7 -- insert  
8 INSERT INTO student VALUES (1, 'john', 'cricket');  
9 INSERT INTO student VALUES (2, 'virat', 'cricket');  
10 INSERT INTO student VALUES (3, 'sindhu', 'batmintion');  
11  
12 -- fetch  
13 SELECT * FROM student;  
14  
15 -- update  
16 update student set dept="all" where Id=2;  
17 SELECT * FROM student;  
18  
19 -- delete  
20 delete from student where Id=2;  
21 SELECT * FROM student;
```

 Run Save

Output

Id	name	dept
1	john	cricket
2	virat	cricket
3	sindhu	batmintion



Id	name	dept
1	john	cricket
2	virat	all
3	sindhu	batmintion

Id	name	dept
1	john	cricket
3	sindhu	batmintion

[Execution complete with exit code 0]

2. CREATE TABLES AND PERFORM JOINS IN MYSQL.

```
MySQL  MySQL 
1 CREATE TABLE student (
2   Id INTEGER PRIMARY KEY,
3   name TEXT NOT NULL,
4   dept TEXT NOT NULL
5 );
6
7 -- insert
8 INSERT INTO student VALUES (1, 'john', 'cricket');
9 INSERT INTO student VALUES (2, 'virat', 'cricket');
10 INSERT INTO student VALUES (3, 'sindhu', 'batmintion');
11
12 -- fetch
13 SELECT * FROM student;
14
15 -- create
16 CREATE TABLE course (
17   Id INTEGER PRIMARY KEY,
18   cname TEXT NOT NULL
19 );
20
21 -- insert
22 INSERT INTO course VALUES (1, 'batsnam');
23 INSERT INTO course VALUES (2, 'bowler');
24
25 -- fetch
26 select * from course;
27
28 -- inner join
```

```
MySQL  MySQL 
11
12 -- fetch
13 SELECT * FROM student;
14
15 -- create
16 CREATE TABLE course (
17   Id INTEGER PRIMARY KEY,
18   cname TEXT NOT NULL
19 );
20
21 -- insert
22 INSERT INTO course VALUES (1, 'batsnam');
23 INSERT INTO course VALUES (2, 'bowler');
24
25 -- fetch
26 select * from course;
27
28 -- inner join
29 select student.name, course.cname from student inner join course on student.Id=course.Id;
30
31 -- left join
32 select student.name, course.cname from student left join course on student.Id=course.Id;
33
34 -- right join
35 select student.name, course.cname from student right join course on student.Id=course.Id;
36
37 -- cross join
38 select name from student crosss join course;
```

 Run Save

Output

```
Id      name      dept
1       john      cricket
2       virat      cricket
3       sindhu     batmintion
Id      cname
1       batsnam
2       bowler
name    cname
john    batsnam
virat    bowler
name    cname
john    batsnam
virat    bowler
sindhu  NULL
name    cname
john    batsnam
virat    bowler
```

 Run Save

Output

```
virat    bowler
name     cname
john     batsnam
virat    bowler
sindhu   NULL
name     cname
john     batsnam
virat    bowler
name
john
john
virat
virat
sindhu
sindhu
```

[Execution complete with exit code 0]

3. CREATE, UPDATE, DELETE COMMANDS IN MONGODB

CREATE

MongoDB

```
1 db.student.insertMany([
2   {Id: 1, name: 'john', dept: 'cricket' },
3   {Id: 2, name: 'virat', dept: 'cricket' },
4   {Id: 3, name: 'sindhu', dept: 'batminton' }
5 ]);
6
7 db.student.find();
```

Run Save

Output

```
mycompiler_mongodb> ... .. {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6474665f5bc5a30e26383486"),
    '1': ObjectId("6474665f5bc5a30e26383487"),
    '2': ObjectId("6474665f5bc5a30e26383488")
  }
}
mycompiler_mongodb>
mycompiler_mongodb> [
  {
    _id: ObjectId("6474665f5bc5a30e26383486"),
    Id: 1,
    name: 'john',
    dept: 'cricket'
  },
  {
    _id: ObjectId("6474665f5bc5a30e26383487"),
    Id: 2,
    name: 'virat',
    dept: 'cricket'
  },
  {
    _id: ObjectId("6474665f5bc5a30e26383488"),
    Id: 3,
    name: 'sindhu',
    dept: 'batminton'
  }
]
```


 Run Save

Output

```
{
  },
  {
    _id: ObjectId("6474665f5bc5a30e26383487"),
    Id: 2,
    name: 'virat',
    dept: 'cricket'
  },
  {
    _id: ObjectId("6474665f5bc5a30e26383488"),
    Id: 3,
    name: 'sindhu',
    dept: 'batmintion'
  }
]
mycompiler_mongodb>
```

[Execution complete with exit code 0]

Update

MongoDB ⓘ

```
1 db.student.insertMany([
2   {Id: 1, name: 'john', dept: 'cricket' },
3   {Id: 2, name: 'virat', dept: 'cricket' },
4   {Id: 3, name: 'sindhu', dept: 'batminton' }
5 ]);
6
7 db.student.find();
8
9 db.student.updateOne( { Id: 1 }, { $set: { dept: "batminton"}})
10
11 db.student.find();
```

▶ Run

📁 Save

Output

```
mycompiler_mongodb> ... .. {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("647466aa6a776f9c1853f9da"),
    '1': ObjectId("647466aa6a776f9c1853f9db"),
    '2': ObjectId("647466aa6a776f9c1853f9dc")
  }
}
mycompiler_mongodb>
mycompiler_mongodb> [
  {
    _id: ObjectId("647466aa6a776f9c1853f9da"),
    Id: 1,
    name: 'john',
    dept: 'cricket'
  },
  {
```


 Run Save

Output

```
{
  _id: ObjectId("647466aa6a776f9c1853f9db"),
  Id: 2,
  name: 'virat',
  dept: 'cricket'
},
{
  _id: ObjectId("647466aa6a776f9c1853f9dc"),
  Id: 3,
  name: 'sindhu',
  dept: 'batminton'
}
]
mycompiler_mongodb>
mycompiler_mongodb> {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
```

 Run Save

Output

```
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
mycompiler_mongodb>
mycompiler_mongodb> [
  {
    _id: ObjectId("647466aa6a776f9c1853f9da"),
    Id: 1,
    name: 'john',
    dept: 'batminton'
  },
  {
    _id: ObjectId("647466aa6a776f9c1853f9db"),
    Id: 2,
    name: 'virat',
    dept: 'cricket'
  }
]
```

 Run Save

Output

```
{,
{
  _id: ObjectId("647466aa6a776f9c1853f9db"),
  Id: 2,
  name: 'virat',
  dept: 'cricket'
},
{
  _id: ObjectId("647466aa6a776f9c1853f9dc"),
  Id: 3,
  name: 'sindhu',
  dept: 'batminton'
}
]
mycompiler_mongodb>
```

[Execution complete with exit code 0]

Delete

 MongoDB ▾

```
1 - db.student.insertMany([
2   {Id: 1, name: 'john', dept: 'cricket' },
3   {Id: 2, name: 'virat', dept: 'cricket' },
4   {Id: 3, name: 'sindhu', dept: 'batminton' }
5  ]);
6
7  db.student.find();
8
9  db.student.remove({'Id':2})
10 db.student.find();
```

 Run Save

Output

```
mycompiler_mongodb> ... .. {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("647466aa6a776f9c1853f9da"),
    '1': ObjectId("647466aa6a776f9c1853f9db"),
    '2': ObjectId("647466aa6a776f9c1853f9dc")
  }
}
mycompiler_mongodb>
mycompiler_mongodb> [
  {
    _id: ObjectId("647466aa6a776f9c1853f9da"),
    Id: 1,
    name: 'john',
    dept: 'cricket'
  },
  {
```

 Run Save

Output

```
{
  _id: ObjectId("647466f6cb822b2842c45bea"),
  Id: 2,
  name: 'virat',
  dept: 'cricket'
},
{
  _id: ObjectId("647466f6cb822b2842c45beb"),
  Id: 3,
  name: 'sindhu',
  dept: 'batminton'
}
]
mycompiler_mongodb>
mycompiler_mongodb> DeprecationWarning: Collection.remove() is deprecated. Use deleteOne() or deleteMany() instead.
{ acknowledged: true, deletedCount: 1 }
mycompiler_mongodb> [
```

 Run Save

Output

```
mycompiler_mongodb>
mycompiler_mongodb> DeprecationWarning: Collection.remove() is deprecated. Use deleteOne() or deleteMany() instead.
{ acknowledged: true, deletedCount: 1 }
mycompiler_mongodb> [
  {
    _id: ObjectId("647466f6cb822b2842c45be9"),
    Id: 1,
    name: 'john',
    dept: 'cricket'
  },
  {
    _id: ObjectId("647466f6cb822b2842c45beb"),
    Id: 3,
    name: 'sindhu',
    dept: 'batminton'
  }
]
```