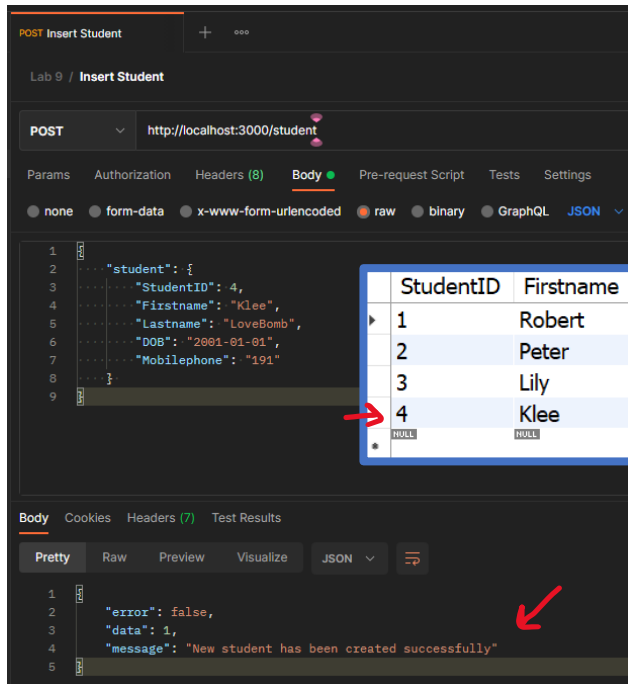


CRUD Operations

1. Insert Student



POST Insert Student

Lab 9 / Insert Student

POST http://localhost:3000/student

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "student": {
3     "StudentID": 4,
4     "Firstname": "Klee",
5     "Lastname": "LoveBomb",
6     "DOB": "2001-01-01",
7     "Mobilephone": "191"
8   }
9 }
```

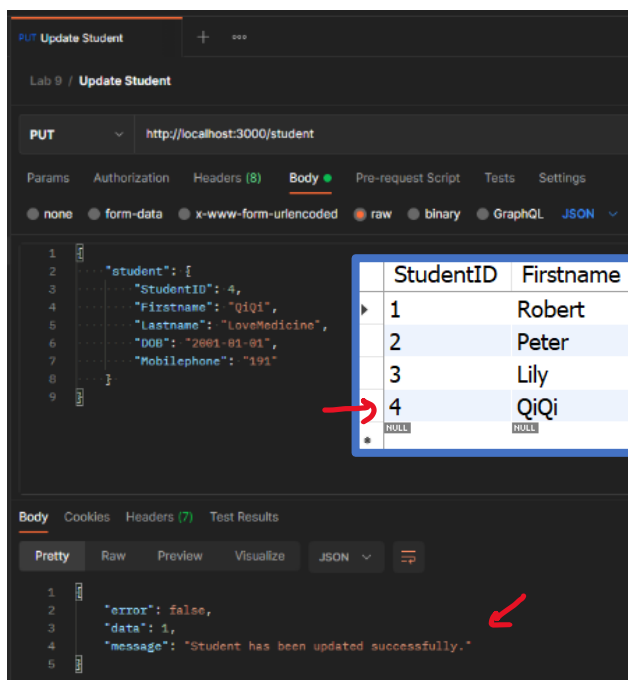
StudentID	Firstname	Lastname	DOB	Mobilephone
1	Robert	Dolls	1985-01-20 00:00:00	0919998877
2	Peter	Jones	1980-06-10 00:00:00	0834455667
3	Lily	James	1991-10-20 00:00:00	0889988776
4	Klee	LoveBomb	2001-01-01 00:00:00	191
* NULL	NULL	NULL	NULL	NULL

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "error": false,
3   "data": 1,
4   "message": "New student has been created successfully"
5 }
```

2. Update Student



PUT Update Student

Lab 9 / Update Student

PUT http://localhost:3000/student

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "student": {
3     "StudentID": 4,
4     "Firstname": "QiQi",
5     "Lastname": "LoveMedicine",
6     "DOB": "2001-01-01",
7     "Mobilephone": "191"
8   }
9 }
```

StudentID	Firstname	Lastname	DOB	Mobilephone
1	Robert	Dolls	1985-01-20 00:00:00	0919998877
2	Peter	Jones	1980-06-10 00:00:00	0834455667
3	Lily	James	1991-10-20 00:00:00	0889988776
4	QiQi	LoveMedicine	2001-01-01 00:00:00	191
* NULL	NULL	NULL	NULL	NULL

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "error": false,
3   "data": 1,
4   "message": "Student has been updated successfully."
5 }
```

3. Delete Student

The screenshot shows a REST client interface for a DELETE request to `http://localhost:3000/student`. The request body is a JSON object: `{ "StudentID": 4 }`. A table overlay displays the current student data:

	StudentID	Firstname	Lastname	DOB	Mobilephone
▶	1	Robert	Dolls	1985-01-20 00:00:00	0919998877
	2	Peter	Jones	1980-06-10 00:00:00	0834455667
	3	Lily	James	1991-10-20 00:00:00	0889988776
	NULL	NULL	NULL	NULL	NULL

The response body is a JSON object indicating success:

```

1 {
2   "error": false,
3   "data": 1,
4   "message": "Student has been deleted successfully."
5 }

```

4. Select a Student

The screenshot shows a REST client interface for a GET request to `http://localhost:3000/student/4`. The response body is a JSON object:

```

1 {
2   "error": false,
3   "data": {
4     "StudentID": 4,
5     "Firstname": "QiQi",
6     "Lastname": "LoveMedicine",
7     "DOB": "2000-12-31T17:00:00.000Z",
8     "Mobilephone": "191"
9   },
10  "message": "Student retrieved"
11 }

```

5. Select all Students

The screenshot shows a REST client interface for a GET request to `http://localhost:3000/students`. The response body is a JSON array of student objects:

```

1 {
2   "error": false,
3   "data": [
4     {
5       "StudentID": 1,
6       "Firstname": "Robert",
7       "Lastname": "Dolls",
8       "DOB": "1985-01-20T17:00:00.000Z",
9       "Mobilephone": "0919998877"
10    },
11    {
12       "StudentID": 2,
13       "Firstname": "Peter",
14       "Lastname": "Jones",
15       "DOB": "1980-06-10T17:00:00.000Z",
16       "Mobilephone": "0834455667"
17    },
18    {
19       "StudentID": 3,
20       "Firstname": "Lily",
21       "Lastname": "James",
22       "DOB": "1991-10-20T17:00:00.000Z",
23       "Mobilephone": "0889988776"
24    },
25    {
26       "StudentID": 4,
27       "Firstname": "QiQi",
28       "Lastname": "LoveMedicine",
29       "DOB": "2000-12-31T17:00:00.000Z",
30       "Mobilephone": "191"
31    }
32  ],

```