Testing :

Install Jest

Look for package.json, if there is no package.json file in your project, just create it using npm init.

Then in the package.json

```json
{
  "name": "testing",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "directories": {
    "test": "test"
  },
  "dependencies": {
    "axios": "^0.19.0",
    "body-parser": "^1.19.0",
    "cors": "^2.8.5",
    "express": "^4.17.0",
    "jest": "^24.8.0",
    "jsonwebtoken": "^8.5.1",
    "mongoose": "^5.5.9",
    "multer": "^1.4.1",
    "supertest": "^4.0.2"
  },
  "devDependencies": {},
  "scripts": {
    "test": "jest"
  },
  "author": "",
  "license": "ISC"
}
```

As you can see your package.json will look like this.

Do not copy, just put jest as a value in the scripts (you can see the red color section)

Step 2 :

Create a folder called tests in your root:

Then copy all the codes below in a file inside tests folder(name of the file example user.test.js)

Note the name of the file must be something.test.js.

```javascript
// use the path of your model
const Product = require('../models/product');
const mongoose = require('mongoose');
// use the new name of the database
const url = 'mongodb://localhost:27017/new_database_name';
beforeAll(async () => {
    await mongoose.connect(url, {
        useNewUrlParser: true,
        useCreateIndex: true
    });
});

afterAll(async () => {

    await mongoose.connection.close();
});

describe('Product  Schema test anything', () => {
// the code below is for insert testing
    it('Add product testing anything', () => {
        const product = {
            'pname': 'Nokia',
            'pprice': '21'
        };

        return Product.create(product)
            .then((pro_ret) => {
                expect(pro_ret.pname).toEqual('Nokia');
            });
    });
// the code below is for delete testing
    it('to test the delete product is working or not', async () => {
        const status = await Product.deleteMany();
        expect(status.ok).toBe(1);
});
```

```
it('to test the update', async () => {

    return Product.findOneAndUpdate({_id :Object('5d20c71c0da2982d681e4bf0')},
{$set : {pname:'ram'}})
    .then((pp)=>{
        expect(pp.pname).toEqual('ram')
    })

});

})
```

Step 3 :

Runing the test using following command :

Npm test