**[Practical-5]**

**Create an Application to perform CRUD Operations using Mongoose**

1. Create a Blogger's Database having schema with different Data Structures available.

2. Create a Model with Post collection to perform all CRUD Operations.

3. Insert multiple posts and read all.

4. Read, Update and Delete a record using Mongoose model.

5. Adding validation to the data structure and validate it.

**Instructions:**

- Perform all the above activities.

- Student have to submit code of "app.js" into the implementation section.

**Implementation:**

const mongoose=require("mongoose");

1. mongoose.connect('mongodb://127.0.0.1:27017/Bloggers',{useNewUrlParser:true, useUnifiedTopology:true, useFindAndModify:false}). then(()=>console.log("connected")).catch(()=>console.log("error"));

2.

const col=new mongoose.model("Post",schema); const p1=new col({ Author:"Jai", BlogCategory:"X", num\_Blogs:11, Publish\_Date:Date.now(), Published:true }) const p2=new col({ Author:"Raj", BlogCategory:"Y", num\_Blogs:25, Publish\_Date:Date.now(), Published:false });

3.

col.insertMany([p1,p2]); col.find({}).then((d)=>console.log(d)).catch((e)=>console.log(e));

4.

col.find({Author:"Jai"}).then((d)=>console.log(d)); col.remove({Author:"Raj"}).then((d)=>console.log(d)); col.update({Author:"Jai"},{$set:{Author:"Jai Sharma"}}).then((d)=>console.log(d));

5.

const schema=new mongoose.Schema({ Author:{type:String, required:[true,"Name must be required"]}, BlogCategory:String, num\_Blogs:Number, Publish\_Date:{type:Date,default:Date.now}, Published:Boolean });