**Ejs:**

var express = require("express");

var app = express();

var mysql = require("mysql");

var bodyParser = require("body-parser");

app.use(bodyParser.urlencoded({ extended: true }));

app.use(bodyParser.json());

app.set("view engine", "ejs");

var conn = mysql.createConnection({

  host: "localhost",

  user: "root",

  password: "",

  database: "muntazer\_shah",

  port: 3306

});

conn.connect(function (err) {

  if (err) throw err;

  console.log("Connection Sucessful");

});

app.get("/", function (req, res) {

  res.render("insert");

});

app.post("/insert", function (req, res) {

  var fname = req.body.fname;

  var lname = req.body.lname;

  var email = req.body.email;

  var password = req.body.password;

  var cpassword = req.body.cpassword;

  var Province = req.body.Province;

  var Address = req.body.Address;

  var phone = req.body.phone;

  var gender = req.body.gender;

  var sdate=req.body.sdate;

  var edate=req.body.edate;

  var field=req.body.field;

  var sql = `insert into student(FirstName,LastName, Email, Password, CPassword, Province,Address,Phone,Gender,StartDate,EndDate,field) values('${fname}', '${lname}', '${email}', '${password}', '${cpassword}', '${Province}', '${Address}', '${phone}','${gender}', '${sdate}', '${edate}', '${field}')`;

  conn.query(sql, function (err, results) {

    if (err) throw err;

    res.send("<h1>Data Inserted.</h1>");

  });

});

var server = app.listen(4001, function () {

  console.log("App running on port 4001");

});

**Ajax:**

var express = require("express");

var cors = require("cors");

var app = express();

app.use(cors());

var mysql = require("mysql");

var bodyParser = require("body-parser");

app.use(bodyParser.urlencoded({ extended: true }));

app.use(bodyParser.json());

app.set("view engine", "ejs");

var conn = mysql.createConnection({

  host: "localhost",

  user: "root",

  password: "",

  database: "abubakar",

});

conn.connect(function (err) {

  if (err) throw err;

  console.log("Connection Sucessful");

});

app.get("/", function (req, res) {

  res.render("insert");

});

app.post("/insert", function (req, res) {

  var name = req.body.name;

  var email = req.body.email;

  var password = req.body.password;

  var sql = `insert into student2(id,name, email, password) values(',','${name},', '${email}', '${password}')`;

  conn.query(sql, function (err, results) {

    if (err) throw err;

    res.send("<h1>Data Inserted.</h1>");

  });

});

app.get("/getdata", function (req, res) {

    var id = req.query.id;

    var sql = `SELECT \* FROM student2 WHERE id=${id}`;

    conn.query(sql, function (err2, results) {

        if (err2) {

            console.error(err2);

        } else {

            res.json(results);

        }

    });

});

app.delete("/delete", function (req, res) {

    var id = req.body.id;

    var sql = `DELETE FROM student2 WHERE id = ${id}`;

    conn.query(sql, function (err, result) {

      if (err) throw err;

      res.send(result);

    });

  });

  app.post('/update', function(req, res) {

    var id = req.body.id;

    var name = req.body.name;

    var email = req.body.email;

    var password = req.body.password;

    var sql = "UPDATE student2 SET name = ?, email = ?, password = ? WHERE id = ?";

    conn.query(sql, [name, email, password, id], function(err, result) {

      if (err) throw err;

      res.send(result);

    });

  });

var server = app.listen(2999, function () {

  console.log("App running on port 2999");

});

**Insert html:**

unction submit()

{

  var  name = $("#name").val();

  var  email = $("#email").val();

  var  password = $("#password").val();

    var data = {name : name, email : email, password: password};

    let post = JSON.stringify(data);

    console.log(post);

    $.ajax({

        url: 'http://localhost:4000/insert',

  headers: {

        'Content-Type':'application/json'

    },

        type: 'POST',

        contentType: 'application/json',

        data: post,

        success: function(){

    alert("Data Inserted");

        },

        error: function(){

            alert('error occured');

        }

    });

}

function getdata() {

    var id = $("#id2").val();

    $.ajax({

        url: 'http://localhost:4000/getdata?id=' + id,

        headers: {

            'Content-Type': 'application/json'

        },

        type: 'GET',

        contentType: 'application/json',

        success: function (result) {

            result.forEach(function (obj) {

                console.log(obj.name);

                $("#name").val(obj.name);

                $("#email").val(obj.email);

                $("#password").val(obj.password);

            });

        },

        error: function () {

            alert('error');

        }

    });

}

function deleteData() {

  var id = $("#id2").val();

  var data = { id: id };

  let post = JSON.stringify(data);

  $.ajax({

    url: "http://localhost:4000/delete",

    headers: {

      "Content-Type": "application/json",

    },

    type: "DELETE",

    contentType: "application/json",

    data: post,

    success: function () {

      alert("Data Deleted");

    },

    error: function () {

      alert("Error");

    },

  });

}

function updateData() {

  getdata();

  var id = $("#id2").val();

  var name = $("#name").val();

  var email = $("#email").val();

  var password = $("#password").val();

  var data = {id: id, name: name, email: email, password: password};

  var postData = JSON.stringify(data);

  $.ajax({

    url: 'http://localhost:4000/update',

    headers: {'Content-Type': 'application/json'},

    type: 'POST',

    contentType: 'application/json',

    data: postData,

    success: function(result) {

      console.log(result);

      alert("Data updated successfully");

    },

    error: function() {

      alert('Error updating data');

    }

  });

}

**Google Map: (Register.Html)**

var map;

function initAutocomplete() {

        var map = new google.maps.Map(document.getElementById('map'), {

            center : {

                lat : 30.3753,

                lng : 69.3451

            },

            zoom : 5

        });

         var options = {

          componentRestrictions: {country: "pk"}

         };

        var input = (document.getElementById('pac-input'));

        var autocomplete = new google.maps.places.Autocomplete(input, options);

        autocomplete.bindTo('bounds', map);

        var infowindow = new google.maps.InfoWindow();

        var marker = new google.maps.Marker({

            draggable:true,

            map : map

        });

        autocomplete.addListener('place\_changed', function() {

            var place = autocomplete.getPlace();

            var lattitude = place.geometry.location.lat();

            var longitude = place.geometry.location.lng();

google.maps.event.addListener(marker, 'dragend', function (event) {

    document.getElementById("lattitude").value = this.getPosition().lat();

    document.getElementById("longitude").value = this.getPosition().lng();

    //Marker Location Changed

    document.getElementById("lattitude").value = this.getPosition().lat();

    document.getElementById("longitude").value = this.getPosition().lng();

});

            //alert(lattitude + ' ' + longitude);

            document.getElementById("lattitude").value = lattitude;

            document.getElementById("longitude").value = longitude;

            //Selected Place Changed

            //alert("Hello");

            infowindow.close();

            marker.setVisible(false);

            var place = autocomplete.getPlace();

            if (!place.geometry) {

                window.alert("Autocomplete's returned place contains no geometry");

                return;

            }

            // If the place has a geometry, then present it on a map.

            if (place.geometry.viewport) {

                map.fitBounds(place.geometry.viewport);

            } else {

                map.setCenter(place.geometry.location);

                map.setZoom(17);

                // Why 17? Because it looks good.

            }

            marker.setPosition(place.geometry.location);

            marker.setVisible(true);

            var address = '';

            if (place.address\_components) {

                address = [(place.address\_components[0] && place.address\_components[0].short\_name || ''), (place.address\_components[1] && place.address\_components[1].short\_name || ''), (place.address\_components[2] && place.address\_components[2].short\_name || '')].join(' ');

            }

            infowindow.setContent('<div><strong>' + place.name + '</strong><br>' + address);

            infowindow.open(map, marker);

            // iterate through address\_component array

        });

**Node Mailer: (Rest User)**

const nodemailer = require("nodemailer");

app.post("/send-code", (req, res) => {

  const { name, email, code } = req.body;

  const transporter = nodemailer.createTransport({

    service: "gmail",

    auth: {

      user: "f201023@cfd.nu.edu.pk",

      pass: "03074659133",

    },

  });

  const mailOptions = {

    from: "f201023@cfd.nu.edu.pk",

    to: email,

    subject: "Authentication Code",

    text: `Hi ${name},\n\nYour authentication code is: ${code}\n\nBest regards,\nAuthentication Team`,

  };

  transporter.sendMail(mailOptions, function (error, info) {

    if (error) {

      console.log(error);

      res.status(500).send("Failed to send code.");

    } else {

      console.log("Email sent: " + info.response);

      res.send("Code sent to your email address.");

    }

  });

});

**Multer: (Server.js)**

const multer = require("multer");

const storage1 = multer.diskStorage({

  destination: function (req, file, cb) {

    cb(null, "./Register/RegisterImage/");

  },

  filename: function (req, file, cb) {

    cb(null, `${Date.now()}-${file.originalname}`);

  },

});

const upload = multer({ storage: storage1 });

app.post("/insert", upload.single("image3"), function (req, res) {

  res.setHeader("Access-Control-Allow-Origin", "\*");

  res.setHeader(

    "Access-Control-Allow-Methods",

    "GET, POST, OPTIONS, PUT, PATCH, DELETE"

  ); // If needed

  res.setHeader(

    "Access-Control-Allow-Headers",

    "X-Requested-With,content-type"

  ); // If needed

  res.setHeader("Access-Control-Allow-Credentials", true); // If needed

**Session: (Admin Login)\**

function sendData() {

    var username = name2;

    sessionStorage.setItem("name", username);

    window.location.href = "../Admin/AdminDashboard.html";

  }

**Admin Dashboard:**

const myText = sessionStorage.getItem("name");

  var myParagraph = document.getElementById("name");

  myParagraph.innerHTML = myText;

**Bcrypt: (Server.js)**

const bcrypt = require("bcrypt");

// Generate a salt for hashing

const salt = '$2b$10$abcde12345fghij67890kl';

 const hash = bcrypt.hashSync(password, salt);

console.log(hash);

  var sql = `insert into lostfound values( ' ' ,'${firstname}', '${lastname}', '${email}' , '${age}', '${phone}', '${gender}', '${address}', '${city}', '${zipcode}', '${hash}', '${lat}', '${long}')`;