var express = require('express')

var app = express();

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

var mysql = require('mysql');

const e = require('express');

const { json } = require('body-parser');

const { query } = require('express');

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({

extended: true

}));

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

return res.send({ error: true, message: 'Test Student Web API' })

});

var dbConn = mysql.createConnection({

host: 'localhost',

user: 'root',

password: '',

database: 'shabu'

});

dbConn.connect();

//User

app.get('/allregister', function (req, res) {

dbConn.query('SELECT \* FROM user', function (error, results, fields) {

if (error) throw error;

return res.send(results);

});

});

//registerUser

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

var register = req.body

if (!register) {

return res.status(400).send({ error: true, message: 'Please provide user' });

}

dbConn.query("INSERT INTO user SET ?", register, function (error, results, fields) {

if (error) throw error;

return res.send(results);

});

});

//loginUser

app.post("/login", (req, res) => {

let email = req.body.user\_email;

let password = req.body.user\_password;

dbConn.query('SELECT \* FROM user WHERE user\_email = ?', email, function (error, results, fields) {

if (error) throw error;

if (results && results.length) {

let passwordFromDb = results[0].user\_password

if (password == passwordFromDb) {

return res.send(results[0]);

} else {

return res.status(400).send({ error: true, message: "Password does not match" });

}

} else {

return res.status(400).send({ error: true, message: "User does not exist" })

}

});

});

app.put('/user/:id', function(req,res){

let user\_id = req.params.id;

let user = req.body

if(!user\_id || !user){

return res.status(400).send({error:true,message: 'Please provide student id and student data'});

}

dbConn.query('UPDATE user SET ? WHERE id\_user = ?', [user,user\_id], function(error,results,fields){

if(error) throw error;

return res.send({error: false, message: 'Student has been updated seccessfully'});

});

})

// app.put('/editProfile/:id\_user', function (req, res) {

// let userId = req.params.id\_user;

// let name = req.body.user\_name;

// let email = req.body.user\_email;

// let password = req.body.user\_password;

// let tel = req.body.tel;

// if (!userId) {

// return res.status(400).send({ error: true, message: "Please provide emp id" });

// } else {

// dbConn.query('UPDATE user SET ? WHERE user\_id = ?', [name, email,password,tel], function (error, results, fields) {

// if (error) throw error;

// return res.send(results)

// });

// }

// });

//Admin

app.get('/allAdmin', function (req, res) {

dbConn.query('SELECT \* FROM admin', function (error, results, fields) {

if (error) throw error;

return res.send(results);

});

});

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

var table = req.body

if(!table){

return res.status(400).send({error:true,message:'Please provide student'});

}

dbConn.query("INSERT INTO desk SET ?",table , function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

//Admin

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

let email = req.body.admin\_email;

let password = req.body.admin\_password;

dbConn.query('SELECT \* FROM admin WHERE admin\_email = ?', email, function (error, results, fields) {

if (error) throw error;

if (results && results.length) {

let passwordFromDb = results[0].admin\_password

if (password == passwordFromDb) {

return res.send(results[0]);

} else {

return res.status(400).send({ error: true, message: "Password does not match" });

}

} else {

return res.status(400).send({ error: true, message: "User does not exist" })

}

});

});

app.put('/admin/:id', function(req,res){

let admin\_id = req.params.id;

let admin = req.body

if(!admin\_id || !admin){

return res.status(400).send({error:true,message: 'Please provide student id and student data'});

}

dbConn.query('UPDATE admin SET ? WHERE id\_admin = ?', [admin,admin\_id], function(error,results,fields){

if(error) throw error;

return res.send({error: false, message: 'Student has been updated seccessfully'});

});

})

//table

app.get('/alltable', function(req, res){

dbConn.query('SELECT \* FROM desk', function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

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

var table = req.body

if(!table){

return res.status(400).send({error:true,message:'Please provide student'});

}

dbConn.query("INSERT INTO desk SET ?",table , function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

app.put('/table/:id', function(req,res){

let table\_id = req.params.id;

let table = req.body

if(!table\_id || !table){

return res.status(400).send({error:true,message: 'Please provide student id and student data'});

}

dbConn.query('UPDATE desk SET ? WHERE id\_desk = ?', [table,table\_id], function(error,results,fields){

if(error) throw error;

return res.send({error: false, message: 'Student has been updated seccessfully'});

});

})

//tabletype

app.get('/alltabletype', function(req, res){

dbConn.query('SELECT \* FROM desk\_type', function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

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

var table = req.body

if(!table){

return res.status(400).send({error:true,message:'Please provide student'});

}

dbConn.query("INSERT INTO desk\_type SET ?",table , function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

app.put('/tabletype/:id', function(req,res){

let tabletype\_id = req.params.id;

let tabletype = req.body

if(!tabletype\_id || !tabletype){

return res.status(400).send({error:true,message: 'Please provide student id and student data'});

}

dbConn.query('UPDATE desk\_type SET ? WHERE id\_desk\_type = ?', [tabletype,tabletype\_id], function(error,results,fields){

if(error) throw error;

return res.send({error: false, message: 'Student has been updated seccessfully'});

});

})

app.delete('/tabletype/:id', function(req,res){

let tabletype\_id = req.params.id;

if(!tabletype\_id){

return res.status(400).send({ error: true, message: 'Please provide student id'});

}

dbConn.query('DELETE FROM desk\_type WHERE id\_desk\_type = ?',tabletype\_id,function (error,results, fields){

if (error) throw error;

return res.send({error: false,message: 'Student has been deleted successfully'});

});

})

//booking

app.get('/allbooking', function (req, res) {

dbConn.query('SELECT \* FROM booking', function (error, results, fields) {

if (error) throw error;

return res.send(results);

});

});

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

var booking = req.body

if(!booking){

return res.status(400).send({error:true,message:'Please provide student'});

}

dbConn.query("INSERT INTO booking SET ?",booking , function(error,results,fields){

if(error) throw error;

return res.send(results);

});

});

app.delete('/booking/:id', function(req,res){

let booking\_id = req.params.id;

if(!booking\_id){

return res.status(400).send({ error: true, message: 'Please provide student id'});

}

dbConn.query('DELETE FROM booking WHERE id\_booking = ?',booking\_id,function (error,results, fields){

if (error) throw error;

return res.send({error: false,message: 'Student has been deleted successfully'});

});

})

app.listen(3000, function () {

console.log('Node app is running on port 3000');

});

module.exports = app;