**LAB6**

**Update online shopping application, change CRUD operations on Product Model to use MongoDB.**

** Admin: save/edit/delete product, view all products**

**Database.js**

const mongoClient = require('mongodb').MongoClient;

let \_db;

const mongoconnect = function(callback){

mongoClient.connect('mongodb://localhost:27017', {useUnifiedTopology: true})

.then(client => {

\_db = client.db('onlineshopping');

//callback(client);

callback();

})

.catch(err => {

console.log(err);

throw new Error('DB connection failed...');

});

}

const getDb = () => {

if(\_db) {

return \_db;

} else {

throw new Error('Db connect failed');

}

}

//module.exports = mongoconnect;

exports.mongoconnect = mongoconnect;

exports.getDb = getDb;

**Save product**

**Product.js**

const getDb = require('../util/database').getDb;

const ObjectId = require('mongodb').ObjectID;

save() {

const db = getDb();

db.collection('products').insertOne(this)

.then(result => {

console.log(result.result);

client.close();

})

.catch(err => console.log(err));

}

**Edit Product**

**Index.ejs**

<a class="btn btn-primary" href="/admin/edit-product/<%=p.\_id%>" role="button">Edit</a>

**Product.js**

static findById(prodId) {

//return products.filter(p => p.id == prodId);

const db = getDb();

return db.collection('products')

.findOne({\_id: new ObjectId(prodId)});

}

**Admin.js (controller)**

exports.editProductPage = (req, res, next) => {

Product.findById(req.params.prodId)

.then(product => {

res.render('edit-product', { product: product, path: '/', pageTitle: 'Edit Product', name: 'Tina' });

}).catch(err => console.log(err));

}

exports.editProductPost = (req, res, next) => {

const updatedProduct = new Product(req.body.id, req.body.title, req.body.price, req.body.imageURL, req.body.description);

updatedProduct.update()

.then(result => {

res.redirect('/products/' + updatedProduct.\_id);

})

.catch(err => console.log(err));

}

**Delete product**

**Index.ejs**

<form action="/admin/delete-product" method="POST">

<input type="hidden" name="id" value="<%=p.\_id%>">

<button type="submit" class="btn btn-primary">Delete</button>

</form>

**Product.js**

static deleteById(prodId) {

const db = getDb();

return db.collection('products')

.remove({ \_id: new ObjectId(prodId) });

}

**Admin.js (controller)**

exports.deleteProduct = (req, res, next) => {

Product.deleteById(req.body.id)

.then(result => {

res.redirect('/');

})

.catch(err => console.log(err));

}

** Shop: view detail of product, view all products**

**Product.js**

static findById(prodId) {

//return products.filter(p => p.id == prodId);

const db = getDb();

return db.collection('products')

.findOne({\_id: new ObjectId(prodId)});

}

**Shop.js (controller)**

exports.getProductDetail = (req, res, next) => {

//const products = Product.findById(req.params.prodId);

Product.findById(req.params.prodId)

.then(product => {

res.render('product-detail', { prod: product, pageTitle: 'Product Detail', path: '/', name: 'Edward' });

})

.catch(err => console.log(err));

}