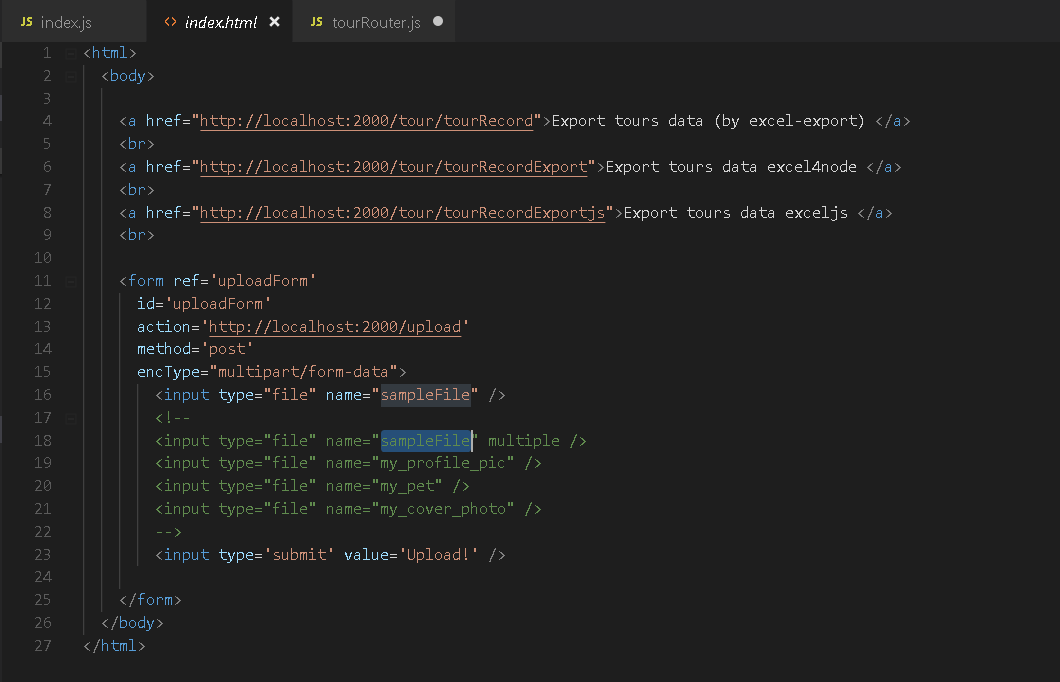
# express-fileupload

<https://www.npmjs.com/package/express-fileupload>

<https://github.com/richardgirges/express-fileupload/tree/master/example#basic-file-upload>

UI –



Backend/Node part

const fileUpload = require('express-fileupload');

// default options

app.use(fileUpload());

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

if (Object.keys(req.files).length == 0) {

return res.status(400).send('No files were uploaded.');

}

// The name of the input field (i.e. "sampleFile") is used to retrieve the uploaded file

let sampleFile = req.files.sampleFile;

console.log("sampleFile::::"+sampleFile);

var mimetype = sampleFile.mimetype;

console.log("file type:::"+mimetype);

var fullName = sampleFile.name;

console.log("file fullName:::"+fullName);

if(fullName){

var fields = fullName.split('.')

var name = fields[0];

var imageType = fields[1];

//console.log('name::::'+name+':::type:::'+imageType);

}

// Use the mv() method to place the file somewhere on your server

sampleFile.mv(path.join(\_\_dirname, 'public', 'uploadedFile', fullName), function(err) {

if (err){

console.log("err::::"+err);

return res.status(500).send(err);

}

});

------\*\*\*\*\*\*\*\*\*\*\*\*\*\*--------------

**Compress-images**

<https://www.npmjs.com/package/compress-images>

var compress\_images = require('compress-images');

//compress images

const INPUT\_path\_to\_your\_images = 'public/uploadedFile/'+fullName;

//const INPUT\_path\_to\_your\_images = 'public/uploadedFile/'+name+'.'+imageType;//'.{jpg,JPG,jpeg,JPEG,png,svg,gif}';

console.log("INPUT\_path\_to\_your\_images::::"+INPUT\_path\_to\_your\_images);

//const INPUT\_path\_to\_your\_images = 'public/uploadedFile/\*\*/\*.{jpg,JPG,jpeg,JPEG,png,svg,gif}';

const OUTPUT\_path = 'public/compressedImages/';

compress\_images(INPUT\_path\_to\_your\_images, OUTPUT\_path, {compress\_force: false, statistic: true, autoupdate: true}, false,

{jpg: {engine: 'mozjpeg', command: ['-quality', '10']}},

//{jpg: {engine: 'jpegRecompress', command: ['--quality', 'high', '--min', '10']}},

{png: {engine: 'pngquant', command: ['--quality=20-50']}},

{svg: {engine: 'svgo', command: '--multipass'}},

{gif: {engine: 'gifsicle', command: ['--colors', '64', '--use-col=web']}},

function(err, completed, statistic){

if(completed)

{

console.log("Compress Process completed");

}

});

//

---------------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---------------

# Write/Read data to excel

## 1)exceljs

<https://www.npmjs.com/package/exceljs>

function exportCollectionExcelJS(item,res){

var Excel = require('exceljs');

var workbook = new Excel.Workbook();

var sheet1 = workbook.addWorksheet('My Sheet', {properties:{tabColor:{argb:'FFC0000'}}});

// create a sheet where the grid lines are hidden

var sheet2 = workbook.addWorksheet('My Sheet', {properties: {showGridLines: false}});

// create a sheet with the first row and column frozen

var sheet3 = workbook.addWorksheet('My Sheet', {views:[{xSplit: 1, ySplit:1}]});

var imageId1 = workbook.addImage({

filename: './public/compressedImages/john-volante-745624-unsplash.jpg',

extension: 'jpeg',

});

sheet1.columns = [

{ header: 'tourName', key: 'tourName', width: 20 },

{ header: 'tourPackage', key: 'tourPackage', width: 20 },

{ header: 'tourPrice.', key: 'tourPrice', width: 10 }

];

sheet1.addImage(imageId1, {

tl: { col: 1.5, row: 3 },

br: { col: 9, row: 9.5 }

});

workbook.xlsx.writeFile("exceljs1.xlsx")

.then(function() {

console.log("Excel creation done");

res.send("excel created exceljs");

});

}

## 2) excel4node

<https://www.npmjs.com/package/excel4node>

// Require library

var xl = require('excel4node');

// Create a new instance of a Workbook class

var wb = new xl.Workbook();

// Add Worksheets to the workbook

var ws = wb.addWorksheet('Project');

var ws2 = wb.addWorksheet('Asset');

// Create a reusable style

var style = wb.createStyle({

font: {

color: '#FF0800',

size: 12,

},

numberFormat: '$#,##0.00; ($#,##0.00); -',

});

// Set value of cell A1 to 100 as a number type styled with paramaters of style

ws.cell(1, 1)

.string('Tour Packages')

.style(style);

// Set value of cell B1 to 200 as a number type styled with paramaters of style

ws.cell(1, 2)

.string('$200')

.style(style);

// Set value of cell C1 to a formula styled with paramaters of style

ws.cell(1, 3)

.string('Tour desc')

.style(style);

// Set value of cell A2 to 'string' styled with paramaters of style

ws.cell(2, 1)

.string('project')

.style(style);

// Set value of cell A3 to true as a boolean type styled with paramaters of style but with an adjustment to the font size.

ws.cell(3, 1)

.bool(true)

.style(style)

.style({font: {size: 14}});

ws.addImage({

path: './public/compressedImages/john-volante-745624-unsplash.jpg',

type: 'picture',

position: {

type: 'twoCellAnchor',

from: {

col: 1,

colOff: 0,

row: 4,

rowOff: 0,

},

to: {

col: 6,

colOff: 0,

row: 9,

rowOff: 0,

},

},

});

//fpor worksheet 2

ws2.cell(1, 1)

.string('Tour Packages')

.style(style);

// Set value of cell B1 to 200 as a number type styled with paramaters of style

ws2.cell(1, 2)

.string('$200')

.style(style);

// Set value of cell C1 to a formula styled with paramaters of style

ws2.cell(1, 3)

.string('Tour desc')

.style(style);

// Set value of cell A2 to 'string' styled with paramaters of style

ws2.cell(2, 1)

.string('project')

.style(style);

// Set value of cell A3 to true as a boolean type styled with paramaters of style but with an adjustment to the font size.

ws2.cell(3, 1)

.bool(true)

.style(style)

.style({font: {size: 14}});

ws2.addImage({

path: './public/compressedImages/joshua-rawson-harris-804523-unsplash.jpg',

type: 'picture',

position: {

type: 'twoCellAnchor',

from: {

col: 1,

colOff: 0,

row: 4,

rowOff: 0,

},

to: {

col: 6,

colOff: 0,

row: 9,

rowOff: 0,

},

},

});

//

wb.write('Excel.xlsx');

res.send("excel created");

}

## 3) excel-export

[**https://www.npmjs.com/package/excel-export**](https://www.npmjs.com/package/excel-export)

<https://nnayek.wordpress.com/2014/09/17/export-database-data-to-excel-file-in-node-js-with-express/>

router.get('/tourRecord', function (req, res, next) {

var Promise = require('bluebird');

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

mongoClient.connectAsync('mongodb://localhost:27017', { useNewUrlParser: true })

.then(function (client) {

const db = client.db("learning\_mongo");

const collection = db.collection('tours');

return db.collection('tours').find({}).toArrayAsync();

})

.then(function (item) {

exportCollection(item,res);

//res.send("<b>"+item[0].tourBlurb+"</b>");

//res.send(item);

//res.json({"tour1":"US"});

//res.end(item);

})

.catch(function (err) {

console.log("err:" + err);

});

});

function exportCollection(item,res){

console.log("hey "+item);

var nodeExcel=require('excel-export');

var dateFormat = require('dateformat');

var conf={}

conf.cols=[{

caption:'Sl.',

type:'number',

width:3

},

{

caption:'Tour Name',

type:'Tour Package',

width:200

},

{

caption:'Date',

type:'string',

width:200

}

];

arr=[];

for(i=0;i<item.length;i++){

tourName=item[i].tourName;

tourPackage=item[i].tourPackage;

a=[i+1,tourName,tourPackage];

arr.push(a);

}

conf.rows=arr;

var result=nodeExcel.execute(conf);

res.setHeader('Content-Type','application/vnd.openxmlformates');

res.setHeader('Content-Disposition','attachment;filename='+'exportData.xlsx');

res.end(result,'binary');

}