Node.js express server

# example code (index.js)

|  |
| --- |
| const express = require('express');  const app = express();  const tests = require('./export');  const port = 8000; /\* port setting =>  Occasionally, permission denied problems may occur  depending on the setting of the user network,  and can be modified when a problem occurs.\*/  // To serve static files such as images, CSS files, and JavaScript files, use the express.static built-in middleware function in Express.  app.use(express.static('sample\_files'))  app.use(express.static('example\_html'))  // HTTP GET method route  app.get('/',(req, res) => {  res.send('hello Express (get)')  });  // HTTP POST method route  app.post('/',(req, res) => {  res.send('hello Express (post)')  });  /\*  Special route method (all) =>  Used to load middleware functions from one path for all request methods.  \*/  app.all('/secret',(req,res,next)=>{  console.log('Accessing the secret section ...');  next();  }, function(req,res){  console.log("secret request!");  res.send("secret key : gachon.ac.kr");  });  // chainable route handlers for a route path by using app.route().  // Because the path is specified at a single location,  // creating modular routes is helpful, as is reducing redundancy and typos.  app.route('/route')  .get(function(req,res){  res.send('/route - > get')  })  .post(function(req,res){  res.send('/route - > post')  })  .put(function(req,res){  res.send('/route - > put')  });  /\*Transfers the file at path as an “attachment”. Typically, browsers will prompt the user for download. By default,  the Content-Disposition header “filename=” parameter is path (this typically appears in the browser dialog).  verride this default with the filename parameter. \*/  app.route('/test\_download.jpg')  .get(function(req,res){  res.download('./sample\_files/gachon\_logo.png')  })  .post(function(req,res){  res.download('./sample\_files/gachon\_logo.png')  })  app.route('/test\_download2.jpg')  .get(function(req,res){  res.download('./sample\_files/gachon\_logo.png','changed\_name.jpg',function(err){  if(err){  console.log('error happen');  res.send('error happen sorry.');  }else{  console.log('transport complete');  }  })  })  .post(function(req,res){  res.download('./sample\_files/gachon\_logo.png','changed\_name.jpg',function(err){  if(err){  console.log('error happen');  res.send('error happen sorry.');  }else{  console.log('transport complete');  }  })  })  app.route('/test\_download\_err.jpg')  .get(function(req,res){  res.download('/sample\_files/gachon\_logo.png',function(err){  if(err){  console.log('error happen');  res.send('error happen sorry.');  }else{  console.log('transport complete');  }  })  })  .post(function(req,res){  res.download('/sample\_files/gachon\_logo.png',function(err){  if(err){  console.log('error happen');  res.send('error happen sorry.');  }else{  console.log('transport complete');  }  })  })  // bind ~/testing\_boom at tests (export.js)  app.use('/testing\_boom',tests)  // html data send.  app.route('/test\_home')  .get(function(req,res){  res.sendFile(\_\_dirname + "/example\_html/index.html")  })  .post(function(req,res){  res.sendFile(\_\_dirname + "/example\_html/index.html")  })  // bind and listen the connections on the specified host and port (port: 8000)  app.listen(port, ()=>{  console.log('express server listen...')  }); |

# example code (export.js)

|  |
| --- |
| var express = require('express')  var router = express.Router()  /\*  express.Router class -> create modular, mountable route handlers.  A Router instance is a complete middleware and routing system;  for this reason, it is often referred to as a “mini-app”.  \*/  // middleware that is specific to this router  router.use(function timeLog (req, res, next){  console.log('Time: ', Date.now())  next()  })  // define the exports home route.  router.get('/',function (req,res,next){  console.log('export route activated');  next()  }, function (req,res){  res.download('./sample\_files/IT.png','IT\_building.jpg',function(err){  if(err){  res.send('sorry server error happen');  console.log('klee download err');  }else{  console.log('klee download complete!');  }  })  })  // define the about route  router.get('/about', function(req,res){  res.send('About tests');  })  module.exports = router |

# **Result**

|  |
| --- |
|  |

|  |
| --- |
|  |