1. feladat  
npm init -y

npm I express

npm I nodemon –save-dev

package.jsonba beírni: “type”: “modul”, “start” : “node app.js” ;“dev” : “nodemon app.js”

szerver elindítása npm run dev  
  
2,3,4,5. feladat  
import \* from express

const Books = [{id: 1, title:”alma”, author:“Béla”}]

const port = 3000

const app = express()

app.use(express.json)

app.get(“books”, (req, res) => {

res.status(200).json(books)

})

app.post(“books”, (req, res) => {

const {title, author} = req.body

if(!title || !author)

{

return res.status(400).json({message : “hiányos adat.”})

}

const newId = +books.last().id+1

const newBook = {id: newId, title: title, author:author}

books.push(newBook)

res.status(201).json(newBook)

})

app.delete(“books/:id”, (req, res) => {

const selectedId = req.params.id

if(!id)

{

return res.status(400).json({message : “Nincs Id.”})

}

let book;

for(let i = 0; I < books.length; i++)

{

if(books[i].id == selectedId)

{

book = books[i]

}

}

if(!book)

{

return res.status(400).json({message : “Nincs ilyen könyv.”})

}

books.splice(book, 1)

res.status(200).json({message : “sikeresen törölve.”})

})

app.get(“books/:id”, (req, res) => {

const selectedId = req.params.id

if(!id)

{

return res.status(400).json({message : “Nincs Id.”})

}

let book;

for(let i = 0; I < books.length; i++)

{

if(books[i].id == selectedId)

{

book = books[i]

}

}

if(!book)

{

return res.status(400).json({message : “Nincs ilyen könyv.”})

}

res.status(200).json(book)

})

app.use(port, () =>{ console.log(`A szerver fut a ${port}-on`}))