

```
const router = express.Router()
router.route('/bears').get((req,res) => res.send(bears.list))
```

เอามาใส่แถม app.get ให้ router

สร้างใหม่ `npm init --y`

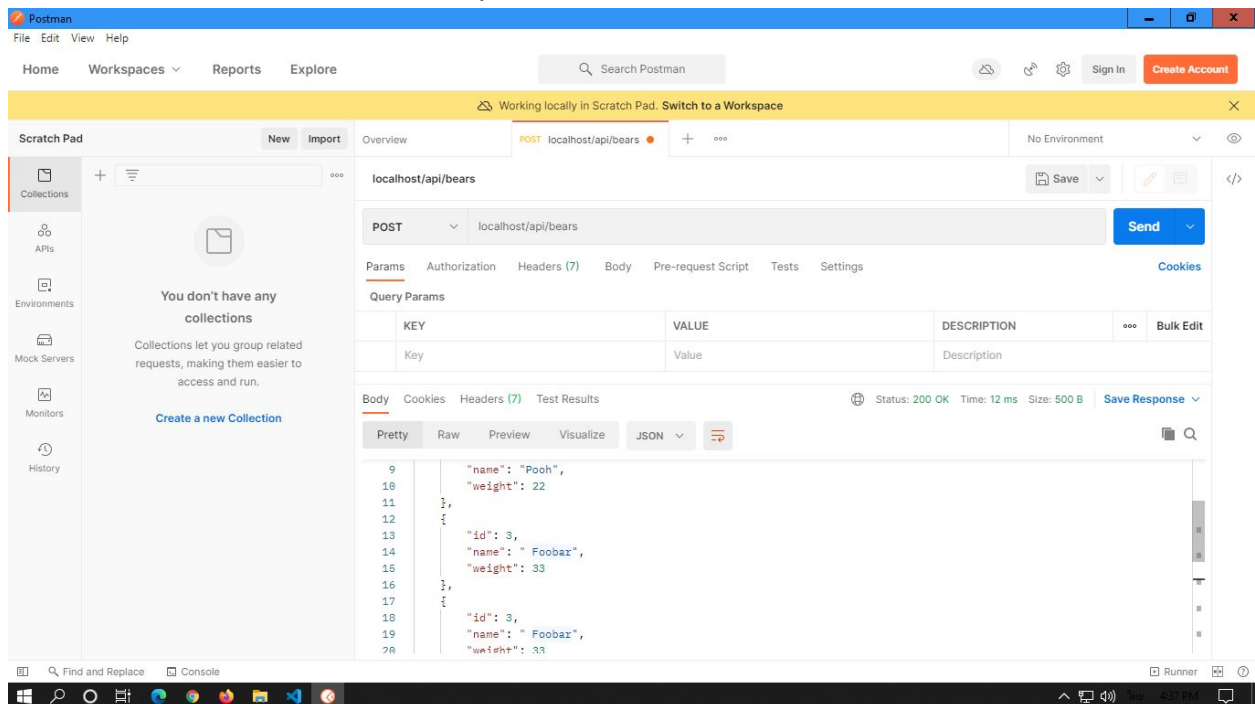
`npm i -s express`

สร้าง index.js เอง

## Post เพิ่มข้อมูล

```
.post((req, res) => {  
  bears.list =  
    [  
      ...bears.list, ข้อมูลเก่า บวกเพิ่มอันใหม่  
      { id: 3, name: " Foobar", weight: 33 }  
    ]  
  res.json(bears.list) ส่งค่าbears.listใหม่กลับไป  
})
```

ถ้าpost ในpostmanมันจะเพิ่ม id:3ให้เรื่อยๆ

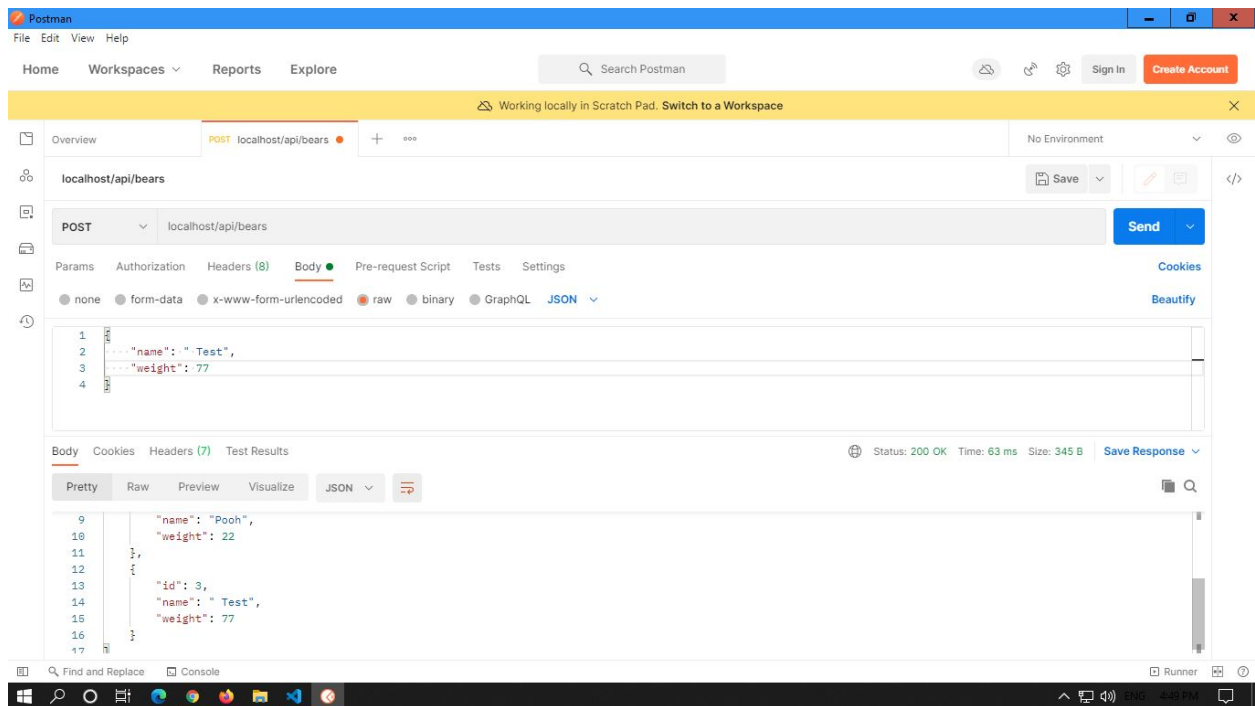


เพิ่มข้อมูลเอง รัน id เอง

```
let bodyParser = require('body-parser');

.post((req, res) => {

    let id =(bears.list.length)?
bears.list[bears.list.length-1].id+1:1
    let name = req.body.name
    let weight = req.body.weight
    bears.list =
    [
        ...bears.list,
        { id, name, weight }
    ]
    res.json(bears.list)
})
```



## Code ทั้งหมด

```
const express = require('express')
const app = express()
let bodyParser = require('body-parser');
const router = express.Router()
const PORT = 80

let bears = {
  list: [
    { id: 1, name: 'Winnie', weight: 99 },
    { id: 2, name: 'Pooh', weight: 22 }
  ]
}

app.use('/api', bodyParser.json(), router)

router.route('/bears')
  .get((req, res) => res.send(bears.list))
  .post((req, res) => {

    let id = (bears.list.length)?
bears.list[bears.list.length-1].id+1:1
    let name = req.body.name
    let weight = req.body.weight
    bears.list =
      [
        ...bears.list,
        { id, name, weight }
      ]
    res.json(bears.list)
  })

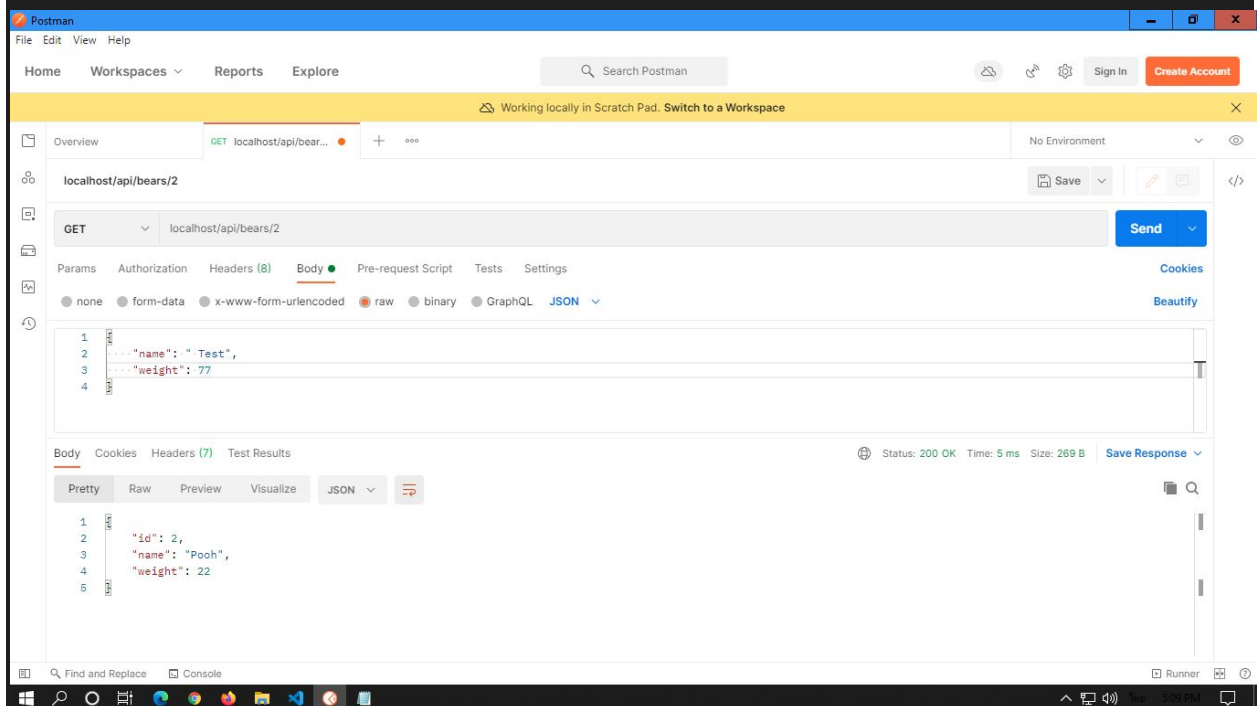
app.listen(PORT, () => console.log(`listen at ${PORT}`))
```

อย่าลืมรัน node index ใหม่ นะ

หาหมี

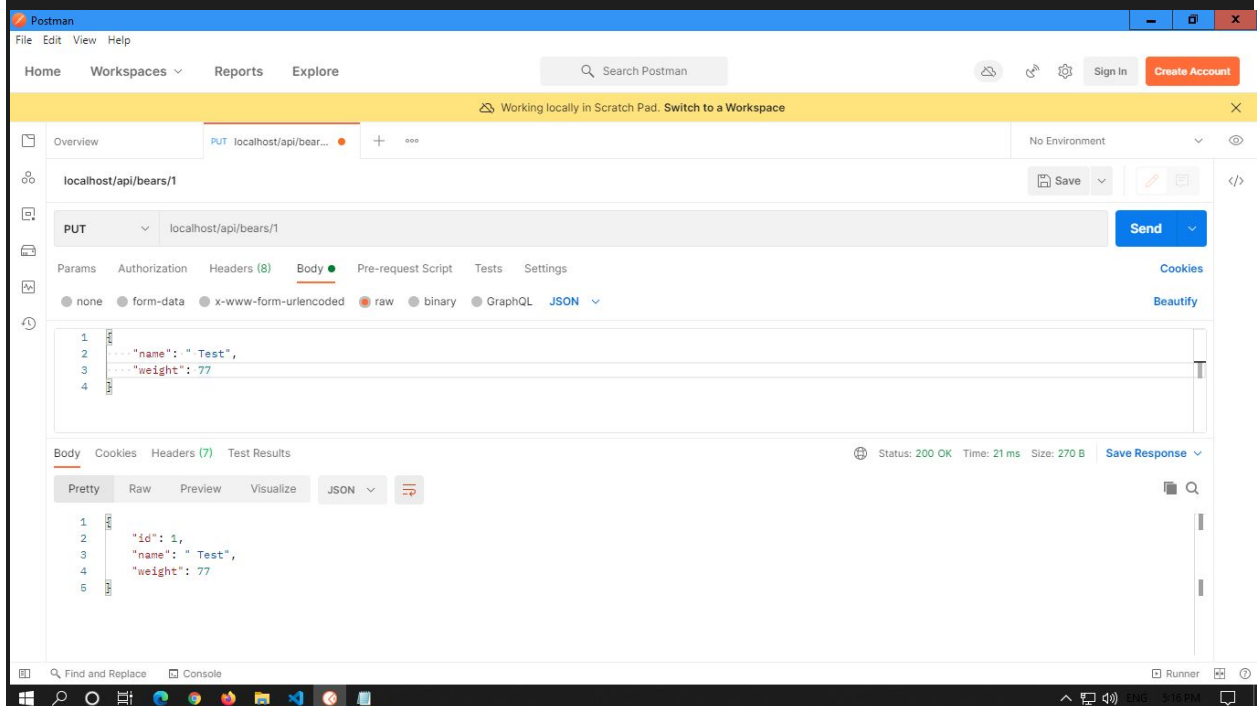
```
router.route('/:bear_id')
  .get((req, res) => {
    let id =bears.list.findIndex((item)=>item.id ===
+req.params.bear_id)
    res.json(bears.list[id])
  })
```

+req.params.bear\_id เป็นสตริงต้องใส่ + จะได้เป็นตัวเลข เป็นการแปลงเป็นตัวเลขง่ายๆ สั้นๆ  
.findIndex วนในอาร์เรย์



แก้หมี

```
router.route('/bears/:bear_id')
  .put((req, res) => {
    let id = bears.list.findIndex((item) => item.id ===
+req.params.bear_id)
    bears.list[id].name = req.body.name
    baars.list[id].weight = req.body.weight
    res.json(bears.list[id])
  })
```



Codeทั้งหมด

```
const express = require('express')
const app = express()
let bodyParser = require('body-parser');
const router = express.Router()
const PORT = 80

let bears = {
  list: [
    { id: 1, name: 'Winnie', weight: 99 },
    { id: 2, name: 'Pooh', weight: 22 }
  ]
}

app.use('/api', bodyParser.json(), router)

router.route('/bears')
  .get((req, res) => res.send(bears.list))
  .post((req, res) => {

    let id = (bears.list.length) ?
bears.list[bears.list.length-1].id+1:1
    let name = req.body.name
    let weight = req.body.weight
    bears.list =
      [
        ...bears.list,
        { id, name, weight }
      ]
    res.json(bears.list)
  })

router.route('/bears/:bear_id')
  .get((req, res) => {
    let id = bears.list.findIndex((item) => item.id ===
+req.params.bear_id)
    res.json(bears.list[id])
  })
```

```
.put((req, res) => {  
  let id =bears.list.findIndex((item)=>item.id ===  
+req.params.bear_id)  
  bears.list[id].name = req.body.name  
  bears.list[id].weight = req.body.weight  
  res.json(bears.list[id])  
})  
app.listen(PORT, () => console.log(`listen at ${PORT}`))
```



ลบหมี

Bears.list.filter ลบเงื่อนไขที่ไม่เป็นจริง ที่เป็นจริงจะเก็บไว้

```
.delete((req, res) => {  
  let id = req.params.bear_id  
  bears.list = bears.list.filter(item => +item.id !== +id)  
  res.send(bears.list)  
})
```

โค้ดทั้งหมด

```
const express = require('express')
const app = express()
let bodyParser = require('body-parser');
const router = express.Router()
const PORT = 80

let bears = {
  list: [
    { id: 1, name: 'Winnie', weight: 99 },
    { id: 2, name: 'Pooh', weight: 22 }
  ]
}

app.use('/api', bodyParser.json(), router)

router.route('/bears')
  .get((req, res) => res.send(bears.list))
  .post((req, res) => {

    let id =(bears.list.length)?
bears.list[bears.list.length-1].id+1:1
    let name = req.body.name
    let weight = req.body.weight
    bears.list =
      [
        ...bears.list,
        { id, name, weight }
      ]
    res.json(bears.list)
  })

router.route('/bears/:bear_id')
  .get((req, res) => {
    let id =bears.list.findIndex((item)=>item.id ===
+req.params.bear_id)
```

```
        res.json(bears.list[id])
    })
    .put((req, res) => {
        let id =bears.list.findIndex((item)=>item.id ===
+req.params.bear_id)
        bears.list[id].name = req.body.name
        bears.list[id].weight = req.body.weight
        res.json(bears.list[id])
    })
    .delete((req, res) => {
        let id = req.params.bear_id
        bears.list = bears.list.filter(item => +item.id !== +id)
        res.send(bears.list)
    })
app.listen(PORT, () => console.log(`listen at ${PORT}`))
```

การบ้าน ที่ถ้ากรอกเลขไม่เจอให้ขึ้น not found

```
const express = require('express')
const app = express()
let bodyParser = require('body-parser');
const router = express.Router()
const PORT = 80

let students = {
  list: [
    { id: 6035512072, name: 'Panjira', gpa: 2.01 },
    { id: 6035512071, name: 'Chamrim', gpa: 3.10 }
  ]
}

app.use('/api', bodyParser.json(), router)

router.route('/students')
  .get((req, res) => res.send(students.list))
  .post((req, res) => {

    let id =(students.list.length)?
students.list[students.list.length-1].id+1:1
    let name = req.body.name
    let gpa = req.body.gpa
    students.list =
      [
        ...students.list,
        { id, name, gpa }
      ]
    res.json(students.list)
  })

router.route('/students/:students_id')
  .get((req, res) => {
    let id =students.list.findIndex((item)=>item.id ===
+req.params.students_id)
    if(id < 0){
      res.json('No found')
```

```

    }
    else{
        res.json(students.list[id])
    }

}))

.put((req, res) => {
    let id = students.list.findIndex((item)=>item.id ===
+req.params.students_id)
    if(id < 0){
        res.json('No found')
    }
    else{
        students.list[id].name = req.body.name
        students.list[id].gpa = req.body.gpa
        res.json(students.list[id])
    }

}))

.delete((req, res) => {
    let id = req.params.students_id
    if(students.list.findIndex((item)=>item.id === +id)<0)
    {
        res.json('No found')
    }
    else{
        students.list = students.list.filter(item => +item.id !== +id)
        res.send(students.list)
    }

}))

app.listen(PORT, () => console.log(`listen at ${PORT}`))

```