

Creamos un nuevo proyecto en Visual Studio Code e instalamos los módulos necesarios para el proyecto (typescript, mysql y express, este último es un conjunto de apis que facilitan la creación de aplicaciones en NodeJS):

```
npm install typescript
```

```
npm install mysql
```

```
npm install express
```

Configuramos tsconfig.json

```
{
  "compilerOptions": {
    /* Basic Options */
    "target": "es6",
    "module": "commonjs",
    "outDir": "./dist",
    "strict": false,
    "resolveJsonModule": true,
    "esModuleInterop": true
  },
  "include": [
    "./src/**/*.ts",
    "./tests/**/*.ts",
  ]
}
```

Creamos el archivo package.json

```
npm init -y
```

E instalamos todas las dependencias (en caso de que faltase alguna):

```
npm install
```

Creamos la carpeta src folder y el archivo index.ts

```
let DDBB = require("mysql");

let con = DDBB.createConnection({
  host: "localhost",
  user: "test",
  password: "",
  database: "api"
});

con.connect(function(err) {
  if (err) throw err;

});

// Load modules y create app
```

```
let express = require("express");
let app = express();
let bodyParser = require('body-parser');
app.use(bodyParser.json()); // JSON supported
app.use(bodyParser.urlencoded({ extended: true }));

//Sample: GET http://localhost:8080/items
app.get('/items', function(req, res, next) {
  if(req.query.title) {
    next();
    return;
  }
  let sql="SELECT * FROM items";
  con.query(sql, function (err, result) {
    if (err) throw err;
    console.log(result);
    res.send(result);
  });
});

//Sample: GET http://localhost:8080/items?title=ABC
app.get('/items', function(req, res) {
  let filter = req.query.title;
  let sql="SELECT * FROM items WHERE title='"+filter+"'";
  con.query(sql, function (err, result) {
    if (err) throw err;
    console.log(result);
    res.send(result);
  });
});

//Ejemplo: GET http://localhost:8080/items/10
app.get('/items/:id', function(req, res, next) {
  let itemId = req.params.id;
  let sql="SELECT * FROM items WHERE id="+itemId;
  con.query(sql, function (err, result) {
    if (err) throw err;
    console.log(result);
    res.send(result);
  });
});

//Sample: POST http://localhost:8080/items
app.post('/items', function(req, res) {
  let body = req.body;
```

```

    let sql='INSERT INTO items SET ?';
    con.query(sql, body, (err, res2) => {
        if(err) throw err;
        body['id']=res2.insertId;
        console.log(res2);
        res.send(body);
    });

});

//Ejemplo: PUT http://localhost:8080/items
app.put('/items/:id', function(req, res) {
    let itemId = req.params.id;
    let data = [req.query.title, req.query.description]
    let sql='UPDATE items SET title = ?,description= ? Where ID = ?';
    con.query(sql, [...data, itemId], (err, res2) => {
        if(err) throw err;
        res.send({id:itemId,...req.body});
    });
});

//Sample: DELETE http://localhost:8080/items
app.delete('/items/:id', function(req, res) {
    let itemId = req.params.id;
    let sql='DELETE FROM items Where ID = ?';
    con.query(sql, itemId, (err, res2) => {
        if(err) throw err;
        console.log(res2);
        res.send({id:itemId});
    });
});

var server = app.listen(8080, function () {
    console.log('Server is running..');
});

```