

# NODE.JS

## 1. Que es NODE.js?

- Node.js és un entorn d'execució per JavaScript construït sobre el motor V8 de Chrome. Permet executar codi JavaScript fora del navegador, la qual cosa fa possible utilitzar JavaScript també al costat del servidor, proporcionant una plataforma eficient per a la creació d'aplicacions web, serveis, i molt més.

### Exemple de codi bàsic d'una web amb Node.js

```
const http = require('http');

const servidor = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hola, món!');
});

const port = 3000;
servidor.listen(port, () => {
  console.log(`El servidor s'executa a http://localhost:${port}`);
});
```

## 2. Per què es va crear Node.js?

- Node.js es va crear amb l'objectiu de permetre l'execució de codi JavaScript fora del navegador aprofitant la velocitat i l'eficiència del motor V8 de Chrome. Això permet utilitzar JavaScript per a crear aplicacions del costat del servidor, aconseguint un entorn ràpid i escalable.

## 3. Per a què serveix Node.js?

- Servidors web
- Aplicacions en temps real
- API i microserveis
- Eines de línia de comandes
- Aplicacions de xarxa

## Instal·lar Node.js:

- Obre el terminal (Ctrl + Alt + T).
- Crea la carpeta del projecte: mkdir incidència.
- Accedeix a la carpeta: cd incidència.
- Inicialitza el projecte Node.js: npm init -y.

## Instal·lar Express:

- Instal·la Express: npm install express.

## Estructura del projecte:

- Crea el fitxer principal: touch app.js.
- Crea una carpeta per a les dades: mkdir data.
- Crea el fitxer de dades de motocicletes: touch data/motocicletes.json.
- Inicialitza motocicletes.json amb un array buit: [ ].

## Importació de mòduls i configuració inicial

Es crea una aplicació Express i es configura per gestionar sol·licituds amb cos en format JSON.

```
const express = require('express');
const path = require('path');
const sequelize = require('./db');
const app = express();
const incidenciaRoutes = require('./routes/incidencies.routes');

// Configurar EJS como motor de plantillas
app.set('view engine', 'ejs');

app.set('views', path.join(__dirname, 'views')); // Ruta a tus vistas

// Rutas de incidencias
app.use('/incidencias', incidenciaRoutes);

// Ruta de prueba
app.get('/', (req, res) => {
  res.render('index'); // Renderiza el archivo index.ejs
});
```

```
// Iniciar el servidor
const port = process.env.PORT || 3000;
app.listen(port, () => {
  console.log(`Servidor escuchando en http://localhost:${port}`);
});
```

## Funcions per gestionar les dades

- llegirDades(): Llegeix les dades del fitxer JSON (motocicletes.json).
- escriureDades(dades): Escriu les dades al fitxer JSON

```
function llegirDades() {
  const dadesJSON = fs.readFileSync('./routes/incidencies.routes.json', 'utf8');
  return JSON.parse(dadesJSON);
}

function escriureDades(dades) {
  const dadesJSON = JSON.stringify(dades, null, 2);
  fs.writeFileSync('./routes/categories.routes.json', dadesJSON);
}
```

## Rutes de l'API

- POST /motocicletes: Crea una nova motocicleta.
- GET /incidencies: Obté totes les incidencies.
- GET /incidencies/:id: Obté una incidencies per ID.
- PUT /incidencies/:id: Actualitza una incidencies per ID.
- DELETE /incidencies/:id: Elimina una incidencies per ID.

### Exemple de ruta POST per crear una motocicleta:

```
router.post('/create', async (req, res) => {
  try {
    const { nom, tipus, descripcio } = req.body; // Los campos que el usuario
    ingresa

    // Rellenar los campos adicionales
    const dataCreacio = new Date(); // Fecha actual para la creación
    const estat = "Pendent d'assignacio"; // El estado inicial por defecto
    const dataResolucio = null; // La fecha de resolución está vacía
    inicialmente

    // Crear la incidencia en la base de datos
    await Incidencia.create({
      nom,
      tipus,
      descripcio,
```

```
dataCreacio,  
estat: estat,  
dataResolucio: dataResolucio, // Solo los campos necesarios  
});  
  
res.redirect('/incidencias'); // Redirigir al listado de incidencias  
} catch (error) {  
  console.error('Error al crear la incidencia:', error);  
  res.status(500).send("No s'ha pogut crear l'incidencia");  
}  
});
```

## Iniciar el servidor

- S'inicia el servidor Express per escoltar a <http://localhost:3000>.