

Número i nom del curs		IFC154_3 - Desenvolupament d'aplicacions amb tecnologies web
Mòdul Formatiu que s'avalua		MF0492_3: Programació web a l'entorn servidor
✓	Unitat Formativa que s'avalua	UF1844: Desenvolupament d'aplicacions web a l'entorn servidor

Nom i Cognoms de l'alumne/a	
NIF de l'alumne/a	
Data de la prova	
Signatura	

INSTRUCCIONS DE LA PROVA
<ul style="list-style-type: none"> ▪ <i>En aquest full trobarà la informació necessària per a la realització de la prova.</i> ▪ <i>Abans de resoldre la pràctica llegeixi amb atenció l'enunciat de l'exercici i comprovi que té tots els materials i equipament necessari.</i> ▪ <i>Per qualsevol aclariment consulti a l'avaluador/ i/o formador/a.</i> <p><i>Cas de ser necessari, utilitzi els equips de protecció individual necessaris per la realització de la pràctica tenint en compte les normes de seguretat i higiene.</i></p>

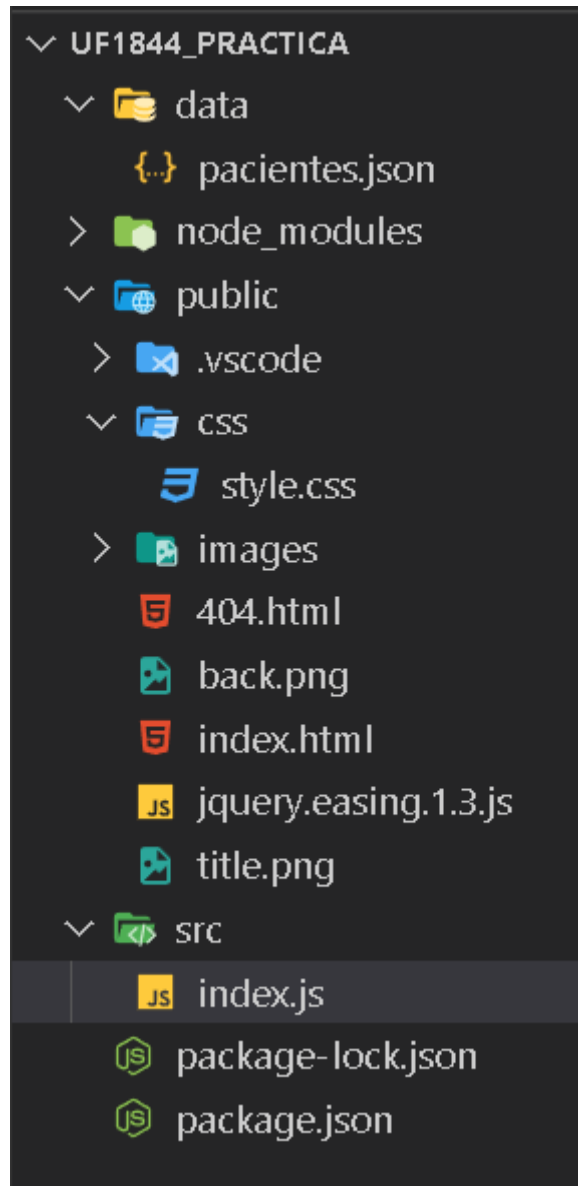
DESCRIPCIÓ DE LA PRÀCTICA	
DENOMINACIÓ	Utilització de les eines NODE per a gestionar informació
ESPECIFICACIONS TÈCNIQUES	<i>En el text següent</i>
MATERIAL	<i>En fitxers adjunts</i>
TEMPS	<i>3 hores</i>

Tenemos un encargo de una clínica, para gestionar datos de sus servicios.

Disponemos de un conjunto de ficheros estáticos y un JSON con datos de pacientes. Tienes que:

1. Preparativos:

Crear una estructura como esta:



Crear una configuración de inicio (package.json).

Debes usar express para levantar el servidor.

El programa se iniciará así

```
a\UF1844_practica>node src/index 3000
```

El tercer parámetro es el puerto de conexión. No uses yargs.

Ejercicios API

En los ejemplos se muestra el puerto 3000 pero puede ser otro según se inicie el programa.

2. localhost:3000/api/pacientes
Mostrará todos los pacientes ordenados alfabéticamente por apellido de la A a la Z, pero solo su apellido y nombre, de este modo:
Abnuy, Rafael
Adam, Nathaniel
....
¡Ojo! No es un JSON...

3. localhost:3000/api/servicio/urgencias o localhost:3000/servicio/consultas
Crear un filtro por tipo de servicio, que se añadirá a la url:
localhost:3000/servicio.

Si se introduce un servicio inexistente debe aparecer el mensaje:

“No existe ningun servicio con ese nombre” (sin las comillas)

4. localhost:3000/api/pacientes/nif
Filtrar los pacientes según su nif, mostrando sus datos completos en formato JSON. Por ejemplo:
localhost:3000/api/pacientes/26902806M
Si se introduce un nif inexistente debe aparecer el mensaje:
“No existe ningun paciente con ese nif” (sin las comillas)

5. localhost:3000/api/ciudades
Lista de pacientes en formato JSON ordenados por ciudades.

6. localhost:3000/api/ciudades/ciudad
Filtrar los pacientes según su ciudad, mostrando sus datos completos en formato JSON. Por ejemplo:
localhost:3000/api/ciudades/Barcelona
Si se introduce una ciudad inexistente debe aparecer el mensaje:
“No existe consta ningun paciente de esa ciudad” (sin las comillas)

7. localhost:3000/api/pacientes-ciudad
Contar los pacientes por ciudad .
Si se introduce una ciudad inexistente debe aparecer el mensaje:
“No existe consta ningun paciente de esa ciudad” (sin las comillas)

8. Mostrar la página 404.html cuando se introduza una url inexistente.

No te olvides de poner el nombre al principio del fichero index.js

Comenta el código de forma adecuada. No hay que escribir una novela, solo indicar que hacen las partes más importantes del código (constantes, variables, métodos, funciones...)

Puntuació	0	0.5	1
Per a cada pregunta	Exercici no resolt o amb errors greus	Exercici no completament resolt i/o amb errors lleus	Exercici perfectament resolt
Globalment	Codi molt mal organitzat	Codi ordenat en general, amb problemes lleus	Codi net i ordenat
Globalment	Codi no comentat o fet de manera negligent	Codi comentat, encara que amb alguna explicació poc clara	Codi perfectament comentat
Nombre de respostes segons l'avaluació			
Resultat			
Total puntuació			

OBSERVACIONS**Signatura formador/a**
Signatura i segell del centre de formació