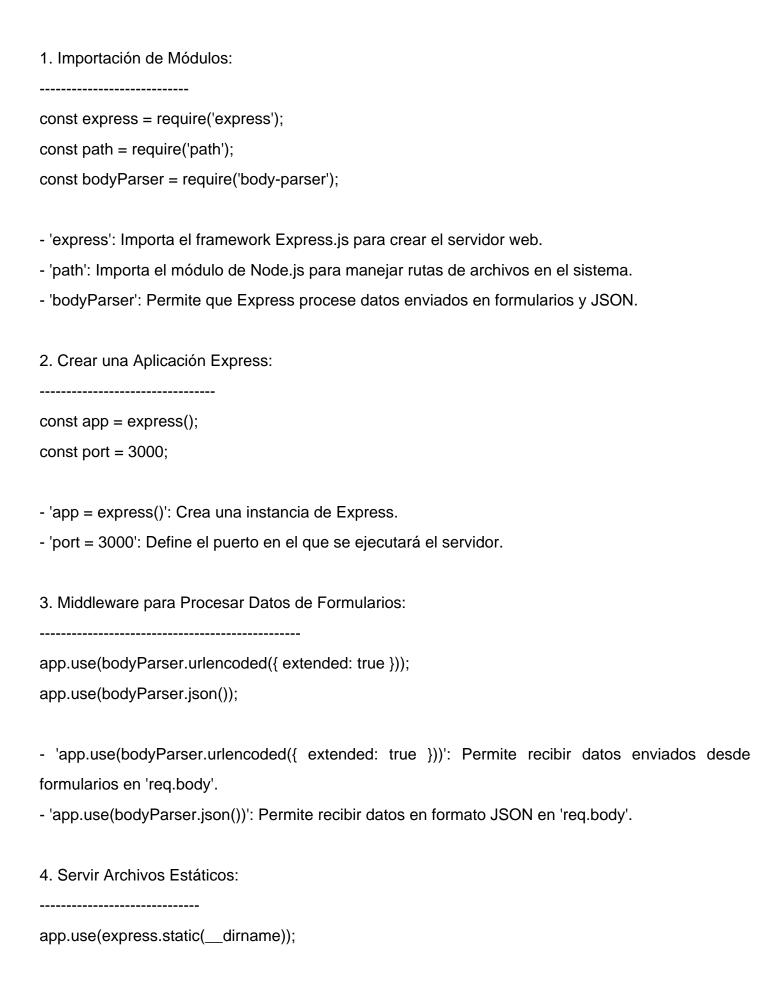
## Explicación Detallada del Codigo server.js



- 'express.static(dirname)': Permite que Express sirva archivos estáticos como 'index.html', CSS, JS, imágenes, etc.
5. Ruta para Servir el Formulario:
<pre>app.get('/validanodejs', (req, res) =&gt; {   res.sendFile(path.join(dirname, 'index.html')); });</pre>
<ul> <li>'app.get('/validanodejs', (req, res) =&gt; {})': Define una ruta GET en 'http://localhost:3000/validanodejs'.</li> <li>'res.sendFile(path.join(dirname, 'index.html'))': Envía el archivo 'index.html' al navegador.</li> </ul>
6. Ruta para Recibir los Datos del Formulario:
app.post('/recibir-datos', (req, res) => {
- 'app.post('/recibir-datos', (req, res) => {})': Define una ruta POST que se activa cuando se envía el formulario.
7. Capturar Datos del Formulario:
const { nombre, contra, fnac, antig, estat, email, precio } = req.body;
- Desestructura los valores del formulario desde 'req.body'.
8. Imprimir los Datos en la Terminal:
console.log('Datos recibidos:', { nombre, contra, fnac, antig, estat, email, precio });
- Muestra los datos en la terminal cuando se recibe el formulario.

```
9. Responder con los Datos en HTML:
res.send(`
  <h2>Datos Recibidos en Node.js</h2>
  <strong>Nombre:</strong> ${nombre}
  <strong>Contraseña:</strong> ${contra}
  <strong>Fecha de Nacimiento:</strong> ${fnac}
  <strong>Antigüedad:</strong> ${antig} años
  <strong>Estatura:</strong> ${estat} metros
  <strong>Email:</strong> ${email}
  <strong>Precio:</strong> ${precio}
  <br>
  <a href="/validanodejs">Volver al Formulario</a>
`);
- Genera una página HTML con los datos recibidos y los muestra en el navegador.
10. Iniciar el Servidor:
-----
app.listen(port, () => {
  console.log(`Servidor corriendo en http://localhost:${port}/validanodejs`);
});
- Inicia el servidor en el puerto 3000.
```

- Muestra la URL en la terminal.