

REACT 18+ • TYPESCRIPT • NEXT.JS

DOMINA REACT

La guía definitiva para principiantes y desarrolladores
que quieren dominar React de forma profesional

■ Hooks

■ Componentes

■ Estado

■ Redux

■ Next.js

■ Testing

- ✓ 14 capítulos progresivos desde cero
- ✓ +60 ejemplos de código listos para usar
- ✓ Proyectos reales: ToDo, Dashboard, E-commerce
- ✓ Ejercicios prácticos en cada capítulo
- ✓ Patrones de código de nivel industria

De CERO a PROFESIONAL en React

Incluye acceso a código fuente • Actualizaciones gratuitas • Comunidad privada



CAPÍTULO 00

Introducción — ¿Por qué React?

El punto de partida para tu viaje como desarrollador React

00

Bienvenido a **Domina React**. Si estás leyendo esto, ya diste el primer paso más importante: decidiste aprender la biblioteca más popular del mundo para construir interfaces de usuario. Este libro fue diseñado para llevarte desde cero hasta un nivel profesional, con ejemplos reales y proyectos que puedes mostrar en tu portafolio desde hoy.

¿Por qué aprender React en 2026?

React sigue siendo la opción número uno en el mercado laboral. Según la encuesta de Stack Overflow 2024, React es el framework/biblioteca web más usado por quinto año consecutivo, con más del 40% de los desarrolladores web utilizándolo en sus proyectos.

- **Mercado laboral:** Más del 60% de las ofertas de trabajo front-end piden React.
- **Salario:** Los desarrolladores React ganan entre un 20-35% más que otros front-end.
- **Ecosistema:** Next.js, React Native, Remix — un skill que escala a múltiples plataformas.
- **Comunidad:** Millones de desarrolladores, documentación excelente, soporte activo.
- **Meta + Vercel:** Respaldado por empresas que invierten millones en su desarrollo.

NOTA

No necesitas saber JavaScript avanzado para empezar este libro. Solo necesitas conocer variables, funciones, arrays y objetos básicos. Si dudas, revisa el Apéndice de prerequisites antes de continuar.

Cómo usar este libro

Cada capítulo sigue la misma estructura para que tu aprendizaje sea progresivo y consistente:

- ■ **Concepto explicado** — Teoría clara y sin jerga innecesaria.
- ■ **Código comentado** — Ejemplos que puedes copiar y ejecutar ahora mismo.
- ■ **Consejos pro** — Lo que los tutoriales de YouTube no te cuentan.
- ■ ■ **Errores comunes** — Los bugs que todos cometen y cómo evitarlos.
- ■ ■ **Ejercicios** — Práctica guiada al final de cada sección.
- ■ ■ **Proyecto** — Capítulos clave incluyen un mini-proyecto real.

Configuración del entorno de desarrollo



Terminal — Configuración inicial

```
# 1. Instalar Node.js (v18 o superior recomendado)
# Descarga desde: https://nodejs.org

# 2. Verificar instalación
node --version    # Debe mostrar v18.x o superior
npm --version     # Debe mostrar 9.x o superior

# 3. Crear tu primer proyecto con Vite (recomendado)
npm create vite@latest mi-primer-react -- --template react
cd mi-primer-react
npm install
npm run dev
# Abre http://localhost:5173 en tu navegador

# 4. Extensiones de VS Code recomendadas:
# - ES7+ React/Redux/React-Native snippets
# - Prettier - Code formatter
# - ESLint
# - Auto Rename Tag
```

■ RESUMEN DEL CAPÍTULO

- ✓ React es la biblioteca más demandada del mercado laboral front-end
- ✓ Este libro usa React 18+, Vite y TypeScript como stack base
- ✓ Cada capítulo incluye ejemplos, consejos y ejercicios prácticos
- ✓ Los proyectos reales te permitirán construir un portafolio profesional



CAPÍTULO 01

Fundamentos de React

JSX, Virtual DOM, componentes básicos y las reglas del juego

01

Antes de escribir tu primera línea de React, necesitas entender los conceptos sobre los que todo lo demás se construye. No te saltes este capítulo: los malentendidos sobre JSX y el Virtual DOM causan el 80% de los bugs que confunden a los principiantes.

¿Qué es el Virtual DOM?

El DOM (Document Object Model) es la representación en memoria del HTML de tu página. Manipularlo directamente es lento porque cada cambio puede forzar al navegador a recalcular estilos, layouts y repintados. React resuelve esto con el **Virtual DOM**: una copia ligera del DOM real que vive en memoria.

Cuando el estado de tu app cambia, React: **(1)** crea un nuevo Virtual DOM, **(2)** lo compara con el anterior (proceso llamado *diffing*), **(3)** calcula los cambios mínimos necesarios y **(4)** actualiza solo esos nodos en el DOM real.

JSX — JavaScript con superpoderes

JSX — Sintaxis completa

```
// JSX parece HTML pero es JavaScript transformado por Babel
const elemento = (
  <div className="tarjeta" onClick={handleClick}>
    <h1>Hola, {nombre}</h1>          {/* {} = expresión JS */}
    <p>Tienes {calcularEdad()} años</p>
    <img src={usuario.foto} alt="Foto de perfil" />
  </div>
);

// REGLAS ESENCIALES DE JSX:
// 1. Un solo elemento raíz (o Fragment <>)
const ok = <><h1>Hola</h1><p>Mundo</p></>;

// 2. class → className, for → htmlFor
<div className="contenedor">
  <label htmlFor="email">Email:</label>
</div>

// 3. Expresiones JS entre llaves (no statements)
<p>{activo ? 'Activo' : 'Inactivo'}</p>      // ternario ■
<p>{lista.map(i => <span key={i.id}>{i}</span>)}</p> // map ■
```

¡ATENCIÓN!

La propiedad 'key' es obligatoria cuando renderizas listas con .map(). React la usa internamente para optimizar actualizaciones. Usa siempre un identificador único como key, NUNCA el índice del array.



Tu primer componente real

Tu primer componente — TarjetaUsuario

```
import { useState } from 'react';

function TarjetaUsuario({ nombre, email, rol, avatar }) {
  const [seguido, setSeguido] = useState(false);

  return (
    <div className="tarjeta-usuario">
      <img src={avatar || '/default-avatar.png'} alt={`Foto de ${nombre}`} />
      <div className="info">
        <h2>{nombre}</h2>
        <p>{email}</p>
        <span className={`badge badge--${rol}`}>{rol}</span>
      </div>
      <button
        onClick={() => setSeguido(prev => !prev)}
        className={`btn ${seguido ? 'btn--seguido' : 'btn--seguir'}`}
      >
        {seguido ? '✓ Siguiendo' : '+ Seguir'}
      </button>
    </div>
  );
}

// Uso del componente
function App() {
  return (
    <TarjetaUsuario
      nombre="Maria García"
      email="maria@ejemplo.com"
      rol="admin"
    />
  );
}

export default App;
```

Renderizado condicional



Renderizado condicional — 3 patrones

```
function Panel({ usuario, loading, error }) {
  // Early return para múltiples estados
  if (loading) return <Spinner />;
  if (error) return <ErrorMessage mensaje={error.message} />;
  if (!usuario) return <p>No encontrado</p>;

  return <PerfilUsuario usuario={usuario} />;
}

function Notificaciones({ items }) {
  return (
    <div>
      /* && solo renderiza si la condición es true */
      {items.length > 0 && (
        <span className="badge">{items.length}</span>
      )}
    </div>
  );
}
```

 ■ EJERCICIO 1 — Componente de Perfil

1. Crea un componente Perfil que reciba nombre, bio y skills (array).
2. Muestra el nombre en un h1 y la bio en un párrafo.
3. Renderiza la lista de skills con .map() usando etiquetas para cada una.
4. Agrega un botón 'Editar' que solo aparezca si el usuario es el propietario.
5. BONUS: Agrega un estado local para mostrar/ocultar la bio.

■ RESUMEN DEL CAPÍTULO

- ✓ JSX es JavaScript transformado — usa className no class; htmlFor no for
- ✓ El Virtual DOM hace que React sea eficiente al actualizar solo lo necesario
- ✓ Los componentes son funciones en PascalCase que retornan JSX
- ✓ Renderizado condicional: ternario para dos opciones, && para una, early return para múltiples
- ✓ La prop key es obligatoria en listas — usa siempre un ID único



¿Te gustó esta muestra? ■

El libro completo incluye 17 capítulos, 3 proyectos reales y todo lo que necesitas para conseguir trabajo como desarrollador React.

Lo que te esperan en el libro completo:

- ✓ 02 — Componentes en Profundidad
- ✓ 03 — Estado con useState
- ✓ 04 — Efectos con useEffect
- ✓ 05 — Hooks Esenciales
- ✓ 06 — Hooks Avanzados
- ✓ 07 — Props y Comunicación
- ✓ 08 — Context API
- ✓ 09 — Estado Global (Redux/Zustand)
- ✓ 10 — React Router v6
- ✓ 11 — Formularios Profesionales
- ✓ 12 — Testing con Jest y RTL
- ✓ 13 — Rendimiento y Optimización
- ✓ 14 — Patrones Avanzados
- ✓ 15 — TypeScript con React
- ✓ 16 — Next.js en Producción
- ✓ 17 — Proyecto Final E-commerce

Consigue el libro completo en leanpub.com/domina-react