

# WORDPRESS

## Parte 1 — Che cos'è WordPress

Concetti chiave:

- WordPress è un **CMS (Content Management System)** scritto in PHP.
- Serve per creare siti web, blog, ecommerce, vetrine, forum ecc.
- Oggi gestisce oltre il **40% dei siti al mondo**.
- Ci sono due versioni:
  1. **WordPress.com** → hosting gestito (limitato per sviluppatori).
  2. **WordPress.org** → software libero da installare sul proprio hosting (quello “vero” usato da professionisti).

Perché è importante:

- Non devi programmare tutto da zero.
  - Puoi espandere il sito con **plugin** e cambiare design con **temi**.
- 

## Parte 2 — Come funziona tecnicamente WordPress

Tecnologie usate:

- PHP (logica backend).
- MySQL/MariaDB (database).
- HTML/CSS/JS (uscita verso il browser).
- Apache o Nginx (server).

WordPress è composto da:

- File core → /wp-admin , /wp-includes , /wp-content
- Database con tabelle come:
  - wp\_posts
  - wp\_users
  - wp\_postmeta
  - wp\_options
  - ecc.

Il cuore della struttura è la tabella `wp_posts`:

- ✓ contiene pagine
  - ✓ blog post
  - ✓ prodotti WooCommerce
  - ✓ media
  - ✓ menu
  - ✓ qualsiasi “contenuto”
- 

## Parte 3 — Installazione

Per ambiente locale:

- XAMPP, MAMP o WAMP
- Crei database → apri `localhost/wordpress` → installazione guidata.

Per hosting:

- Molti provider hanno “installazione 1-click”.
- Carichi file via FTP + crei DB + esegui installazione.

Dati richiesti:

- Nome database
  - Username
  - Password
  - Host (di solito `localhost`)
- 

## Parte 4 — Dashboard WordPress

Elementi principali del pannello di controllo:

- **Bacheca** → riepilogo sito
- **Articoli** → blog
- **Media** → immagini, file
- **Pagine** → pagine statiche
- **Aspetto** → tema, menu, widget
- **Plugin** → nuove funzionalità
- **Utenti** → ruoli (Admin, Editor, Autore, Collaboratore, Sottoscrittore)

- **Impostazioni** → titolo sito, permalink, lingua, ecc.
- 

## Parte 5 — Temi

I temi controllano:

- layout
- colori
- font
- header / footer
- struttura generale del sito

Tipi di tema:

1. **Temi classici**
2. **Temi a blocchi (Full Site Editing)** — es. Twenty Twenty-Four
3. **Temi builder** → GeneratePress, Astra, OceanWP

Un tema **non** dovrebbe aggiungere funzionalità pesanti → quelle vanno nei plugin.

---

## Parte 6 — Page Builder

Molti siti moderni usano builder:

- Elementor
- Divi
- Beaver Builder
- Bricks Builder
- Gutenberg (nativo WordPress)

A cosa servono?

- Trascini blocchi
  - Costruisci pagine “a mano”
  - Crei layout complessi senza codice
-

## Parte 7 — Plugin

Un plugin è un “pezzo di software” che aggiunge funzionalità.

Esempi utili:

- **Yoast SEO / RankMath** → SEO
- **WooCommerce** → ecommerce
- **Contact Form 7** → moduli di contatto
- **WP Super Cache / LiteSpeed Cache** → performance
- **All in One Security** → sicurezza
- **UpdraftPlus** → backup

Regola fondamentale:

**Non installare 40 plugin inutili — rallentano o creano conflitti.**

---

## Parte 8 — WordPress e MySQL

Come salva i dati:

- Ogni contenuto è un “post” nella tabella `wp_posts`.
- Le informazioni extra (campi aggiuntivi) finiscono in `wp_postmeta`.
- Impostazioni generali sono in `wp_options`.
- Utenti in `wp_users`.

Esempio:

Se crei un nuovo prodotto WooCommerce:

- WooCommerce crea un post di tipo `product`
  - Aggiunge meta come `price`, `stock`, `sku`, `gallery`, ecc.
- 

## Parte 9 — Struttura dei contenuti

In WordPress hai:

- **Pages (pagine)** → statiche (Home, Contatti)
- **Posts (articoli)** → con categorie e tag
- **Custom Post Types** → contenuti speciali (Portfolio, Prodotti, Eventi)

Esempi di Custom Post Type:

```
add_action('init', function() { register_post_type('eventi', [ 'label' => 'Eventi', 'public' => true, 'supports' => ['title','editor','thumbnail'] ]);});
```

---

## Parte 10 — Template Hierarchy

È lo “schema” che WordPress usa per decidere quale file PHP mostrare.

Esempio:

- /page.php → tutte le pagine
- /single.php → articoli
- /single-product.php → prodotti WooCommerce
- /category.php → categorie
- /author.php → pagina autore

È la parte più importante per chi sviluppa temi.

---

## Parte 11 — Child Theme (Tema figlio)

Fondamentale per modificare un tema senza perderti tutto quando aggiorni.

Contiene almeno:

- style.css
- functions.php

Esempio style.css:

```
/* Theme Name: Mio Tema Child Template: astra */
```

---

## Parte 12 — WooCommerce (se il video lo tratta)

Aggiunge:

- prodotti
- carrello

- checkout
- pagamenti
- gestione ordini
- aree membri

Tipi di prodotto:

- Semplice
  - Variabile
  - Downloadabile
  - Esterno/Affiliazione
- 

## Parte 13 — Sicurezza

Da fare **sempre**:

- Aggiornare WordPress, temi e plugin
  - Installare plugin sicurezza (WordFence, All-In-One Security)
  - Password forti
  - Limitare tentativi login
  - Disabilitare editor file interno
  - Backup regolari
- 

## Parte 14 — Performance

Cosa rallenta un sito?

- Troppi plugin
- Hosting scarso
- Immagini non ottimizzate
- Mancanza di cache

Soluzioni:

- Cache lato server (LiteSpeed, Cloudflare)
- CDN
- Minimizzazione CSS/JS

- Lazy load immagini
  - Comprimere le immagini
- 

## Parte 15 — SEO in WordPress

Punti chiave:

- Impostare permalink SEO friendly
  - Installare plugin SEO (Yoast o RankMath)
  - Scrivere title e meta description
  - Sitemap XML automatica
  - Ottimizzare immagini
  - Velocizzare il sito (Google PageSpeed)
- 

## Parte 16 — Strumenti avanzati

Per sviluppatori:

- **WP-CLI** (comandi da terminale)
- **ACF (Advanced Custom Fields)** → campi personalizzati
- **Custom Post Types** avanzati
- **Custom taxonomy**
- **Shortcode** personalizzati
- **REST API WordPress** per integrare app esterne

Esempio shortcode:

```
function saluto_shortcode(){ return "Ciao da WordPress!"; }
add_shortcode('saluto', 'saluto_shortcode');
```

---

## Parte 17 — REST API WordPress

WordPress ha una REST API nativa.

Esempio:

```
GET https://tuosito.it/wp-json/wp/v2/posts
```

Serve per:

- app mobile
  - frontend in React/Vue
  - headless WordPress
- 

## Parte 18 — Backup, staging e deploy

Backup:

- UpdraftPlus
- Duplicator
- All-in-One Migration

Staging:

- Crei una copia separata del sito per testare modifiche.
- Molti hosting lo offrono.

Deploy:

- FTP
- Git + hosting avanzati
- Staging → produzione