



---

# *Corso Web Developer Full Stack*

---



## FRONTEND | PARTE 1 - WEB DESIGN – HTML, CSS E BOOTSTRAP

### WWW. WORLD WIDE WEB

- Come è fatto un sito Web
- Browser, e altri programmi interpreti
- IDE
- Introduzione HTML/CSS/JS (Web development trifecta)
- Accenni / storia HTML
- Struttura HTML head e body
- Esercitazione

### PREREQUISITI

- Installazione Ambiente di Sviluppo
- GIT, versionamento del codice
- GITHUB
- Git Pages

### HTML, TAG DI BASE

- Tag per il testo
- Tag per enfasi del testo
- Tag di base “secondari”
- Tag hr
- Tag per elenchi
- Commenti e indentazione del codice
- Tabelle
- Esercitazione

### TAG LINK, MAILTO, IMAGES

- Tag per link e attributi del tag
- Ancore HTML
- Link assoluti e relative
- Navigazione tramite link ipertestuali
- Tag iFrame

- Esercitazione
- MailTo Link
- Il tag per le immagini (tag img)
- Esercitazione

## ELEMENTI BLOCK E INLINE

- Tag DIV
- Tag Span
- Attributo style
- Esercitazione

## HTML5, NUOVE FUNZIONALITÀ INTRODOTTE CON LA VERSIONE 5

- Tag del Web semantico (nav, article, section, aside, header, footer)
- Nuovi attributi definiti per Form, Input e Button
- Esercitazione, utilizzo dei tag del web semantico
- Creazione di Form con HTML
- Nuovi Elementi introdotti per i Form con HTML5
- Validazione form (Built-in, e Javascript)
- Tag FieldSet
- Esercitazione
- Tag iFrame
- Tag Multimediali (audio/video)
- Datalist, e Output Element
- Tag Canvas
- Esercitazione

## ALTRI TAG – FONDAMENTI SEO

- Meta Tag e cenni di Search Engine Optimizatio (SEO)
- Viewport
- Source link Tag
- Web Design Tag
- Esercitazione

## CSS, CASCADING STYLE SHEET

- Introduzione
- CSS in linea e Incorporare file CSS
- Media e Import
- Commenti CSS
- Regola CSS
- Browser Inspector
- Esercitazione

## PROPRIETÀ CSS, FONT, COLORI E SPAZIATURE

- Proprietà del testo
- Font, Google fonts
- Colori in CSS
- Spaziature in CSS, margin e padding
- Bordi
- Box model, altezza e larghezza
- Background
- Cursori
- Esercitazione

## SELETTORI CSS

- Selettori di base
- Selettori di classe e ID
- Selettore universal
- Selettore attributo
- Pseudo classi
- Pseudo elementi
- Conflitti di styling
- Esercitazione

## CSS3, POSIZIONAMENTO

- Display
- Position
- Spacing

- Flex
- Grid
- Z-index
- Media Query
- Responsive utilities
- Esercitazione

### CSS3, FUNZIONALITÀ AVANZATE

- Transizioni
- Opacità
- Animazioni
- Gradienti
- Trasformazioni
- Colonne
- Filtri
- Unità di misura
- Esercitazione

### CSS3, PREPROCESSORI

- Cosa sono e perchè si usano i CSS pre-processor
- Preprocessori più popolari (SASS, LESS, Stylus, PostCss)
- SASS
- Esercitazione

### VALIDAZIONE E TEST DI PAGINE WEB

- Validazione e test di una pagina Web (anche per dispositivi mobile)
- Determinazione della compatibilità cross-browser
- HTML Code validation
- Universal markup creation
- Esercitazione
- RWD, Responsive Web Design

## BOOTSTRAP E I WEB TEMPLATES

- Cosa sono, e perchè sono utili
- Installare con CDN
- Installare in locale
- Esercitazione

## LAYOUT CON BOOTSTRAP

- Breakpoints
- Containers
- Grid
- Columns
- Gutters
- Css Grid

## COMPONENTI DI BOOTSTRAP

- Alert
- Accordion
- Badge, Breadcrumb
- Button
- Card
- Form
- Altri componenti
- Esercitazione

## BOOTSTRAP AVANZATO

- Utilities
- Npm
- Scss

## ESERCITAZIONE, CREARE UN SITO WEB

## FRONTEND | PARTE 2 - WEB DEVELOPMENT - JAVASCRIPT

### LE BASI

- Introduzione a Javascript
- Installazione Ambiente di Sviluppo
- Tour di Javascript
- Hello World
- Esercitazione
- Variabili, costanti, comments
- Operatori aritmetici, logici, di confronto ternario
- Control flow (costrutti di condizione ed iterazione, comandi di gestione del flow code)
- Data Types e Typeof
- Array e Object
- Function and Scope
- Funzionalità aritmetica
- Date
- Lavorare con le stringhe: trim, replace, concat, length, toUpperCase, toLowerCase
- Introduzione all'Oriented Object Program
- Handling Exception (try-catch-finally-throw)
- Debugging e Use Strict

## DOCUMENT OBJECT MODEL

- Introduzione al DOM
- Vanilla Javascript, interagire con il DOM
- Inner Html, Html attributes e elements
- Dialog box
- Form input validation, post submit
- Object BOM: Navigator-Window-History-Screen-Location

## MODERN JAVASCRIPT

- Intro Javascript Es
- Let vs Var
- Template literal
- Arrow function
- Array e Object destructuring
- Spread operator
- Array find, map, filter

- Esercitazione
- Promises
- Async Await
- Fetch API
- JSON

## REACTJS

- Evoluzione del Web: i Web components
- Panoramica dei Framework per la creazione di Web Components
- SPA Single Page Application
- Perché usare ReactJS
- Principali caratteristiche, e confronto con gli altri framework

## CONFIGURAZIONE E INSTALLAZIONE AMBIENTE DI SVILUPPO PER REACTJS

- Installazione e setup
- NPM, utilizzo e Package.Json
- Introduzione ai Transpillers (Babel)
- Introduzione a WebPack e ES6
- Organizzazione codice sorgente
- ReactJS plugin/devtools

## TYPESCRIPT

- Panoramica di TypeScript, (Type safety)
- Ts Compiler
- Built-In Types
- Any e Unknown
- Disabilitare Type checking
- Null e Undefined
- Tipi dichiarati dallo Sviluppatore
- Union Types
- Types Guards
- Type Assertions
- Type Compatibility
- String Literal Types



- Mapped Types

## REACT COMPONENTS

- Il mio primo React component Hello World
- Esercitazione
- JSX
- Component State (useState)
- React Hooks Api
- Gerarchia e applicazioni
- Props (Proprietà dei component)

## REACT APPLICATION

- React change detection management
- Virtual DOM
- Esercitazione
- Gerarchia dei component (Smart/Dumb component)
- Data flow
- Esercitazione: costruisci una applicazione React
- Function components vs Class Components
- Proprietà di un Class Component
- Error boundary
- Chiamate al server

## CICLO DI VITA DEI COMPONENTI

- Initial Render
- Changes
- Component Unmount

## GESTIONE DEGLI EVENTI

- Panoramica Eventi
- Gestione degli eventi attraverso component
- Esercitazione

## REACT STYLES

- CSS e Inline Styles

- Configurazione React per TailwindCSS
- Implementazione React TailwindCSS components
- Esercitazione

## REDUX

- Gestione dello Stato tra i Componenti
- One Store
- Provider Component
- Actions
- Reducer
- Redux
- Esercitazione

## REACT ROUTER

- Panoramica React Router
- Configurazione
- Passaggio e Ricezione Parametri
- React-cookie integration
- Esercitazione

## UNIT TESTING

- Tool per la fase di Unit Testing
- React Unit Testing
- React Component Testing
- React Router Testing
- Esercitazione

## APPROFONDIMENTI

## BACKEND OPZIONE A | PHP – MYSQL - LARAVEL

## PARTE 1 – I DATABASE RELAZIONALI - MYSQL

## COME AMMINISTRARE IL DATABASE MYSQL

- Installazione MySQL
- Accedere al database MySQL attraverso la riga di comando
- Effettuare l'avvio e l'arresto dei servizi MySQL
- Comprendere come creare un nuovo account MySQL
- Aggiungere ed eliminare i privilegi ad un utente
- Modificare le proprietà di un account per l'accesso e la gestione di MySQL
- Eliminare un account MySQL
- Gestire il backup dei dati MySQL
- Amministrare MySQL attraverso software specifici
- Esercitazione

## MYSQL ARCHITECTURE

- Usare il terminal client MySQL per interagire con il MySQL Server
- Cosa sono i SQL Modes e che impatto hanno sul comportamento di MySQL
- La Session scope

## CREAZIONE E PROGETTAZIONE DI UNO DATABASE SCHEMA IN MYSQL

- Tabelle, Tipi dei campi, Indici
- Le relazioni e i vincoli tra le tabelle

## QUERY

- La basic SELECT statement
- Limitare i risultati
- Limitare le colonne ritornate
- Ordinamento
- Raggruppamento dei risultati
- Esercitazione

## JOINS, SUBQUERIES AND UNION

- Cosa sono le JOIN
- Cosa sono le SUBQUERIES
- Cosa sono le UNION

- Esercitazione

## OTTIMIZZAZIONE DELLE QUERY

- Identificare costrutti che richiedono ottimizzazione
- Creare indici per ottimizzare le query
- Strategie di ottimizzazione per INNODB
- Ottimizzazione performance attraverso la data normalization
- Esercitazione

## MODIFICA DEI DATI

- INSERT
- REPLACE
- UPDATE
- TRUNCATE
- LOAD DATA
- DELETE
- Esercitazione

## AVANZATE

- Viste
- FUNZIONI ed ESPRESSIONI fornite da MySQL per tutti i tipi di dato
- COUNT(), MAX(), SUM(), NOW(), Like
- Commenti
- Prepared Statements
- Le Transactions e transaction isolation levels
- Esercitazione

## STORED PROGRAMS

- Triggers
- Stored Procedures
- Stored Functions
- Esercitazione

## PARTE 2 – PHP

## EVOLUZIONE DI PHP

## INSTALLAZIONE CONFIGURAZIONE AMBIENTE DI SVILUPPO

## WEB SERVER

- Comprendere la definizione dei processi che si occupano di fornire i servizi web su richiesta di un browser: I server web Apache / IIS
- Effettuare le installazioni e le configurazioni dei server web Apache / IIS
- Esercitazione

## FONDAMENTI PHP | CONCETTI GENERALI

- Riconoscere la sintassi corretta per scrivere codice PHP all'interno di una pagina HTML
- Includere porzioni di linguaggio in varie sezioni della pagina web utilizzando i relativi tag
- Riconoscere le varie tecniche di visualizzazione di una pagina dinamica attraverso l'utilizzo di un server web

## STILE DI PROGRAMMAZIONE

- Riconoscere e saper utilizzare le diverse tipologie di commenti per rendere il codice più comprensibile
- Utilizzare una struttura ad albero per una corretta leggibilità del codice

## STRUTTURE DATI

- Comprendere come identificare una porzione di memoria destinata al contenimento di dati
- Tipologie di dati per la gestione degli script PHP, come ad esempio tipi booleani, interi, virgola mobile, stringhe
- Analizzare i diversi operatori di assegnazione per la gestione dei dati
- Comprendere l'utilizzo del concatenamento di stringhe per la gestione dei testi
- Gestire le operazioni aritmetiche tra variabili utilizzando i relativi operatori Riconoscere come incrementare e decrementare una variabile
- Analizzare i diversi operatori booleani per la gestione dei dati

- Analizzare gli operatori di confronto utili per la gestione delle condizioni
- Esercitazione

#### GESTIRE L'OUTPUT DEGLI SCRIPT PHP

- Istruzione echo per restituire l'output di uno script PHP
- Comprendere l'utilizzo della funzione var\_dump per fornire il valore ed il tipo di una espressione.
- Istruzione print per restituire l'output di uno script PHP.
- Riconoscere i vantaggi e gli svantaggi dei costrutti echo e print.
- Esercitazione

#### LE ISTRUZIONI CONDIZIONALI, DI CONTROLLO E I CICLI | LA LOGICA CONDIZIONALE

- Riconoscere la logica condizionale mediante l'utilizzo dell'operatore ternario
- Comprendere come utilizzare i controlli di flusso attraverso l'istruzione IF
- Riconoscere le differenze tra le istruzioni Else ed Elseif
- Gestire i controlli multipli per il flusso dei dati attraverso l'istruzione Switch
- Esercitazione

#### LE OPERAZIONI RIPETITIVE

- Comprendere come eseguire operazioni ripetitive attraverso l'utilizzo del ciclo For
- Gestire operazioni ripetitive attraverso la logica condizionale: While
- Comprendere le differenze tra il ciclo While e Do While
- Comprendere come evitare cicli "infiniti"
- Utilizzare le istruzioni fondamentali per interrompere una routine
- Esercitazione

#### STRUTTURE DATI COMPLESSE | ARRAY

- Definizione di una struttura complessa vettoriale per la gestione dei dati
- Comprendere l'utilizzo di una struttura di dati complessa e dei relativi indici
- Modificare una struttura di dati ed eliminarne i relativi elementi
- Effettuare operazioni di interazione tra gli elementi di una o più strutture complesse di dati
- Utilizzare il costrutto FOREACH per la gestione di strutture di dati complesse
- Saper effettuare azioni su ogni elemento di un vettore tramite l'uso dei puntatori
- Effettuare ricerche all'interno di una struttura vettoriale
- Esercitazione

## GESTIONE DEL CODICE PHP | RIUTILIZZARE IL CODICE PHP

- Comprendere come ottenere il codice di programmazione da un altro file Analizzare le diverse sintassi per includere un file
- Esercitazione

## LE FUNZIONI PER LA GESTIONE DELLE VARIABILI

- Comprendere il concetto di visibilità, o scope, di una variabile Valutare il contenuto di una variabile attraverso la funzione empty()
- Comprendere come verificare se una variabile è definita mediante la funzione isset()
- Eliminare una variabile attiva utilizzando la funzione unset()
- La funzione is\_string() per verificare che una variabile sia una stringa
- La funzione is\_numeric() per verificare che una variabile sia una numerica
- Esercitazione

## LE FUNZIONI PER LA GESTIONE DELLE STRINGHE

- Gestione delle stringhe all'interno di raggruppamenti di codice Eliminare tutti gli spazi da una stringa mediante la funzione trim().
- Verificare se una stringa contiene degli spazi iniziali ed eliminarli attraverso la funzione ltrim().
- Verificare se una stringa contiene degli spazi finali ed eliminarli attraverso la funzione rtrim().
- Verificare la lunghezza di una stringa utilizzando la funzione strlen(). Gestire i caratteri maiuscoli e minuscoli di una stringa.
- Comprendere come gestire file di testo con campi delimitati. Comprendere come creare ed accedere ai file di dati
- Esercitazione

## LE FUNZIONI PER LA GESTIONE DELLE STRUTTURE COMPLESSE DI DATI

- Comprendere i concetti relativi all'ordinamento dei dati
- Ordinare gli elementi di un vettore in ordine crescente aggiornando le chiavi
- Ordinare gli elementi di un vettore in ordine decrescente aggiornando le chiavi
- Ordinare gli elementi di un vettore in ordine crescente mantenendo le chiavi di origine
- Ordinare gli elementi di un vettore in ordine decrescente mantenendo le chiavi di origine
- Esercitazione

## LE FUNZIONI PER LA GESTIONE DELLE DATE

- Comprendere il significato di timestamp
  - Comprendere come restituire il timestamp di un preciso istante Restituire la data del timestamp in un formato definito dall'utente Verificare se una determinata data risulta corretta
- ### 5.6 LE FUNZIONI PERSONALIZZATE

- Comprendere come poter personalizzare una funzione per la creazione di nuove funzionalità
- Gestione delle variabili all'internodi raggruppamenti di codice personalizzato Comprendere come inviare valori ad una funzione
- Gestire i valori restituiti da una funzione
- Esercitazione

## INVIO EMAIL IN PHP

- Comprendere come inviare email attraverso PHP
- Inserire le intestazioni in un'email per verificarne i dettagli Comprendere come inviare un'email a più destinatari
- Esercitazione

## RECUPERO E VERIFICA DEI DATI INSERITI IN UN FORM HTML | RECUPERO DATI DA UN MODULO

- Comprendere con quali metodi sono inviati i dati di un form ad uno script
- Accodamento dei dati ad una URL di una pagina attraverso l'utilizzo dell'array \$\_GET
- Invio dei dati di un form in modo invisibile all'utente utilizzando l'array \$\_POST
- Esercitazione

## CONTROLLO DEI DATI DI UN MODULO

- Comprendere come monitorare la compilazione dei campi di un form
- Verificare la lunghezza ed il tipo di un campo di un form per la registrazione di un utente
- Convalidare il campo email di un form
- Convertire caratteri particolari in entità HTML
- Esercitazione

## IL MANTENIMENTO DI INFORMAZIONI IN PHP | LE SESSIONI

- Comprendere come mantenere attivi, durante la navigazione, dati significativi per la gestione delle procedure
  - Avviare e memorizzare i dati in una sessione Gestire i dati di una sessione
  - Comprendere come è possibile eliminare una sessione
- ### 8.0



- Esercitazione

## INTERAZIONE CON IL DATABASE MYSQL | PHP E MYSQL

- Riconoscere il codice PHP per effettuare la connessione e la disconnessione ad un database MySQL
- Implementare uno script PHP per la creazione di una tabella MySQL Inserire i dati in una tabella MySQL attraverso un form
- Visualizzare i dati di una tabella MySQL e formattarli per il web
- Effettuare l'aggiornamento e l'eliminazione dei dati di una tabella MySQL attraverso uno script PHP
- Ottenere il dettaglio di una riga MySQL attraverso uno script PHP
- Comprendere come realizzare un motore di ricerca interno ad un database MySQL
- Protezione delle pagine web e gestione degli accessi: login e logout utente
- Esercitazione

## PROGRAMMAZIONE AD OGGETTI IN PHP | OGGETTI

- Programmazione ad oggetti
- Creazione di un oggetto
- Accesso a proprietà e metodi
- Dichiarazione di una Classe
- Dichiarazione Metodi, Proprietà, Costanti
- Ereditarietà
- Interfacce
- Traits
- Metodi Abstract
- Costruttori
- Distruttori
- Classi anonime
- Introspezione (Reflection)
- Esaminare una classe
- Esaminare un oggetto
- Serializzazione
- Principi SOLID
- Esercitazioni

## PARTE 3 – LARAVEL

## API LARAVEL DEVELOPER - PHP

- Version 7.1+
- Composer
- Autoloading Standards
- Esercitazione/Laravel Setup

## BASIC ARCHITECTURE

- HTTP Verbs
- Request Lifecycle

## ROUTING

- Redirects
- Route Parameters
- Named Routes
- Route Groups
- Route Model Binding
- Rate Limiting

## CONTROLLERS

- Defining Controllers
- Controller Namespacing
- Esercitazione
- Resource Controllers
- Route Caching

## MIDDLEWARE

- Defining / Registering Middleware
- Middleware Parameters
- Esercitazione

## REQUESTS

- PSR-7 Requests
- Request Path and Method
- Retrieving Input

- Uploaded Files
- Proxy Configuration

## RESPONSES

- Attaching Headers / Cookies
- Encryption
- Redirects
- Flash Data
- Response Types
- Response Macros

## COLLECTIONS

- Creating / Extending Collections
- Collection Methods
- Higher-order Messages

## HELPER METHODS

- Arrays / Objects
- Paths
- Strings
- URLs

## DATABASE

- Query Builder
- Pagination
- Migrations
- Seeding

## ELOQUENT ORM

- Conventions
- Relationships
- Eloquent Collections
- Mutators / Accessors
- API Resources
- Serialization

- Scopes

## Esercitazione

### TESTING

- Creating / Running Tests
- HTTP Tests
- Session / Authentication
- Testing File Uploads
- Available Assertions
- Browser Tests / Dusk
- Data Factories
- Fakes / Mocking

## Esercitazione

### ADVANCED ARCHITECTURE

- Service Container Binding and Resolution
- Service Providers
- Facades
- Esercitazione

### VIEWS

- Creating Views
- Passing Data to Views
- View Composer
- Esercitazione

### URL GENERATION

- Named Routes
- Controller Actions
- Default Values

## SESSIONS

- Configuration
- Storing Data

- Retrieving Data
- Deleting Data
- Flash Data
- Custom Drivers
- Esercitazione

## VALIDATION

- Form Requests
- Manually Creating Validators
- Error Messages
- Validation Rules
- Custom Rules
- Esercitazione

## LOGGING

- Configuration
- Writing to Specific Channels
- Creating Custom Channels
- Esercitazione

## FRONTEND

- Blade Templating
- Localization
- Asset Compilation
- Esercitazione

## SECURITY

- Authentication
- Authorization
- Encryption / Hashing
- CSRF Protection
- XSS Protection

## ARTISAN CONSOLE

- Generating Commands

- Command I/O
- Registering Commands
- Executing Commands
- Esercitazione

## WEBSOCKETS

- Broadcasting Events
- Receiving Events
- Broadcasting Channels
- Presence Channels
- Client Events
- Esercitazione

## CACHING

- Drivers / Configuration
- Storing Items
- Retrieving Items
- Cache Tags
- Creating Custom Drivers
- Esercitazione

## EVENTS

- Registering Events / Listeners
- Queued Listeners
- Dispatching Events
- Subscribing to Events
- Esercitazione

## FILE STORAGE

- Configuration / Drivers
- Storing / Retrieving Files
- Custom Filesystems
- Esercitazione

## MAIL

- Drivers / Configuration
- Generating Mailables
- Writing Mail
- Sending Mail
- Markdown
- Local Development
- Esercitazione

## NOTIFICATIONS

- Creating Notifications
- Sending Notifications
- Mail Notifications
- Markdown
- Database Notifications
- Broadcast Notifications
- SMS Notifications
- Slack Notifications
- Custom Channels
- Esercitazione

## PACKAGE DEVELOPMENT

- Discovery
- Service Providers
- Resources
- Commands
- Assets
- Publishing File Groups
- Esercitazione

## QUEUES

- Drivers / Configurations
- Creating / Dispatching Jobs
- Running Queue Workers
- Supervisor
- Handling Failed Jobs

- Esercitazione

## TASK SCHEDULING

- Scheduling Artisan Commands
- Scheduling Queue Jobs
- Scheduling Shell Commands
- Time Zones
- Preventing Task Overlaps
- Maintenance Mode
- Esercitazione

## PROGETTO FINALE (PROGETTO PERSONALE)

- Il progetto finale consisterà nel realizzare un'applicazione web funzionale e completa sfruttando i linguaggi appresi.

## BACKEND OPZIONE B | PYTHON – POSTGRES – DJANGO – DEPLOY – PROGETTO FINALE

## PYTHON

- Introduzione a Python
- Basic data types
- Data structure
- Control Flow
- Functions
- Exceptions
- Classes
- Decorators
- Pip (Package manager) e PyPi
- Modules & Library
- Esercitazione di fine modulo

## DATABASE POSTGRES

- Introduzione a Postgres



- Installazione di PgAdmin
- Tabelle, campi, indici
- Relazioni tra tabelle
- Query SQL
- CRUD SQL

## DJANGO FRAMEWORK

- Introduzione a Django
- Architettura MTV
- Virtual environment
- Installazione e configurazione ambiente di sviluppo
- Creazione di un progetto e apps
- Struttura e avvio di un progetto
- Interfaccia di admin
- *[Progetto di studio per comprendere i vari moduli]*
- Modelli, filtri e ORM (Object-Relation Mapping)
- Admin
- Viste e Urls
- Templates
- Requests (GET, POST)
- Autenticazione, cookie, sessioni, CSRF token
- Forms
- Esercitazione di fine modulo

## DEPLOY

- Server e dominio
- Sistema operativo Linux
- Shell, Autenticazione e comandi fondamentali
- Web server NGINX
- Gunicorn - Supervisor
- Certbot e Https

## PROGETTO FINALE (PROGETTO PERSONALE) [32 ORE]

- Ecommerce site (Amazon clone): Il progetto finale consisterà nel realizzare un eCommerce funzionale e completo.

