



PHP8 livello avanzato

01, 04, 08, 21, 25 Ottobre 2024 orario 09:00-13:00

| | |
|--------------------------------|--|
| OBIETTIVI | Approfondire la conoscenza di PHP8 Avanzato |
| PREREQUISITI | si richiede alta operatività pregressa nello sviluppo di applicativi in PHP e conoscenza della programmazione ad oggetti |
| DESTINATARI | Personale tecnico informatico |
| ENTE ORGANIZZATORE | Università di Firenze |
| ENTE EROGATORE | Università di Firenze |
| LUOGO | FAD - corso on line su piattaforma fornita Naboo Academy |
| REFERENTE ORGANIZZATIVO | Formazione |
| DOCENTI | Docenti Kinetikon |
| DATA INIZIO | |
| DATA FINE | |
| DURATA IN ORE | 20 |
| SCADENZA PRENOTAZIONI | |
| MAX PARTECIPANTI | 15 |



| | |
|------------------|--|
| PROGRAMMA | <p>1 Strumenti e librerie</p> <p>Intro corso</p> <p>Panoramica tools</p> <p>Scaffolding moderno</p> <p>Composer</p> <ul style="list-style-type: none">• motivazioni• installazione e utilizzo• composer.json e composer.lock• autoload• Debug• base con var_dump() e print_r()• xdebug• Step Debugging• Profiling <p>Date and Time</p> <p>Manipolazione stringhe</p> <p>Esercitazioni:</p> <p>installiamo e proviamo cURL e carbon</p> <p>installiamo monolog per parlare di logging</p> <p>2 OOP</p> <p>Ripasso ed implementazione dei concetti</p> <p>Ereditarietà</p> <p>Namespaces</p> <p>Interfacce</p> <p>Magic methods</p> <p>PHP sessions</p> <p>Esercitazioni:</p> <p>Definiamo modelli per webapp corsi</p> <p>Simple OOP rest api</p> |
|------------------|--|

Singleton pattern: un manager (corsi)

Observer pattern: feed notifiche corsi (observer e subject)

3 OOP

Predefined PHP Interfaces

- Iterator (e for each)
- Generator
- Serializable (serializzazione e deserializzazione)

Eccezioni e Runtime Error Handling in PHP

Classi astratte

Esercitazioni:

Error Handling con files

Abstract Factory pattern

4 PHP tipato e cenni programmazione funzionale

Traits

Introspection and Reflection in PHP

Tipi espliciti in php

Funzioni pure e principi programmazione funzionale

Closures / funzioni anonime

Function Feeding e funzioni di secondo livello

Test driven development (TDD)

Esercitazioni:

logger, facciamo due trait e li usiamo entrambi nella nostra classe

Scriviamo unit test per una funzionalità e poi la implementiamo

5 Web Services

Javascript per il DOM manipulation e single page applications

Fetch, json

REST API

Da javascript a NodeJs

Architetture moderne delle webapp

Carrellata su framework javascript

Vue (template, struttura a componenti)