**Lezione 1 – Kick-off & Setup**

**Obiettivi e contenuti**

* Cos’è Python, breve storia, vantaggi e casi d’uso
* Differenze chiave tra Python 2 e Python 3
* Installazione di Python 3.12 + pip + venv (Windows/macOS/Linux)
* Working directory e REPL
* Panoramica IDE/editor: VS Code, PyCharm, Jupyter Notebook
* Regole di indentazione e uso delle virgolette
* Tipi primitivi (int, float, str, bool) e operatori

**Laboratorio / esercizi**

* Installare Python e configurare l’IDE scelto
* Scrivere uno script “Hello World” con input utente e semplice calcolo
* Creare variabili di 4 tipi diversi e stamparle con f-string

**Lezione 2 – Strutture dati Pythonic**

**Obiettivi e contenuti**

* Container: list, tuple, dict, set, str
* Immutabilità, conversioni, unpacking e slicing avanzato
* Comprehension di liste e dizionari

**Laboratorio / esercizi**

* Operazioni CRUD su liste e dizionari (aggiungere, ordinare, filtrare)
* Convertire una lista in set e verificare l’unicità degli elementi
* Manipolare stringhe con slicing

**Lezione 3 – Controllo di flusso & Funzioni**

**Obiettivi e contenuti**

* if / elif / else, while, for con range, enumerate, zip
* Funzioni: definizione, parametri, return, docstring, type-hint
* Scope: variabili globali e locali
* lambda, map, filter, comprehension annidate

**Laboratorio / esercizi**

* Funzione max(a, b) e verifica pari/dispari
* Ciclo for che stampa solo i numeri pari di una lista
* Uso di map e filter su una lista di interi
* Funzione lambda che moltiplica un numero per 10

**Lezione 4 – Moduli & Librerie standard**

**Obiettivi e contenuti**

* Sintassi import, from … import, alias
* Moduli standard: math, random, pathlib, sys.path
* Context manager di base con with

**Laboratorio / esercizi**

* Calcolare fattoriale e funzioni trigonometriche con math
* Generare password casuali o lanciare un dado virtuale con random
* Context manager che misura il tempo di esecuzione di un blocco di codice

**Lezione 5 – NumPy & Analisi descrittiva**

**Obiettivi e contenuti**

* Creare e manipolare array NumPy
* Generazione di numeri casuali
* Statistiche descrittive: media, mediana, deviazione standard
* Cenno a SciPy e modelli predittivi (overview)

**Laboratorio / esercizi**

* Creare array NumPy e calcolarne le statistiche principali
* Analizzare una distribuzione casuale di dati
* Operazioni su array: broadcasting e slicing

**Lezione 6 – File I/O & Gestione errori**

**Obiettivi e contenuti**

* Lettura/scrittura di file testuali e binari con pathlib e with
* Gestione eccezioni: try/except/else/finally
* Logging di base con il modulo logging

**Laboratorio / esercizi**

* Convertire un file CSV in JSON
* Registrare errori su file di log
* Logger con timestamp e livelli personalizzati

**Lezione 7 – Programmazione a Oggetti**

**Obiettivi e contenuti**

* Definizione di classi, \_\_init\_\_, attributi di istanza e di classe
* Ereditarietà, composizione e uso di super()
* Metodi speciali: \_\_str\_\_, \_\_len\_\_, \_\_iter\_\_

**Laboratorio / esercizi**

* Costruire una libreria di libri OO con funzioni di ricerca
* Realizzare il gioco “Battaglia navale” in stile OOP

**Lezione 8 – Packaging, Ambiente & Automation**

**Obiettivi e contenuti**

* Ambienti virtuali avanzati, requirements.txt, pyproject.toml
* Creazione di un pacchetto Python installabile
* Costruire una CLI con Typer/Rich
* Git workflow, linting (ruff) e formatting (black)

**Laboratorio / esercizi**

* Impacchettare una utility CLI e installarla in un venv
* Scrivere uno script di backup schedulabile (cron o Task Scheduler)

**Lezione 9 – Testing, Debug & Concurrency**

**Obiettivi e contenuti**

* Testing: pytest (fixture, parametrize, coverage) vs unittest
* Debug con pdb e breakpoint(), profiling con timeit
* Concorrenza: threading vs multiprocessing, panoramica GIL
* Programmazione asincrona con asyncio, async/await, gather

**Laboratorio / esercizi**

* Portare la copertura test > 90 % sulle funzioni delle lezioni 2-3
* Downloader di pagine web multi-thread
* Mini-chat TCP asincrona con asyncio

**Lezione 10 – pandas + matplotlib & Progetto finale**

**Obiettivi e contenuti**

* Manipolazione dati con DataFrame: join, groupby, aggregazioni
* Plot rapidi con matplotlib
* Refactor, best-practices e CI con GitHub Actions

**Laboratorio / esercizi**

* Analisi vendite e grafico trend con pandas + matplotlib
* Consegna progetto completo con README, test e CI; peer review in aula