

# **Corso di Informatica Base**

# Slide 1: Benvenuti all'Introduzione al Coding!



# **Benvenuti**

Sono entusiasta di accompagnarvi in questo viaggio nel mondo della programmazione.

# Chi sono

- **Nome dell'insegnante:** Mauro Bogliaccino
- **Esperienza:** Oltre 40 anni di pratica
- **Passione:** Rendere la programmazione accessibile a tutti

# Cosa faremo oggi

- Scoprire cos'è il coding
- Esplorare i linguaggi di programmazione
- Comprendere i concetti fondamentali
- Guardare al futuro del vostro percorso tech



# Obiettivo del corso

## **Fornirvi le basi per iniziare il vostro percorso nel mondo della programmazione**

“ *“Tutti dovrebbero imparare a programmare un computer, perché ti insegna a pensare”* - Steve Jobs ”

# Slide 2: Cos'è il Coding? 🤔

# Definizione

**Coding (o programmazione)** è il processo di:

- Scrivere istruzioni che un computer può comprendere ed eseguire
- Creare software, applicazioni, siti web e molto altro
- Risolvere problemi attraverso la logica computazionale

# **Perché è importante?**



## Nel mondo moderno

- **Tutto è digitale:** smartphone, auto, elettrodomestici intelligenti
- **Automazione:** dal banking online allo shopping e-commerce
- **Comunicazione:** social media, messaggistica, videochiamate



## Nel settore tecnologico

- **Richiesta crescente:** milioni di posti di lavoro tech disponibili
- **Stipendi competitivi:** tra i settori meglio retribuiti
- **Innovazione continua:** AI, blockchain, IoT, metaverso

## Il coding in numeri

-  5+ miliardi di utenti internet nel mondo
-  Settore IT da oltre \$5 trilioni globalmente
-  Crescita prevista del 22% nei prossimi 10 anni

# Applicazioni pratiche

- **App mobile** (Instagram, TikTok, WhatsApp)
- **Videogiochi** (Fortnite, Minecraft, FIFA)
- **Intelligenza Artificiale** (ChatGPT, assistenti vocali)
- **E-commerce** (Amazon, eBay, Alibaba)

# Slide 3: Linguaggi di Programmazione

# Cos'è un linguaggio di programmazione?

Un **linguaggio formale** che permette di comunicare con il computer attraverso istruzioni specifiche.

# I linguaggi più popolari



## Caratteristiche:

- Sintassi semplice e leggibile
- Versatile e potente
- Grande comunità di supporto

## **Utilizzi principali:**

- Data Science e Machine Learning
- Automazione e scripting
- Sviluppo web (Django, Flask)
- Analisi dati e visualizzazione

## Esempio:

```
print("Ciao mondo!")
nome = "Marco"
età = 25
print(f"{nome} ha {età} anni")
```

# ♥ JavaScript

## **Caratteristiche:**

- Linguaggio del web
- Eseguito nel browser
- Front-end e back-end (Node.js)

## **Utilizzi principali:**

- Sviluppo web interattivo
- Applicazioni web dinamiche
- App mobile (React Native)
- Server-side (Node.js)

## Esempio:

```
console.log("Ciao mondo!");
let nome = "Anna";
let età = 30;
console.log(`${nome} ha ${età} anni`);
```



## **Caratteristiche:**

- "Write once, run anywhere"
- Orientato agli oggetti
- Molto robusto e sicuro

## **Utilizzi principali:**

- Applicazioni enterprise
- App Android
- Sistemi bancari e finanziari
- Server di grandi dimensioni

## Esempio:

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Ciao mondo!");  
        String nome = "Luigi";  
        int età = 28;  
        System.out.println(nome + " ha " + età + " anni");  
    }  
}
```



## **Caratteristiche:**

- Alta performance
- Controllo a basso livello
- Molto potente ma complesso

## **Utilizzi principali:**

- Videogiochi (Unreal Engine)
- Software di sistema
- Applicazioni real-time
- Grafica 3D e simulazioni

## Esempio:

```
#include <iostream>
using namespace std;

int main() {
    cout << "Ciao mondo!" << endl;
    string nome = "Sara";
    int età = 27;
    cout << nome << " ha " << età << " anni" << endl;
    return 0;
}
```

# Altri linguaggi popolari

Linguaggio	Difficoltà	Utilizzo principale
<b>C#</b>	Media	Sviluppo Windows, Unity (games)
<b>PHP</b>	Facile	Sviluppo web server-side
<b>Swift</b>	Media	App iOS e macOS
<b>Ruby</b>	Facile	Web development (Ruby on Rails)
<b>Go</b>	Media	Cloud computing, microservizi
<b>Rust</b>	Difficile	Sicurezza, performance, sistemi

# **Come scegliere?**

## **Considera:**

- 1. I tuoi obiettivi:** web, app mobile, data science?
- 2. Curva di apprendimento:** quanto è facile iniziare?
- 3. Opportunità lavorative:** domanda di mercato?
- 4. Comunità e risorse:** supporto disponibile?



**Consiglio per principianti: Inizia con Python o JavaScript!**

# La logica di programmazione

La **logica di programmazione** è l'insieme dei **principi, regole e metodi di ragionamento** che permettono di risolvere un problema in modo sistematico attraverso un programma informatico.

In altre parole, è la **capacità di tradurre un problema reale in una sequenza logica di istruzioni** che un computer possa eseguire.



# Definizione

La logica di programmazione consiste nel:

- **analizzare un problema,**
- **identificare le informazioni e le operazioni necessarie,**
- **stabilire un ordine logico** tra i passi da compiere,
- e **rappresentare la soluzione** con strumenti formali (pseudocodice, diagrammi di flusso o codice).

È la base del **pensiero algoritmico**, cioè la capacità di costruire un algoritmo chiaro, corretto e ottimale.



# Concetti fondamentali

1. **Sequenza** – esecuzione ordinata delle istruzioni.
2. **Selezione** – scelte condizionali (if/else, switch).
3. **Iterazione** – ripetizione controllata di azioni (cicli for, while).
4. **Astrazione** – scomposizione del problema in sotto-problemi o funzioni.
5. **Strutture dati** – organizzazione delle informazioni da elaborare.

Questi elementi sono indipendenti dal linguaggio di programmazione scelto: sono universali.



# Importanza nella scrittura del codice

Una buona logica di programmazione consente di:

- **Scrivere codice chiaro, efficiente e privo di errori.**
- **Affrontare problemi complessi in modo strutturato,** riducendo il rischio di bug.
- **Riutilizzare e adattare le soluzioni** in contesti diversi.
- **Collaborare meglio in team**, grazie a codice leggibile e prevedibile.
- **Apprendere nuovi linguaggi** più facilmente: cambia la sintassi, ma non la logica.



## In sintesi

**“ La logica di programmazione è il pensiero razionale che sta dietro al codice.**

È ciò che distingue chi “scrive comandi” da chi “progetta soluzioni”. „