

1. PIANO E CONTENUTO DEL MODULO

Java / Javascript



Versione 1.0

1. PIANO DEL MODULO

Piano del Modulo	
Titolo del modulo	Java e Javascript
Durata	3 ore (in totale)
Obiettivi di apprendimento	Al termine di questo modulo, i partecipanti potranno di: • Comprendere lo scopo e l'uso di Java e JavaScript. • Riconosci le differenze e le somiglianze tra le due lingue. • Scrivi ed esegui programmi semplici in Java (basati su console) e JavaScript (basati su browser). • Applica la logica di programmazione di base, le variabili, l'input/output e le condizioni.
Piano di lezione	Fondamenti del Design e dello Sviluppo Digitale Modulo di Formazione per Giovani Sordi Durata: ~180 minuti (3 argomenti × 50 minuti + 2 pause × 10 minuti) 1. Cosa sono Java e JavaScript? (50 minuti) Obiettivo: Presentate entrambe le lingue e spiegate dove e come vengono usate. Punti principali da trattare:

Piano del Modulo	
	• Cos'è Java? Cos'è JavaScript? • Dove vengono usati? (app, web, giochi, ecc.) • Principali differenze (compilato vs interpretato, server vs browser) • Ambienti web: Replit per Java, JSFiddle per JavaScript • Esempi di applicazioni fatte con ciascun linguaggio. Attività: "Indovina il codice" • Ai partecipanti vengono mostrati 5 brevi frammenti di codice e viene chiesto di identificare se sono Java o JavaScript e spiegare il motivo. Metodologia dell'insegnamento: • Presentazione interattiva con esempi di codice visivo. • Discussione in piccoli gruppi. • Foglio di lavoro per completare (abbinare lo scopo alla lingua). 2. Programmiamo in Java (50 minuti) Obiettivo: Scrivi ed esegui semplici programmi Java usando compilatori online senza bisogno di installazione. Punti principali da trattare: • Struttura di un programma Java (metodo class, main()). • Stampa con System.out.println().

Piano del Modulo	
	• Variabili e tipi di dati (int, stringa). • Input tramite Scanner. • Logica semplice (if, istruzioni, cicli). Attività: • Apri un link per uno strumento online preimpostato con il codice di partenza • Modificalo per mostrare un saluto basato su un nome o un'età • Bonus: Aggiungi un calcolo (ad esempio, età in 10 anni) Metodologia dell'insegnamento: • Editor Java Online (ad esempio jdoodle.com) • Link per l'inizio fornito dall'istruttore (con modello) • Codifica insieme all'istruttore • Supporto di gruppo per test e debug • Codice minimo, massima concentrazione logica 3. Javascript (50 minuti) Obiettivo: Scrivi programmi JavaScript basati su browser con output live usando editor online. Punti principali da trattare: • Nozioni di base di JavaScript: variabili (let), manipolazione DOM • Integrazione HTML + JS

Piano del Modulo	
	• Gestione degli eventi (onclick) Attività: • Modello javascript online per insegnanti aperti • Aggiungi un contatore a un pulsante che aumenta ogni volta che viene cliccato • Aggiungi messaggio quando raggiungi i 5 clic Metodologia dell'insegnamento: • Editor Javascript Online (ad esempio jsfiddle.net) • Template HTML + JavaScript preconfezionato • Code-along + esplorazione basata sul gioco • Feedback visivo immediato • Riflessione e condivisione di gruppo
Materiali Necessari	▪ Desktop, laptop o tablet connessi a Internet ▪ Accesso a: ○ Editor Java Online (ad esempio jdoodle.com) ○ Editor Javascript Online (ad esempio jsfiddle.net)

2. CONTENUTO DEL MODULO

Contenuto	Descrizione	Contenuto
Contenuto della lezione	Termini digitali chiave:	Termini digitali chiave: ■ Variabile (Java e JavaScript): Una posizione di archiviazione nominata per dati, come un numero o una parola. ■ Tipo di Dati (Java e JavaScript): Definisce che tipo di dati contiene una variabile (ad esempio, numero, testo, vero/falso). ■ Funzione / Metodo (Java e JavaScript): Un blocco di codice riutilizzabile che svolge un compito. ■ Class (Java): Un progetto per creare oggetti, definire proprietà e comportamenti. ■ Object (Java e JavaScript): Una raccolta di proprietà e metodi basati su una classe. ■ Loop (Java e JavaScript): Ripete il codice più volte (ad esempio, cicli for e while). ■ Condition / If-Else (Java e JavaScript): Permette al codice di prendere decisioni in base a determinati criteri. ■ Array (Java e JavaScript): Una struttura a forma di lista che contiene più valori in una singola variabile.

Contenuto	Descrizione	Contenuto
		■ Stringa (Java e JavaScript): Un tipo di dato che rappresenta il testo (ad esempio, "Hello world"). ■ Console (Java e JavaScript): Uno spazio dove appaiono messaggi di output o di debug. ■ Input (Java): Ottenere dati dall'utente utilizzando strumenti come la classe Scanner. ■ Output (Java e JavaScript): visualizzazione dei risultati usando System.out.println() in Java o console.log () in JavaScript. ■ DOM (JavaScript): Una rappresentazione strutturata di una pagina HTML che JavaScript può leggere e modificare. ■ Event Listener (JavaScript): Una funzione che viene eseguita in risposta ad azioni dell'utente come clic o pressioni di tasti. ■ On Click (JavaScript): Un evento specifico che avviene quando un utente clicca su un elemento. ■ Compiler (Java): Uno strumento che traduce il codice in bytecode prima dell'esecuzione. ■ Interprete (JavaScript): Un sistema che legge ed esegue codice riga per riga direttamente nel browser. ■ Sintassi (Java e JavaScript): Le regole e la struttura per scrivere codice valido in un linguaggio di programmazione.
Riassunto video	Link su Youtube	● Algoritmo – https://www.youtube.com/watch?v=Ld_iFKCxeS8&

Contenuto	Descrizione	Contenuto
		● Sviluppo backend – Programmazione https://www.youtube.com/watch?v=2y9eJw0ZU4s – https://www.youtube.com/watch?v=FodVPon1Nbk ● Struttura di controllo - https://www.youtube.com/watch?v=BFpcRznhvCI ● Sviluppo frontend - https://www.youtube.com/watch?v=cCNqo7lzhPQ ● Java - https://www.youtube.com/watch?v=APW6WP1Q1Tg ● Javascript - https://www.youtube.com/watch?v=mEUPxfzEgFE ● Loop - https://www.youtube.com/watch?v=9S745R-Xr3k ● Variabile - https://www.youtube.com/watch?v=Mc_Msxqdlks
Attività		1. Attività di gruppo: un compito, due lingue Obiettivo: Imparare come lo stesso compito di programmazione può essere completato sia usando Java che JavaScript, e comprendere le somiglianze e le differenze tra i due linguaggi. I partecipanti lavorano in piccoli team. Ogni team riceve un semplice compito di programmazione: "Crea un programma di saluti personalizzato che chieda un nome e ti saluti." In stile staffetta, i membri della squadra si alternano a scrivere una parte del programma: ● Innanzitutto, in Java usando jdoodle.com ● Poi, in JavaScript usando JSFiddle

Contenuto	Descrizione	Contenuto
		Seguono questi passaggi: • Un membro imposta la struttura (ad esempio, classe in Java o HTML in JS) • Un altro gestisce l'input (ad esempio, Scanner in Java o input element in JS) • Un altro gestisce l'output (ad esempio, System.out.println() o aggiornamento DOM) Ogni squadra presenta le proprie due versioni, spiega il proprio approccio e riflette su: • Quali parti erano simili? • Cosa era più difficile o più facile? • Quale ambiente ti è sembrato più intuitivo? 2. Role-Play: Collaborazione con il codice Obiettivo: Esercitarsi nel lavoro di squadra, nella logica di programmazione di base e nella comunicazione reale nei compiti di programmazione usando Java e JavaScript. Ogni squadra simula una situazione di codifica in cui i membri devono: • Pianificare insieme un piccolo programma • Dividere il compito tra ruoli Java e JavaScript • Comunicare chiaramente i loro passi tra di loro

Contenuto	Descrizione	Contenuto
		Esempi di scenari: • Costruire insieme un programma di saluto (uno in Java, uno in JavaScript) • Un membro del team scrive la logica, un altro gestisce la visualizzazione • Un team debugga il codice mentre un altro spiega la logica Ruoli: • Sviluppatore (Java) – spiega e scrive la versione Java del compito • Sviluppatore (JavaScript) – spiega e scrive la versione JavaScript • Revisore – fornisce feedback su chiarezza, struttura e risultato • Presentatore – riassume cosa ha fatto il team e perché Dopo averlo messo in atto, le squadre discutono: • La comunicazione tra sviluppatori Java e JavaScript era chiara? • Tutti hanno capito entrambe le versioni del compito? • Cosa si potrebbe migliorare per una futura collaborazione? 3. Scenari reali: Obiettivo: Applicare la logica di programmazione e le competenze di base di programmazione in Java e JavaScript a mini-progetti ispirati alla vita reale. I partecipanti ricevono compiti individuali o di coppia come:



