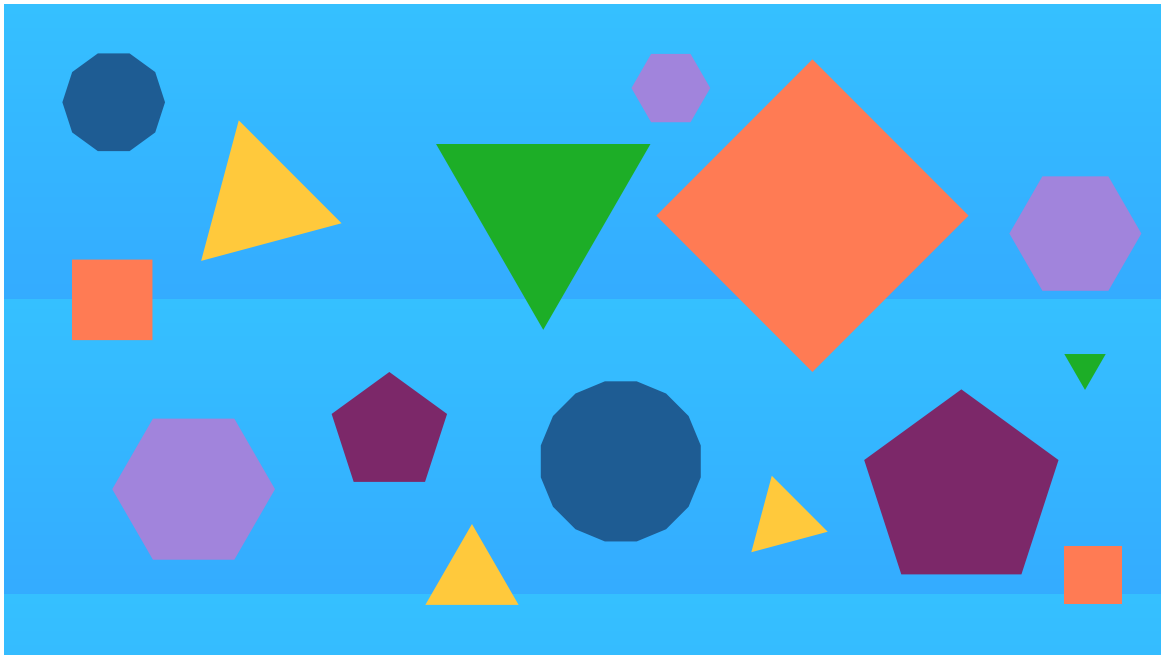




Arte Geometrica: Disegnare Poligoni

Esplora la potenza dei loop 'While' per disegnare programmaticamente forme geometriche perfette, dai triangoli agli ettagoni.

Corsi

- Classi 6-12

Materiali

- Cellulare, tablet o computer
- Connessione a Internet

Obiettivi Educativi

- Comprendere il concetto di un loop "while" nella programmazione.
- Creare un oggetto tecnologico (prototipo) utilizzando un dispositivo.
- Identificare le relazioni tra la tecnologia e il mondo circostante.
- Valutare il proprio lavoro e quello dei pari.
- Partecipare a dialoghi e riflessioni su idee di miglioramento.

Inizio (10 minuti) - Il Loop Condizionale

1. Dai il benvenuto agli studenti e presenta l'attività del giorno: **"Oggi impareremo a disegnare forme geometriche complesse usando il codice."**
2. Introduci un nuovo tipo di loop. Chiedi alla classe: **"Abbiamo visto un loop che funziona per sempre. Ma se volessimo un loop che funzioni solo *mentre* una certa condizione è vera, e poi si fermi?"**
3. Spiega che questo è ciò che fa un loop "while". Poi, collegalo a un compito reale: disegnare un poligono. Chiedi: **"Come disegneremmo un quadrato? Disegniamo un lato, giriamo di 90 gradi, disegniamo un altro lato... ripetiamo l'azione 'disegna e gira' *mentre* abbiamo ancora lati da disegnare."** Questa è l'introduzione perfetta alla logica che costruiranno.

Il Loop 'Ripeti Mentre'

Un loop **'ripeti mentre'** è come una guardia persistente. Prima controlla una condizione. **Mentre** quella condizione è vera, esegue un blocco di codice più e

più volte. Nel momento in cui la condizione diventa falsa, il loop si ferma. Questo è perfetto per situazioni in cui non si conosce il numero esatto di ripetizioni in anticipo, ma si conosce l'obiettivo che si sta cercando di raggiungere.

Cos'è un poligono regolare?

Un poligono regolare è una forma in cui tutti i lati hanno la stessa lunghezza e tutti gli angoli interni sono uguali. Conosci quelli semplici: un triangolo equilatero (3 lati), un quadrato (4 lati), un pentagono (5 lati). Ma che dire di un decagono (10 lati) o di un icosagono (20 lati)? Disegnarli a mano sarebbe molto noioso!

Il Segreto della Geometria: 360 Gradi

Ecco un trucco geometrico interessante: la somma degli angoli esterni di *qualsiasi* poligono regolare è sempre **360°**. Quindi, per capire di quanto la nostra "penna" deve girare ad ogni angolo, basta dividere 360 per il numero di lati! - **Triangolo**: $360 / 3 = 120^\circ$ di rotazione - **Quadrato**: $360 / 4 = 90^\circ$ di rotazione - **Pentagono**: $360 / 5 = 72^\circ$ di rotazione

Combiniamo loop e geometria!

Nota lo schema che si ripete per disegnare qualsiasi poligono: **disegna una linea**, poi **ruota**. Ripetiamo questo processo finché non abbiamo disegnato tutti i lati. Questo è un lavoro perfetto per un loop `ripeti mentre`! Possiamo dire al programma: **MENTRE** il numero di lati che abbiamo disegnato è inferiore al numero totale di lati che vogliamo, **FAI** l'azione "disegna e ruota".

Sviluppo (20-30 minuti) - Il Generatore di Poligoni

1. Ora che gli studenti comprendono la logica dell'uso di un loop 'while' e la semplice matematica dietro i poligoni, è tempo di costruire il loro generatore di forme.
2. Guidali attraverso **le istruzioni per creare il programma che disegna poligoni in base all'input della manopola**, come dettagliato nella sezione pratica qui sotto. Presta molta attenzione alla condizione del loop

`while` (ad es. `while contatore < lati`) e a come viene calcolato l'angolo di rotazione.

Chiusura (5-10 minuti) - Codifica Creativa

1. Una volta che tutti hanno un generatore di poligoni interattivo, è tempo di sperimentare e giocare il ruolo di un programmatore creativo.
2. Usa la sezione finale per sfidarli a modificare le variabili del programma. Questo li incoraggia a vedere come piccoli cambiamenti nel codice possono portare a grandi cambiamenti nell'output visivo, che è un concetto fondamentale nell'arte generativa.

Sfida: Modifiche Creative al Codice

Ora che hai implementato il generatore di poligoni, giochiamo con il codice!

- **Mod #1: Controllo del Perimetro.** Come modificheresti il codice per rendere il poligono più grande o più piccolo?
- **Mod #2: Direzione di Rotazione.** Cosa succede se ruoti a destra invece che a sinistra?
- **Mod #3: Colore Dinamico.** Come possiamo cambiare il colore delle linee in base al numero di lati del poligono?