

Corso Formazione
Coding e Pensiero Computazionale
CRIAD @ Casa Bufalini

[modulo-04]

APPROFONDIMENTI

OBIETTIVI DEL MODULO

- Fornire un approfondimento su un insieme di funzionalità/aspetti scelti
 - ▶ Generazione valori casuali
 - ▶ Creazione dinamica di attori (cloning)
 - ▶ Strutture dati: le liste
 - ▶ Moduli e librerie
 - ▶ Funzioni
 - ▶ Ricorsione

CASUALITA' E VARIABILITA'

- In numerosi contesti è utile disporre della possibilità di generare valori (numeri ma non solo) in modo casuale, ovvero non determinato a priori né dal programma/copione, né dall'input dell'utente
 - esempi: simulazione lancio di una moneta o dado, giochi con movimenti casuali degli elementi,...
- In Snap!: operatore “numero a caso”
 - [\[Esempio Snap!\] Micromondo lancio di un dado](#)

numero a caso tra 1 e 10

CREAZIONE DINAMICA DI ATTORI: CLONING

- ...Possibilità di creare un nuovo attore/script dal programma di un altro attore
 - azione crea/elimina un clone
 - evento “quando sono clonato”
- [Esempio Snap!] [micromondo palline](#)

crea un clone di ▼









elimina questo clone

quando vengo clonato

STRUTTURE DATI

- "Strutture dati"
 - fino ad ora abbiamo trattato come dati, informazioni, valori numerici oppure alfanumerici (stringhe), oppure logici/"booleani" (vero o falso)
 - queste sono chiamate strutture dati "concrete" o "primitive"
 - in generale possiamo pensare di poter definire e fruire di strutture anche più complesse, che si compongono di strutture primitive
 - "strutture dati astratte"
- Esempi
 - liste, tabelle, matrici
 - "alberi", "grafi", "mappe"

LISTE

- Le liste sono strutture che ci permettono di memorizzare e gestire insiemi ordinati di valori (sequenze) a cui è possibile accedere tramite un indice
 - ogni valore ha una posizione, da 1 a N, dove N è la lunghezza della lista
- Snap!: blocchi definiti nella categoria "Variabili"
- Aspetti
 - Costruzione di una lista
 - in Snap! blocco lista 
 - Operatori / funzioni elementari
 - lunghezza 
 - elemento di indice i 
 - altri: contiene elemento X? 
 - Gestione liste dinamiche
 - Aggiunta/rimozione/aggiornamento elementi ad una lista 
 - Snap! - blocchi aggiungi/cancella/ inserisci/sostituisci/rin 


- Primi algoritmi
 - [\[Esempio Snap!\] Micromondo elenco di temperature da stampare e calcolo media](#)
 - [\[Esempio Snap!\] Micromondo pagine](#)

DALLE LISTE A TABELLE

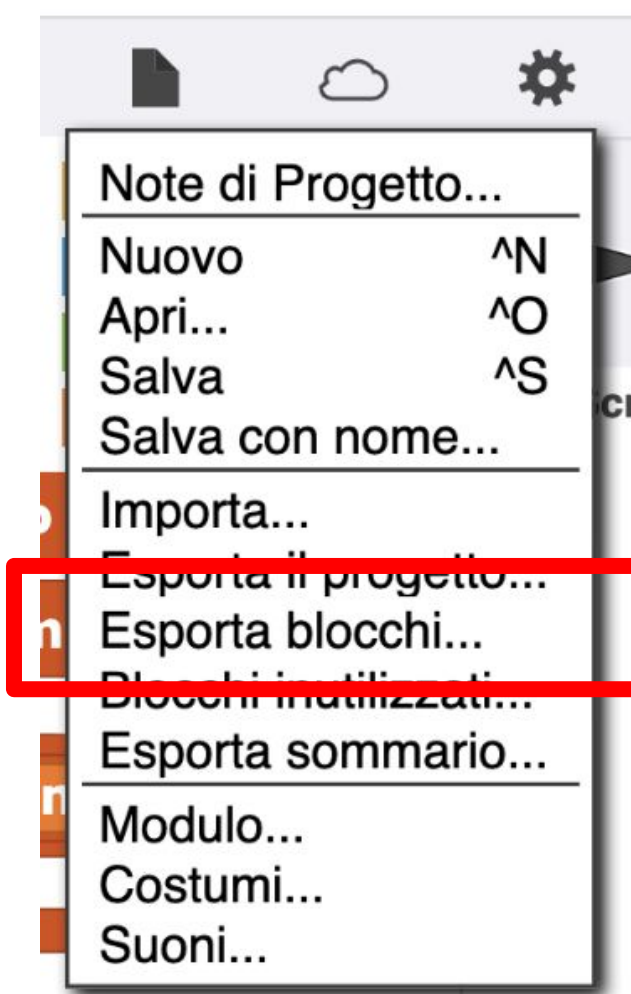
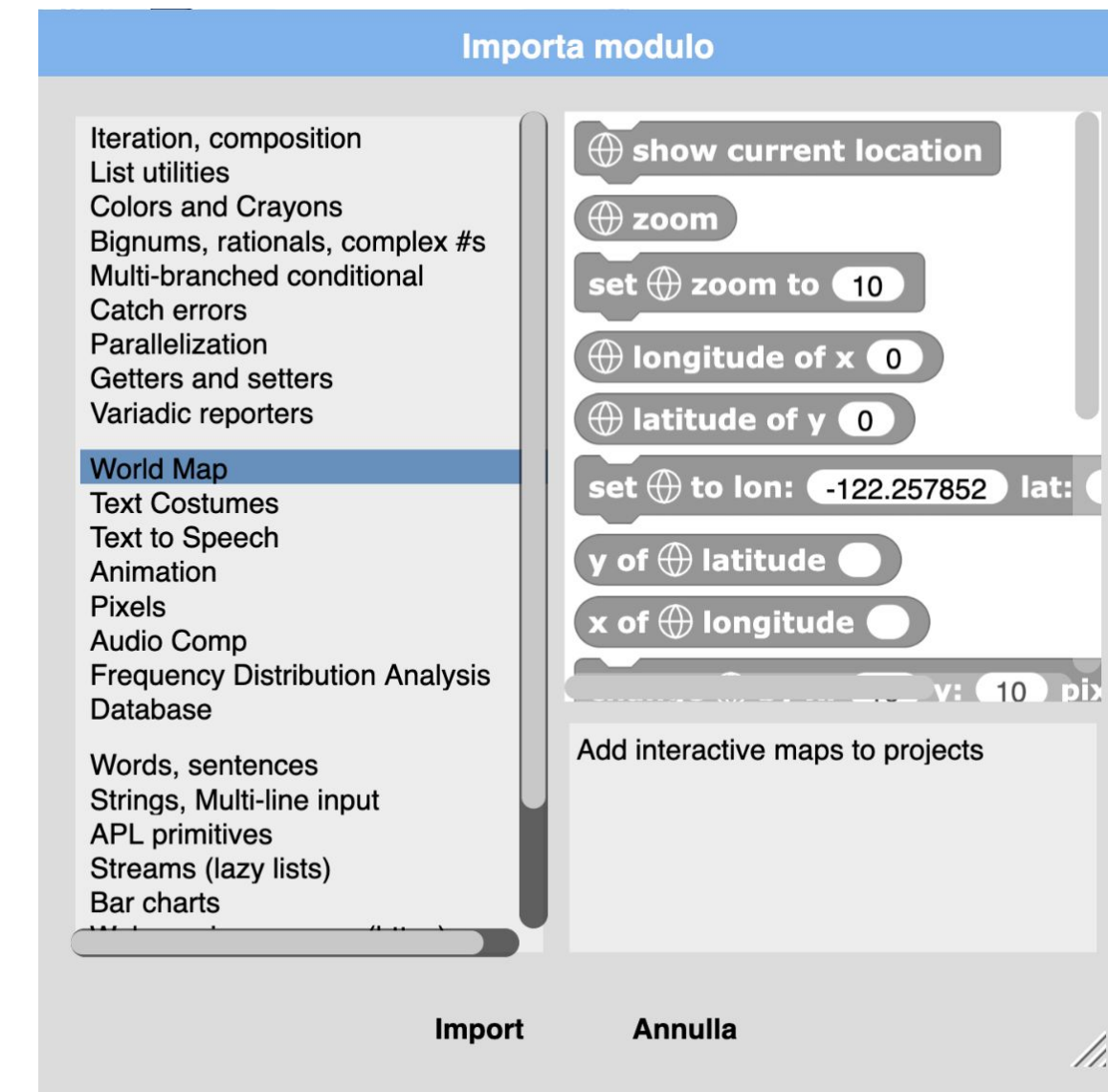
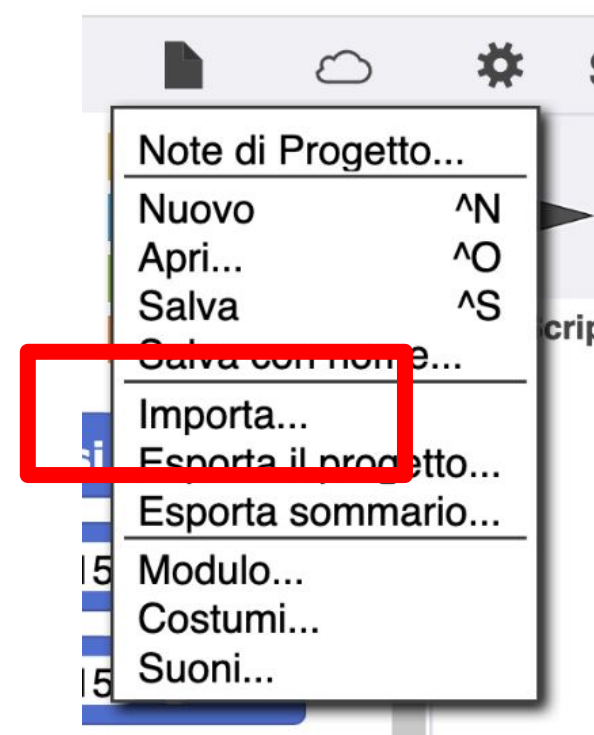
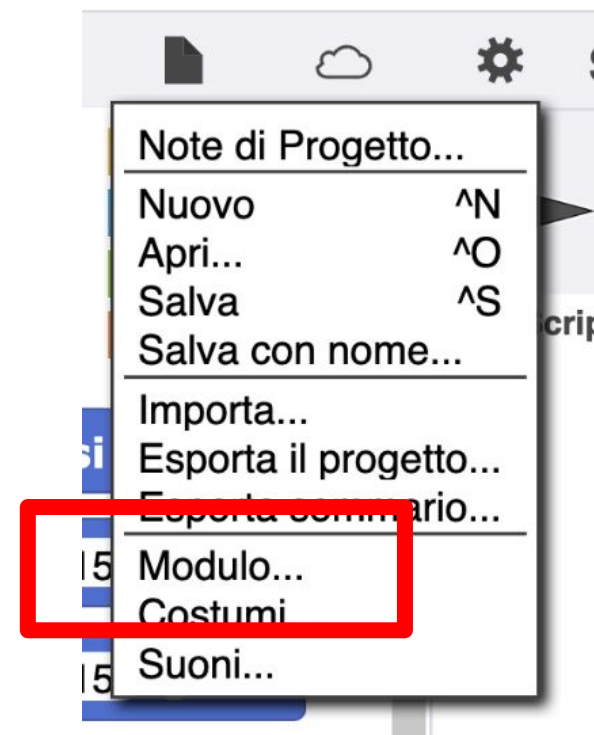
- Un'altra struttura dati molto utile nella costruzione di micromondi è data dalle *tabelle*
 - una tabella ha una o più *righe* e una o più *colonne*
 - l'accesso ad un elemento avviene specificando due indici
 - indice riga e indice colonna
- In Snap!
 - non c'è una specifica struttura dati per le tabelle, ma sono realizzabili come come *liste di liste*, ovvero liste i cui elementi sono a loro volta liste
 - [\[Esempio Snap!\] Micromondo temperature per città e temperatura più alta](#)

MODULI E LIBRERIE

- Per modulo o libreria intendiamo un insieme di elementi e funzionalità non legate o utili solo per uno specifico micromondo (o programma, applicazione), ma pensate e ideate per essere usate e *riusate* nella costruzione di micromondi diversi, a seconda delle necessità
 - ogni modulo/libreria ha tipicamente un nome, è caratterizzato da una propria unità concettuale e definisce un certo livello di astrazione
 - l'unità concettuale/livello di astrazione può concernere gli aspetti/ambiti/domini più disparati
 - sia più orientati a domini/ambiti applicativi specifici (es: modulo mappe geografiche), sia discipline(es: modulo matematico per matrici), sia funzionalità tecniche/informatiche (es: modulo database)...
- Per usare un modulo in un micromondo deve essere prima *importato*
 - importando un modulo, i suoi elementi/funzionalità vengono resi disponibili nell'ambiente

MODULI E LIBRERIE IN SNAP!

- Supporti in Snap!
 - importare moduli predefiniti
 - opzione Modulo...
 - importare moduli custom
 - opzione Importa...
 - creare un modulo
 - indirettamente, esportando blocchi



FRA I MODULI DISPONIBILI...

- Moduli con funzionalità di base
 - ▶ **"Text Costumes"**
 - avere costumi costituiti da lettere, parole
 - [\[Esempio Snap!\] Micromondo pagine esteso](#)
 - ▶ **"Colors and Crayons"**
 - gestione colori RGB e altri sistemi
 - ▶ **"Create variables"**
 - permette di gestire da programma funzionalità relative alle variabili (es: visualizzazione)
 - ▶ **"String, Multi-line inputs"**
 - funzionalità relative alla gestione di stringhe e input
 - ▶ **"Text to Speech"**
 - permette di riprodurre in formato vocale il contenuto di testi, parole, numeri..
 - ▶ **"Audio Comp"**
 - funzionalità relative a produzione e campionamento di suoni

FRA I MODULI DISPONIBILI...

- Moduli con funzionalità orientate a discipline/domini
 - ▶ **"World Map"**
 - per poter aggiungere e gestire mappe geografiche
 - [\[Esempio Snap!\] Micromondo mappa Cesena](#)
 - ▶ **"Bar Charts"**
 - per poter creare grafici che rappresentano dati, memorizzati in tabelle
 - [\[Esempio Snap!\] Micromondo lancio dadi con grafico](#)
 - ▶ **"Words, sentences"**
 - per elaborare parole e testi
 - ▶ **"Frequency Distribution Analysis"**
 - per analizzare e visualizzare distribuzioni statistiche
 - ▶ **"Database"**
 - funzionalità di archiviazione, per memorizzare in modo persistente dati

FRA I MODULI DISPONIBILI...

- Moduli per interfacciarsi con dispositivi e mondo esterno
 - ▶ **"Hummingbird robotics"**
 - per pilotare/interagire con dispositivi usati in ambito maker e robotica educativa
 - microbit
 - ▶ **"Web services access"**
 - per accedere a pagine e servizi web disponibili in rete

ESEMPIO CREAZIONE MODULI

- Costruiamo un modulo Tabella
 - obiettivo: poter definire e usare una tabella con blocchi al livello di astrazione opportuno, senza dover conoscere dettagli in merito all'implementazione, basata su liste di liste
 - Blocchi
 - crea tabella di $\langle N \rangle$ righe e $\langle M \rangle$ colonne
 - assegna il valore: $\langle V \rangle$ all'elemento riga: $\langle R \rangle$ colonna: $\langle C \rangle$
 - accedi a elemento riga: $\langle R \rangle$ colonna: $\langle C \rangle$
 - ...
- Versione più avanzata: gestione più tabelle

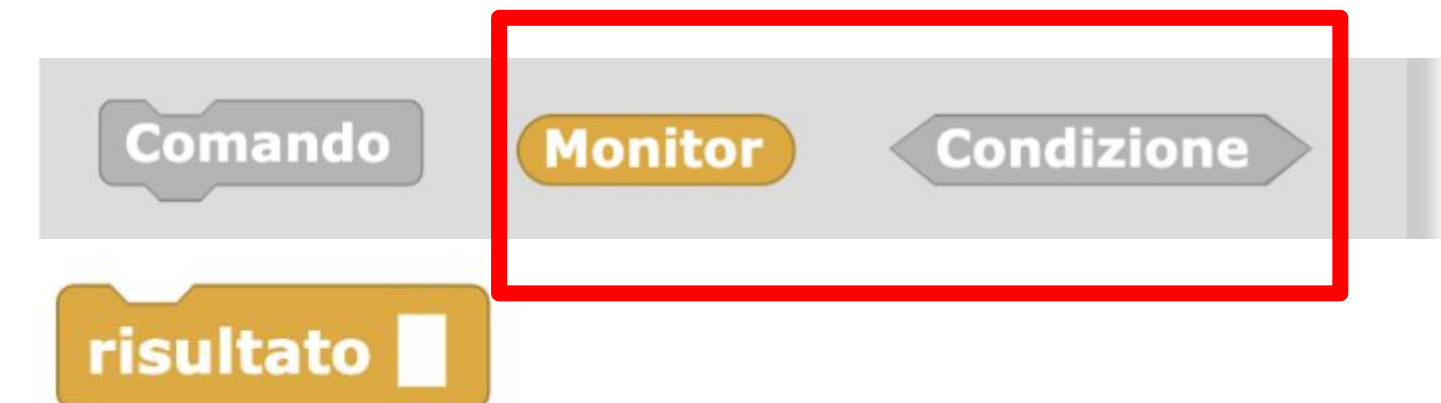
PROGETTO RIASSUNTIVO

- LANCIO DADI -

- Esempio semplice di simulazione in cui si lanciano ripetutamente 2 dati, tenendo traccia del numero di volte (occorrenze) con cui sono usciti i numeri da 1 a 12 e mano a mano visualizzando in un qualche modo i valori
- [\[Esempio Snap!\] micromondo lancio 2 dati e statistiche](#)

FUNZIONI

- In gergo informatico, per "funzione" si intende una procedura, una computazione che si presuppone produrre un risultato
 - come nel caso delle procedure, le funzioni possono avere uno o più parametri di input
 - a differenza delle procedure, devono sempre terminare "restituendo" un valore che rappresenta il risultato, il "valore di ritorno"
 - la chiamata di una funzione produce il valore che, come per le espressioni, può essere assegnato ad una variabile o usato come argomento di altri blocchi o espressioni
- In Snap!
 - creazione blocchi di tipo "Monitor" o "Condizione"
 - blocco "risultato" per specificare il valore di ritorno



RICORSIONE IN GENERALE

- ...