

SCRATCH 3.0

CAPITOLO 11 LE LISTE



Gli argomenti di questo capitolo:

Le liste

Il progetto: Un quiz con risposte nella lista

Il progetto: Partenza del quiz con pulsante

Rendere un oggetto un pulsante cliccabile

Il progetto: Quiz temporizzato,
con scelta della domande

Il progetto: Quiz con domande casuali

Il progetto: Domande e risposte nelle liste

Il progetto: I poligoni, calcolo disegno

Il progetto: Un istogramma a colonne

Il progetto: Le equazioni di primo grado

Cos'è una lista, o **Array** in termini informatici?

Potremmo semplificare le cose dicendo che la **lista è una variabile** molto particolare (ricordi sempre lo scatolone del quale abbiamo parlato, vero?); una variabile che non contiene un singolo oggetto ma può contenerne davvero tanti e tutti devono essere dello stesso tipo. Però questo particolare scatolone al suo interno contiene un foglietto che scrive il numero preciso del posto che occupa ogni elemento che inseriamo nello scatolone! Vedi che bello! Tornando alle liste, ogni singolo oggetto della lista, che si chiama **elemento**, viene individuato in modo univoco da un numero, l'**indice**) che indica esattamente la posizione di quel determinato elemento nella lista. Si può accedere agli elementi di una lista proprio attraverso il suo indice.

Analizziamo la lista mostrata qui in basso (Tabella A), come esempio. Questa lista contiene le prime cinque lettere dell'alfabeto, abbiamo quindi 5 elementi (le lettere alfabetiche) e ognuna ha il suo indice. Ogni lista ha anche una **lunghezza**, data dal numero dei suoi elementi. In questo, la lunghezza di questa lista è **5**.

TABELLA A		TABELLA B	
<i>Indice</i>	<i>Elemento</i>	<i>Indice</i>	<i>Elemento</i>
1	A	1	C
2	B	2	B
3	C	3	A
4	D	4	B
5	E	5	A
		6	A

Per utilizzare l'elemento **C** di questa lista, ci interessa il suo indice, che è **3**.

Un array come questo, lo puoi immaginare come una tabella con due colonne: la prima colonna contiene gli indici (ed è realizzata dal programma in modo automatico), la seconda invece contiene gli elementi.

Osserva la Tabella B, la sua **lunghezza** è **6**. Come vedi ci sono diversi elementi simili, ma ognuno sarà caratterizzato da un suo preciso indice. Quindi anche se ci sono elementi simili, l'indice li distinguerà chiaramente e non si potrà creare confusione.

Cerchiamo di familiarizzare un po' con le liste, poi vedremo come applicarle nelle varie situazioni.

Apriamo la categoria controllo e, come ultima voce sotto i blocchi, troviamo **Crea una lista**. Clicchiamoci sopra: si aprirà la finestra **Nuova Lista**: verrà chie-



sto il nome da dare alla lista e poi occorrerà decidere se lasciarla **pubblica**, cioè utilizzabile da tutti i nostri sprite, o **privata**, valida quindi soltanto per lo sprite che si sta usando in quel momento.

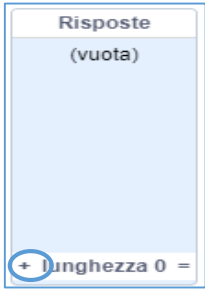
Non dimentichiamo cosa ho detto prima: le liste fanno parte delle variabili, per questo ci troviamo di fronte ad una finestra che è totalmente simile alla finestra che si apre quando decidiamo di creare una nuova variabile. E anche per questo troviamo le due voci che ci permettono di rendere la lista **pubblica** oppure **privata**, esattamente

come succedeva nella creazione delle variabili.



Creiamo allora la nostra lista, lasciamola pubblica e chiamiamola **Risposte**.

Adesso che abbiamo creato la nostra lista, troveremo il suo nome all'interno della categoria **Variabili**, sotto i precedenti blocchi che già possedeva questa categoria. Assieme al nome della nostra lista **Risposte**, troveremo anche tanti nuovi blocchi (prima non erano presenti) che permettono di lavorare con l'intera lista o con i suoi elementi.



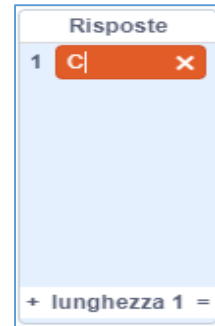
Creiamo anche una seconda lista, con le stesse operazioni precedenti e chiamiamola **domande**.

Osserva quanti nuovi blocchi, nella categoria **Variabili**, ci vengono offerti da Scratch per lavorare con le liste! Sono ben dodici nuovi blocchi, oltre al blocco che riporta il nome della lista.

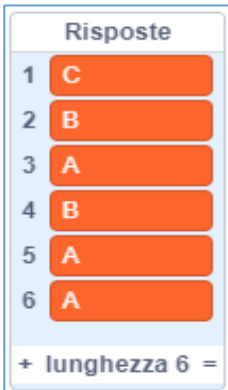
Quando creiamo una lista, sullo stage vedremo subito il riquadro della lista che, all'inizio, sarà vuota (*empty*). Questa è la nostra lista **Risposte** come si presenta appena la

creiamo. Ha zero elementi, **lunghezza** uguale a **0**.

Per riempire la lista è sufficiente cliccare sul segno “+” che vediamo qui sulla sinistra (l’ho cerchiato in rosso per evidenziarlo): comparirà un campo con lo sfondo rosso in cui inserire il primo elemento (vedi immagine a destra). Una volta inserito un elemento, subito accanto comparirà il suo indice. In questa lista come elemento c’è la lettera **C** e ha come indice **1**.



Ogni elemento potrà essere modificato oppure cancellato con un click su quella **x** che compare accanto al suo nome. Clicchiamo nuovamente sul segno “+” in basso e riempiamo tutta la lista con gli elementi indicati nella tabella precedente (C, B, A, B, A, A).



Questa è la lista ormai riempita: ogni elemento ha alla sua sinistra il suo bell’indice, che indica la posizione all’interno della lista. Abbiamo due elementi simili B. Ma il primo elemento “B” è individuato dall’indice “2”, l’altro elemento “B” ha come indice “4”: non si può sbagliare quando si vuole utilizzare un elemento, perché ognuno ha un suo indice univoco.

N.B. Tieni presente che Scratch parte dal numero **1** nell’enumerare gli indici di una lista; normalmente, invece, nella maggior parte dei linguaggi di programmazione si dà

il numero **0** come partenza, come indice del primo elemento.

Come avrai capito, questi elementi della lista **Risposte** sono le risposte corrette delle domande del quiz che abbiamo visto nel capitolo precedente. E’ vero, il precedente progetto del quiz aveva solo 3 domande, ma qui ne ho preparate altre 3, quindi questo quiz, come è adesso è già pronto per 6 domande. Ma vedrai

che utilizzando le liste, non ci saranno problemi di codice anche se le domande dovessero diventare 50!

Questa lista contiene dunque le 6 risposte corrette alle domande che sono contenute nei 6 sfondi del prossimo quiz. Adesso la risposta alla domanda, non sarà più controllata singolarmente, cioè non ci sarà un controllo per ogni singola domanda, ma un unico codice che controlla se la risposta alla domanda numero x del quiz è uguale all'elemento x della lista **Risposte**; cioè si controlla, per esempio, se la risposta alla **domanda 3** del quiz, corrisponde all'elemento 3 della lista **Risposte**. Nella lista che hai appena visto, la risposta corretta alla prima domanda del quiz è indicata dalla lettera "C": adesso, come facciamo a controllare che la risposta inserita per la prima domanda corrisponde davvero al primo elemento della lista **Risposte**?

Bene, rifacciamo tutto il progetto del quiz con 6 domande.

IL PROGETTO: UN QUIZ CON RISPOSTE NELLA LISTA. QUIZ V.2

Questo nostro progetto sarà composto da 6 domande. Per ogni domanda ci saranno 20 secondi di tempo. Per ogni domanda corretta 1 punto in più; per ogni domanda errata 1 punto in meno.

Se non si risponde ad una domanda entro i 20 secondi, il gioco finirà e comparirà il punteggio totalizzato fino a quel momento.

Diversamente soltanto alla fine del gioco vedrai il punteggio totale, scoprendo quindi a quante domande si è risposto in modo corretto (o sbagliato)!

Lo sfondo iniziale che ho scelto è *School*, lo sprite che porgerà le domande sarà *Dee*, creeremo 6 sfondi, come abbiamo già fatto in precedenza; ogni sfondo conterrà una domanda, si risponderà con la tastiera digitando una lettera fra: A, B, oppure C.

Nella pagina successiva ho riportato il codice di *Dee*: prima di tutto viene **nasco-sta** la variabile **tempo**; poi un benvenuto e quindi le spiegazioni su come si svolgerà il quiz. Nello stesso tempo un altro script azzerà le variabili, carica lo sfondo *School*. Dopo 27 secondi di attesa (se fai i calcoli nello script qui in alto, 25 secondi servono a *Dee* per dare il benvenuto e le spiegazioni, 2 secondi sono di attesa) si passa allo sfondo *domanda1* con la prima domanda. A questo punto si **invia a tutti pronti** come messaggio.

Il messaggio pronti viene ricevuto sia dallo sfondo per fa partire il conto alla rovescia, sia da Dee che darà inizio al nuovo codice, il cuore del programma. Per questo programma ci occorrono diverse variabili: **tempo** per il cronometro; **punti** per il punteggio in base alle risposte date.

Domande per tenere il conto del numero della domanda alla quale si sta rispondendo.

Domande_da_fare per conservare la risposta che si è data alla domanda posta.

Vediamo quindi i vari blocchi che esegue Dee **quando riceve pronti**.

