

translated by Google

Questa pagina è stata tradotta dall'API Cloud Translation  
([//cloud.google.com/translate/?hl=it](https://cloud.google.com/translate/?hl=it)).

[Switch to English](#)

# Crea una UI reattiva con ConstraintLayout

Parte di [Android Jetpack](https://developer.android.com/jetpack?hl=it) (<https://developer.android.com/jetpack?hl=it>).

## Prova il metodo Scrivi

Jetpack Compose è il toolkit consigliato per la UI per Android. Scopri come utilizzare i layout in Compose.

### [Layout del vincolo in Scrivi](https://developer.android.com/jetpack/compose/layout) →

(<https://developer.android.com/jetpack/compose/layout/s/constraintlayout?hl=it>)



[ConstraintLayout](https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it) (<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it>) consente di creare layout complessi e di grandi dimensioni con una gerarchia di visualizzazione semplice, senza gruppi di viste nidificati. È simile a [RelativeLayout](https://developer.android.com/reference/android/widget/RelativeLayout?hl=it)

(<https://developer.android.com/reference/android/widget/RelativeLayout?hl=it>), in quanto tutte le viste sono disposte in base alle relazioni tra le viste gemelle e il layout principale, ma è più flessibile di [RelativeLayout](https://developer.android.com/reference/android/widget/RelativeLayout?hl=it) e più facile da utilizzare con l'Editor layout di Android Studio.

Tutta la potenza di [ConstraintLayout](https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it) è disponibile direttamente negli strumenti visivi dell'editor di layout, in quanto l'API di layout e l'editor di layout sono specialmente creati l'uno per l'altro. Puoi creare il tuo layout con [ConstraintLayout](https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it) interamente trascinandolo invece di modificare il file XML.

## Building interfaces with ConstraintL...



Questa pagina mostra come creare un layout con [ConstraintLayout](https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it) in Android Studio 3.0 o versioni successive. Per saperne di più sull'editor layout, consulta [Creare una UI con l'editor layout](#)

(<https://developer.android.com/studio/write/layout-editor?hl=it>).

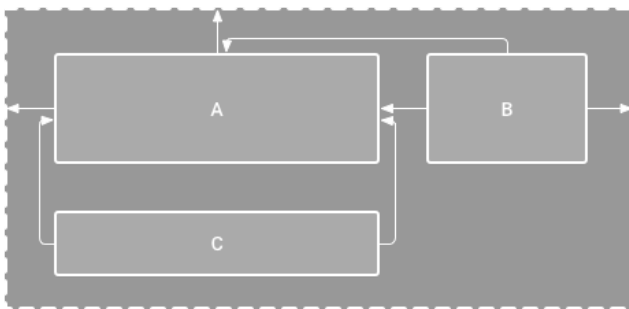
Per vedere una serie di layout che puoi creare con [ConstraintLayout](https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout?hl=it), consulta il [progetto Constraint Layout Example su GitHub](https://github.com/android/views-widgets-samples/tree/main/ConstraintLayoutExamples) (<https://github.com/android/views-widgets-samples/tree/main/ConstraintLayoutExamples>).

## Panoramica dei vincoli

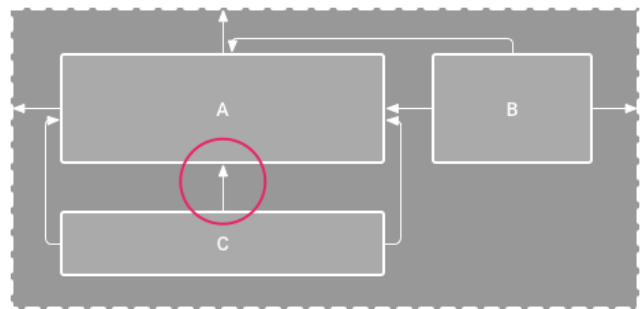
Per definire la posizione di una vista in `ConstraintLayout`, aggiungi almeno un vincolo orizzontale e uno verticale per la vista. Ogni vincolo rappresenta una connessione o un allineamento a un'altra vista, al layout principale o a una linea guida invisibile. Ogni vincolo definisce la posizione della vista lungo l'asse verticale o orizzontale. Ogni vista deve avere almeno un vincolo per ogni asse, ma spesso sono necessari più vincoli.

Quando trascini una visualizzazione nell'Editor del layout, questa rimane nel punto in cui l'avevi lasciata anche se non ha vincoli. Questo solo per semplificare l'editing. Se una vista non ha vincoli quando esegui il layout su un dispositivo, viene disegnata nella posizione [0,0] (nell'angolo in alto a sinistra).

Nella Figura 1, il layout viene visualizzato correttamente nell'editor, ma non è presente un vincolo verticale nella vista C. Quando il layout viene disegnato su un dispositivo, la vista C è allineata orizzontalmente ai bordi sinistro e destro della vista A, ma viene visualizzata nella parte superiore dello schermo perché non ha un vincolo verticale.



**Figura 1.** L'editor mostra la vista C sotto la A, ma non ha un vincolo verticale.



**Figura 2.** La vista C è ora vincolata verticalmente sotto la vista A.

ebbene un vincolo mancante non causi un errore di compilazione, l'Editor layout indica i vincoli mancanti come errore nella barra degli strumenti. Per visualizzare errori e altri avvisi, fai clic su **Mostra avvisi ed errori**.

Per aiutarti a evitare la mancanza di vincoli, l'Editor layout aggiunge automaticamente i vincoli con le funzionalità Collega automaticamente e deduci i vincoli (`#use-autoconnect-and-infer-constraints`).

## Aggiungi ConstraintLayout al tuo progetto

Per utilizzare `ConstraintLayout` nel tuo progetto, procedi come segue:

1. Assicurati di aver dichiarato il repository `maven.google.com` nel file `settings.gradle`:

```

Alla moda Kotlin (#kotlin)
(#alla-moda)

```

```

dependencyResolutionManagement {
    ...
    repositories {
        google()
    }
}

```

2. Aggiungi la libreria come dipendenza nel file `build.gradle` a livello di modulo, come mostrato nell'esempio seguente. La versione più recente potrebbe essere diversa da quella mostrata nell'esempio

**TrendyKotlin** (#kotlin)  
(#trendy)

```
dependencies {
    implementation "androidx.constraintlayout:constraintlayout:2.2.0-alpha13"
    // To use constraintlayout in compose
    implementation "androidx.constraintlayout:constraintlayout-compose:1.1.0-alpha13"
}
```

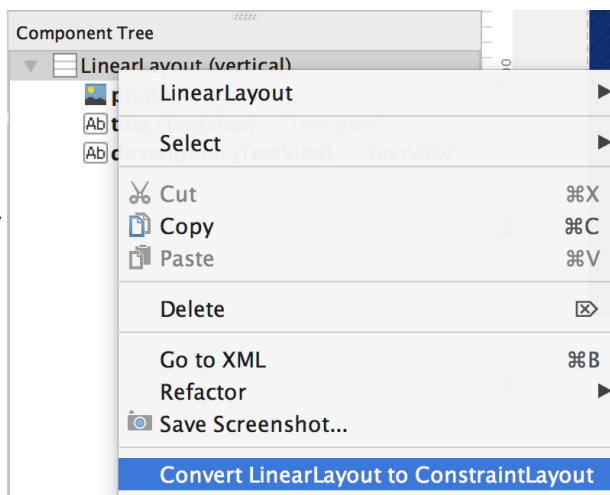
3. Nella barra degli strumenti o nella notifica di sincronizzazione, fai clic su **Sincronizza il progetto con i file Gradle**.

ora è tutto pronto per creare il tuo layout con `ConstraintLayout`.

## Convertire un layout

Per convertire un layout esistente in un layout con vincoli:

1. Apri il tuo layout in Android Studio e fai clic sulla scheda **Design** nella parte inferiore della finestra dell'editor.
2. Nella finestra **Struttura ad albero dei componenti**, fai clic con il pulsante destro del mouse sul layout e fai clic su **Converti LinearLayout in ConstraintLayout**.



**Figura 3.** Il menu per convertire un layout in `ConstraintLayout`.

## Creare un nuovo layout

Per avviare un nuovo file di layout del vincolo:

1. Nella finestra **Progetto**, fai clic sulla cartella del modulo e seleziona **File > Nuovo > XML > Layout XML**.
2. Inserisci un nome per il file di layout e inserisci `"androidx.constraintlayout.widget.ConstraintLayout"` per il tag radice.
3. Fai clic su **Fine**.

## Aggiungi o rimuovi un vincolo

Per aggiungere un vincolo:

1. Trascina una visualizzazione dalla finestra **Tavolozza** nell'editor.

Quando aggiungi una vista in un **ConstraintLayout**, questa viene visualizzata in un riquadro di delimitazione con punti di manipolazione di ridimensionamento quadrati su ogni angolo e punti di manipolazione circolari su ogni lato.


2. Fai clic sulla visualizzazione per selezionarla.

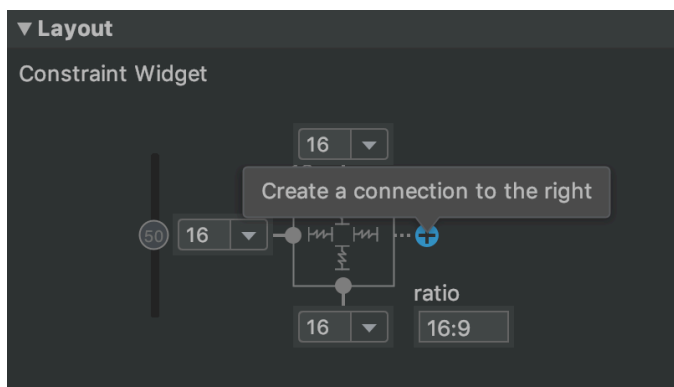
3. Esegui una delle seguenti operazioni:

- Fai clic sul punto di manipolazione di un vincolo e trascinalo in un punto di ancoraggio disponibile. Questo punto può essere il bordo di un'altra vista, il bordo del layout o una linea guida. Tieni presente che quando trascini il punto di manipolazione del vincolo, l'editor del layout mostra potenziali ancoraggi delle connessioni e overlay blu.

0:00 / 0:02

**Video 1.** Il lato sinistro di una visualizzazione è vincolato al lato sinistro dell'immagine principale.

- Fai clic su uno dei pulsanti **Crea una connessione**  nella sezione **Layout** della finestra **Attributi**, come mostrato nella figura 4.



**Figura 4.** La sezione **Layout** della finestra **Attributi** ti consente di creare connessioni.

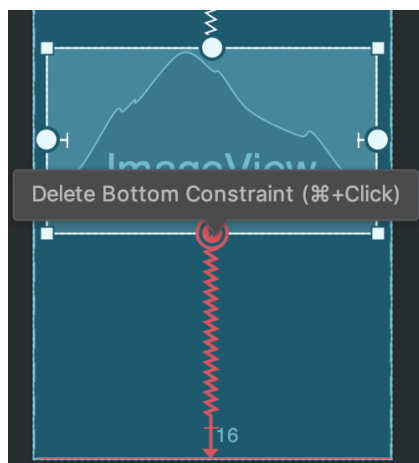
Quando viene creato il vincolo, l'editor gli assegna un margine predefinito (#adjust-the-view-margins) per separare le due viste.

Quando crei vincoli, ricorda le seguenti regole:

- Ogni visualizzazione deve avere almeno due vincoli: uno orizzontale e uno verticale.
- Puoi creare vincoli solo tra il punto di manipolazione di un vincolo e un punto di ancoraggio che condividono lo stesso piano. Un piano verticale (lato sinistro e destro) di una vista può essere vincolato solo a un altro piano verticale, mentre le basi di riferimento possono vincolarsi solo ad altre basi.
- Ogni handle può essere utilizzato per un solo vincolo, ma è possibile creare più vincoli da viste diverse allo stesso punto di ancoraggio.

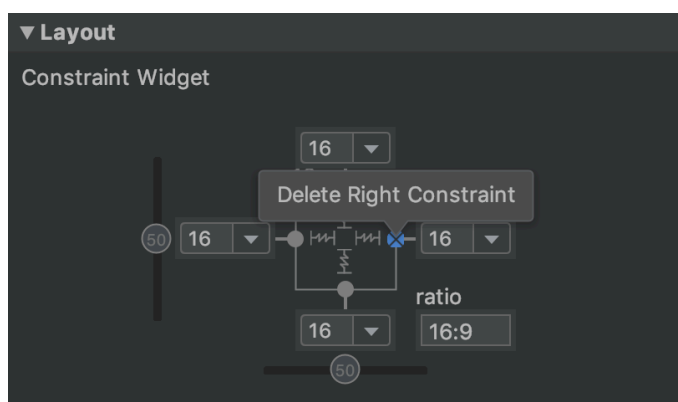
Puoi eliminare un vincolo in uno dei seguenti modi:

- Fai clic su un vincolo per selezionarlo, quindi fai clic su **Elimina**.
- Premi **Ctrl** e fai clic (Comando e fai clic su macOS) su un ancoraggio di un vincolo. Il vincolo diventa rosso per indicare che puoi fare clic per eliminarlo, come mostrato nella Figura 5.



**Figura 5.** Un vincolo rosso indica che puoi fare clic per eliminarlo.

- Nella sezione **Layout** della finestra **Attributi**, fai clic su un ancoraggio di un vincolo, come mostrato nella figura 6.



**Figura 6.** Fai clic sull'ancoraggio di un vincolo per eliminarlo.

e aggiungi vincoli opposti a una vista, le linee di vincolo avvolgono come una molla per indicare le forze opposte, come mostrato nel video 2. L'effetto è più visibile quando le dimensioni della visualizzazione sono impostate su "fisso" o su "contenuti a capo", nel qual caso la vista è centrata tra i vincoli. Se invece vuoi che la vista estenda le sue dimensioni per soddisfare i vincoli, [modifica la dimensione](#) (#adjust-the-view-size) in modo che corrisponda ai vincoli. Se vuoi mantenere la dimensione corrente, ma spostare la vista in modo che non sia centrata, [modifica la bias del vincolo](#) (#adjust-the-constraint-bias).

puoi utilizzare i vincoli per ottenere diversi tipi di comportamento del layout, come descritto nelle sezioni seguenti.

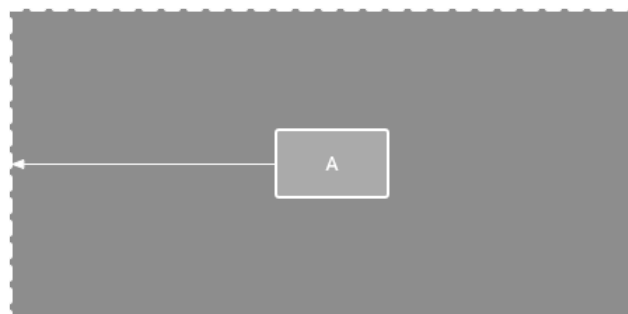
## Posizione padre

Allega il lato di una vista al bordo corrispondente del layout.

0:00 / 0:05

**Video 2.** Aggiunta di un vincolo che si oppone a uno esistente.

ella Figura 7, il lato sinistro della vista è collegato al bordo sinistro del layout principale. Puoi definire la distanza dal bordo con `marginLeft`.

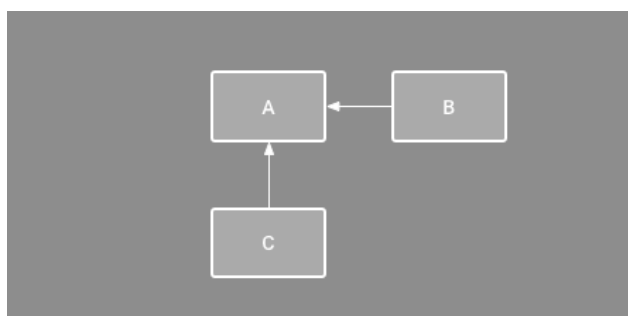


**Figura 7.** Un vincolo orizzontale all'elemento principale.

## Posizione ordine

Definisci l'ordine di visualizzazione di due visualizzazioni, verticalmente o orizzontalmente.

ella Figura 8, B è vincolato a essere sempre a destra di A, mentre C è vincolato al di sotto di A. Tuttavia, questi vincoli non implicano l'allineamento, quindi B può comunque spostarsi verso l'alto e verso il basso.



**Figura 8.** Un vincolo orizzontale e verticale.

## Allineamento

Allinea il bordo di una vista allo stesso bordo di un'altra vista.

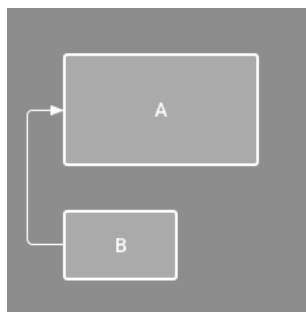
ella figura 9, il lato sinistro di B è allineato a quello sinistro di A. Se vuoi allineare i centri della vista, crea un vincolo su entrambi i lati.

Per eseguire l'offset dell'allineamento, trascina la vista verso l'interno dal vincolo. Ad esempio, la figura 10 mostra la lettera B con un allineamento di offset di 4 dp. L'offset è definito dal margine della visualizzazione vincolata.

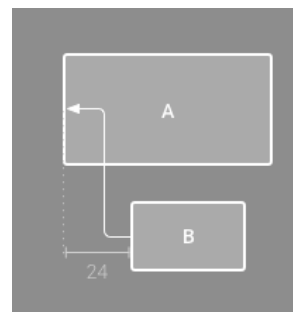
Puoi anche selezionare tutte le viste da allineare e poi fare clic su **Allinea**.



ella barra degli strumenti per selezionare il tipo di allineamento.



**Figura 9.** Un vincolo di allineamento orizzontale.



**Figura 10.** Un vincolo di allineamento orizzontale di offset.

## Allineamento di riferimento

linea la base del testo di una vista a quella di un'altra vista.

Nella figura 11, la prima riga di B è allineata con il testo in A.

Per creare un vincolo di riferimento, fai clic con il pulsante destro del mouse sulla visualizzazione di testo che vuoi vincolare, quindi fai clic su **Mostra base di riferimento**. Quindi fai clic sulla base di riferimento del testo e trascina la linea su un'altra base di riferimento.

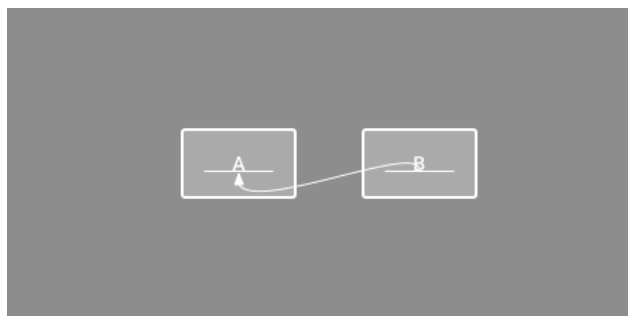


Figura 11. Un vincolo di allineamento di base.

## Ispezionare una linea guida

Puoi aggiungere una linea guida orizzontale o verticale che ti consenta di limitare le visualizzazioni e che non è visibile agli utenti della tua app. Puoi posizionare la linea guida all'interno del layout in base alle unità dp o a una percentuale relativa al bordo del layout.

Per creare una linea guida, fai clic su **Linee guida** nella barra degli strumenti, quindi su **Aggiungi linea guida verticale** o **Aggiungi linea guida orizzontale**.

Trascina la linea tratteggiata per riposizionarla e fai clic sul cerchio ai bordi della linea guida per attivare/disattivare la modalità di misurazione.

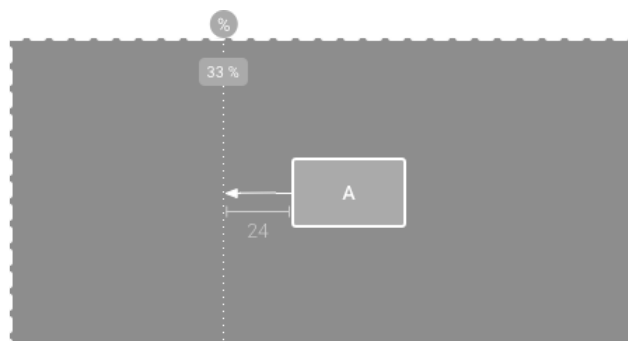


Figura 12. Una visualizzazione vincolata a una linea guida.

## Vincola a una barriera

Come una linea guida, una barriera è una linea invisibile a cui puoi vincolare le viste, ad eccezione di una barriera che non definisce la propria posizione. La posizione della barriera si sposta invece in base alla posizione delle viste al suo interno. Ciò è utile quando vuoi limitare una vista a un insieme di viste anziché a una vista specifica.

Ad esempio, nella Figura 13, la vista C è vincolata al lato destro di una barriera. La barriera è impostata sulla "fine" (o sul lato destro, con un layout da sinistra a destra) sia della vista A sia della vista B. La barriera si sposta a seconda che il lato destro della vista A o della vista B si trovi all'estrema destra.

er creare una barriera:

### 1. Fai clic su **Linee guida**



nella barra degli strumenti, quindi fai clic su **Aggiungi barriera verticale** o **Aggiungi barriera orizzontale**.

### 2. Nella finestra **Struttura dei componenti**, seleziona le viste che vuoi all'interno della barriera e trascinalle all'interno del componente.

### 3. Seleziona la barriera dalla **Struttura dei componenti**, apri la finestra **Attributi**

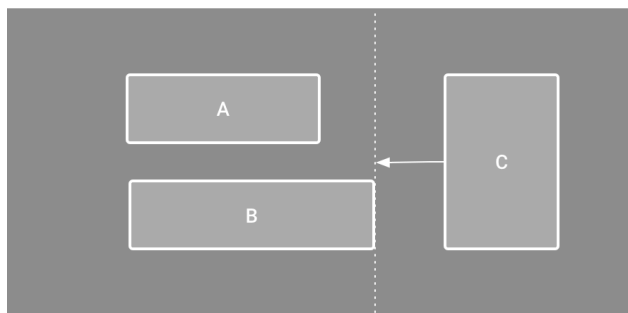
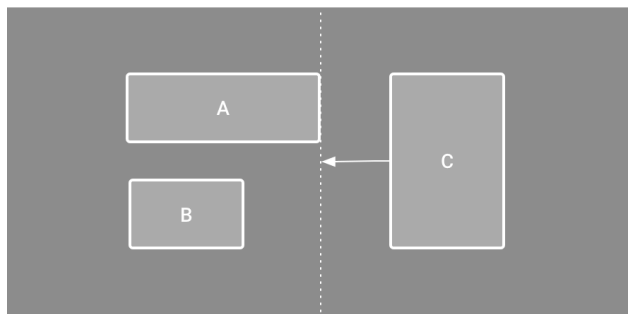


e imposta **barrierDirection**.

ra puoi creare un vincolo da un'altra vista alla barriera.

uoi anche limitare le viste che si trovano *all'interno* della barriera. In questo modo, puoi allineare tutte le visualizzazioni sulla barriera tra loro, anche se non sai quale di queste è la più lunga o la più alta.

uoi anche includere una linea guida all'interno di una barriera per garantirne una posizione "minima".



**Figura 13.** La vista C è vincolata a una barriera, che si sposta in base alla posizione e alle dimensioni sia della vista A che della vista B.

## Regola la parzialità del vincolo

Quando aggiungi un vincolo a entrambi i lati di una vista e le dimensioni della vista per la stessa dimensione sono "fisse" o "contenute a capo", per impostazione predefinita la vista viene centrata tra i due vincoli con un bias del 50%. Puoi regolare i bias trascinando il relativo cursore nella finestra **Attributi** o trascinando la visualizzazione, come mostrato nel video 3.

e invece vuoi che la visualizzazione espanda le proprie dimensioni per soddisfare i vincoli, modifica la dimensione (#adjust-the-view-size) in modo da "corrispondere ai vincoli".

0:00 / 0:11



**idea 3.** Regolazione della bias del vincolo.

## Regolare le dimensioni della visualizzazione

Puoi utilizzare i punti di manipolazione angolari per dimensionare una visualizzazione, ma queste dimensioni vengono impostate come hardcoded, ovvero la visualizzazione non viene ridimensionata a seconda dei contenuti o delle dimensioni dello schermo. Per selezionare una modalità di dimensionamento diversa, fai clic su una visualizzazione e apri la finestra **Attributi**

al lato destro dell'editor.

Nella parte superiore della finestra **Attributi** si trova la scheda di ispezione delle visualizzazioni, che include i controlli per diversi attributi di layout, come mostrato nella Figura 14. Questa opzione è disponibile solo per le visualizzazioni in un layout di vincolo.

Per modificare la modalità di calcolo dell'altezza e della larghezza, fai clic sui simboli indicati con il callout 3 nella figura 14. Questi simboli rappresentano la modalità di dimensionamento come segue. Fai clic sul simbolo per spostarti alle seguenti impostazioni:



- **Fissa:** specifica una dimensione specifica nella casella di testo seguente o ridimensionando la visualizzazione nell'editor.



- **Aggrega contenuto:** la visualizzazione si espande solo quanto necessario per adattarsi ai suoi contenuti.

- **layout\_restrictedwidth**

Imposta questo elemento su `true` per consentire alla dimensione orizzontale di cambiare e rispettare i vincoli. Per impostazione predefinita, un widget impostato su `WRAP_CONTENT` non è limitato da vincoli.



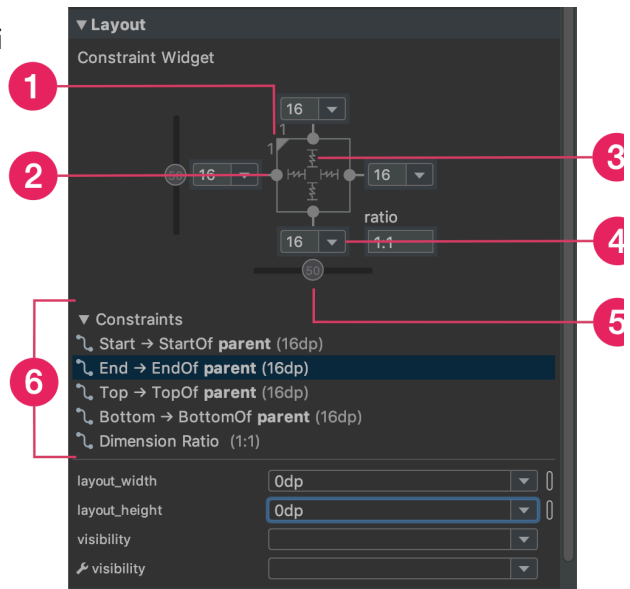
- **Vincoli di corrispondenza:** la vista si espande il più possibile per soddisfare i vincoli su ogni lato, dopo aver tenuto conto dei margini della vista. Tuttavia, puoi modificare questo comportamento con i seguenti attributi e valori. Questi attributi hanno effetto solo quando imposti la larghezza della visualizzazione su "corrisponde a vincoli":

- **layout\_constraintwidth\_min**

Viene utilizzata una dimensione `dp` per la larghezza minima della visualizzazione.

- **layout\_constraintwidth\_max**

Viene utilizzata una dimensione `dp` per la larghezza massima della visualizzazione.



**Figura 14.** Quando selezioni una vista, la finestra **Attributi** include i controlli per il rapporto delle dimensioni di 1, 2 l'eliminazione dei vincoli, 3 la modalità di altezza o larghezza, 4 margini e 5 bias dei vincoli. Puoi anche evidenziare singoli vincoli nell'editor del layout facendo clic su di essi nell'elenco dei vincoli 6.

Tuttavia, se la dimensione specificata ha un solo vincolo, la visualizzazione si espande per adattarsi ai suoi contenuti. Utilizzando questa modalità sull'altezza o sulla larghezza, puoi anche impostare un rapporto delle dimensioni (#set-size-as-a-ratio).

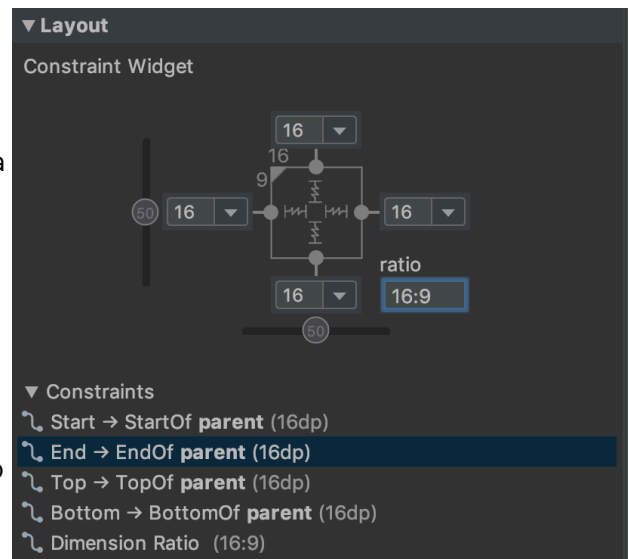
**Nota:** non puoi utilizzare `match_parent` per nessuna vista in `ConstraintLayout`. Utilizza invece i "vincoli di corrispondenza" (`0dp`).

## Imposta la dimensione come rapporto

Per impostare le dimensioni di visualizzazione su un rapporto, ad esempio 16:9, se almeno una delle dimensioni di visualizzazione è impostata su "vincoli di corrispondenza" (`0dp`). Per attivare le proporzioni, fai clic su **Attiva/disattiva vincolo proporzioni** (callout 1 nella figura 14) e inserisci il rapporto *width:height* nell'input visualizzato.

Se sia la larghezza che l'altezza sono impostate su "vincoli di corrispondenza", puoi fare clic su **Attiva/disattiva vincolo proporzioni** per selezionare la dimensione basata su una relazione dell'altra. Controlla l'indicatore che indica quale dimensione è impostata come rapporto collegando i bordi corrispondenti con una linea continua.

Ad esempio, se imposti entrambi i lati su "vincoli di corrispondenza", fai clic due volte su **Attiva/disattiva vincolo formato**: per impostare la larghezza in modo che corrisponda a un rapporto dell'altezza. L'intera dimensione è dettata dall'altezza della vista, che può essere definita in qualsiasi modo, come mostrato nella figura 15.



**Figura 15.** La visualizzazione è impostata su un aspetto 16:9 con la larghezza basata sulle proporzioni dell'altezza.

## Modificare i margini della visualizzazione

Per distribuire le visualizzazioni in modo uniforme, fai clic su **Margine**

(`8dp`).

Nella barra degli strumenti per selezionare il margine predefinito per ogni vista aggiunta al layout. Qualsiasi modifica apportata al margine predefinito si applica solo alle viste aggiunte da quel momento in poi.

Per controllare il margine per ogni vista nella finestra **Attributi** facendo clic sul numero nella riga che rappresenta ciascun vincolo. Nella figura 14, il callout 4 mostra che il margine inferiore è impostato su 16 dp.



**Figura 16.** Il pulsante **Margine** della barra degli strumenti.

Tutti i margini offerti dallo strumento sono fattori di 8 dp per consentire l'allineamento delle viste ai suggerimenti della griglia quadrata di 8 dp di Material Design.

## Controllare gruppi lineari con una catena

Una catena è un gruppo di viste collegate tra loro tramite vincoli di posizione bidirezionali. Le visualizzazioni all'interno di una catena possono essere distribuite verticalmente o orizzontalmente.

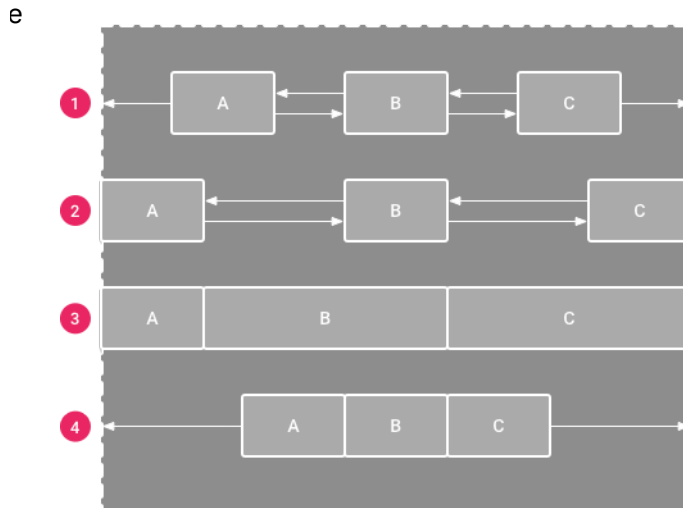


Figura 18. Esempi di ogni stile di catena.

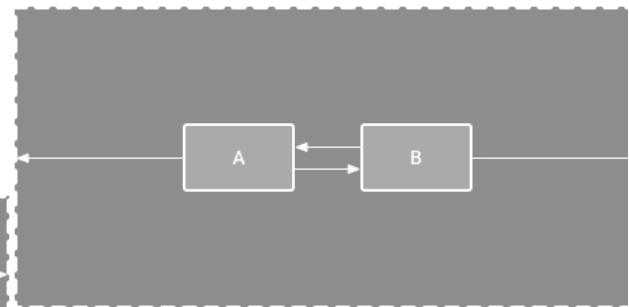


Figura 17. Una catena orizzontale con due viste.

Le catene possono essere definite in uno dei seguenti modi:

- 1 **Distribuzione:** le visualizzazioni vengono distribuite in modo uniforme dopo aver considerato i margini. Questa è l'impostazione predefinita.
- 2 **Distribuite all'interno:** la prima e l'ultima vista sono fissate ai vincoli su ciascuna estremità della catena, mentre il resto è distribuito uniformemente.
- 3 **Ponderata:** quando la catena è impostata su **spread** o **spread all'interno**, puoi riempire lo spazio rimanente impostando una o più visualizzazioni su "corrispondenza vincoli" (`0dp`). Per impostazione predefinita, lo spazio è distribuito uniformemente tra ogni vista impostata su "vincoli di corrispondenza", ma puoi assegnare una ponderazione di importanza a ciascuna vista utilizzando gli attributi `layout_constraintHorizontal_weight` e `layout_constraintVertical_weight`. Funziona allo stesso modo di `layout_weight` in un `layout lineare` (<https://developer.android.com/guide/topics/ui/layout/linear?hl=it>): la visualizzazione con il valore di ponderazione più elevato ottiene la maggiore quantità di spazio, mentre le viste con la stessa ponderazione ricevono la stessa quantità di spazio.
- 4 **Confezionato:** le viste vengono raggruppate dopo aver considerato i margini. Puoi regolare il bias dell'intera catena (a sinistra o a destra, in alto o in basso) modificando la parzialità della visualizzazione "testa" della catena.

La visualizzazione "head" della catena (la visualizzazione più a sinistra in una catena orizzontale (in un layout a sinistra a destra) e la vista più in alto in una catena verticale) definisce lo stile della catena in XML. Tuttavia puoi alternare tra **spread**, **spread all'interno** e **compresso** selezionando una visualizzazione qualsiasi nella catena e facendo clic sul pulsante della catena



sotto la visualizzazione.

Per creare una catena, segui questi passaggi, come mostrato nel video 4:

1. Seleziona tutte le viste da includere nella catena.
2. Fai clic con il tasto destro del mouse su una delle viste.
3. Seleziona **Catene**.
4. Seleziona **Centra orizzontalmente** o **Centra verticalmente**.

0:00 / 0:12

**Video 4.** Creazione di una catena orizzontale.

Ecco alcuni aspetti da considerare quando si utilizzano le catene:

- Una vista può far parte di una catena sia orizzontale che verticale, in modo da poter creare layout a griglia flessibili.
- Una catena funziona correttamente solo se ciascuna estremità è vincolata a un altro oggetto sullo stesso asse, come mostrato nella Figura 14.
- Sebbene l'orientamento di una catena sia verticale o orizzontale, l'utilizzo di una catena non allinea le visualizzazioni in quella direzione. Per ottenere la posizione corretta per ogni vista nella catena, includi altri vincoli, come i [vincoli di allineamento](#) (#alignment).

## Creazione automatica dei vincoli

Invece di aggiungere vincoli a ogni vista man mano che le posizioni nel layout, puoi spostare ciascuna vista alle posizioni che preferisci nell'Editor del layout e fare clic su **Deduci vincoli**



per creare automaticamente i vincoli.

**Deduci vincoli** analizza il layout per determinare l'insieme di vincoli più efficaci per tutte le viste. Restringe la visione alle posizioni correnti garantendo una certa flessibilità. Potresti dover apportare modifiche per fare in modo che il layout risponda in base alle esigenze di dimensioni e orientamenti dello schermo diversi.

**Connessione automatica a genitore** è una funzionalità separata che puoi attivare. Quando è abilitata e aggiungi viste secondarie a un'unità organizzativa, questa funzionalità crea automaticamente due o più vincoli per ciascuna vista man mano che le aggiungi al layout, ma solo quando è opportuno vincolare la vista al layout principale. La connessione automatica non crea vincoli ad altre viste del layout.

a connessione automatica è disattivata per impostazione predefinita. Per attivarlo, fai clic su **Attiva connessione automatica all'elemento principale**



sulla barra degli strumenti dell'Editor layout.

## Animazioni dei fotogrammi chiave

Keyframe animations with Constrai...



All'interno di un elemento `ConstraintLayout`, puoi animare le modifiche apportate alle dimensioni e alla posizione degli elementi utilizzando `ConstraintSet`

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintSet?hl=it>) e `TransitionManager` <https://developer.android.com/reference/android/transition/TransitionManager?hl=it>).

Un `ConstraintSet` è un oggetto leggero che rappresenta i vincoli, i margini e la spaziatura interna di tutti gli elementi secondari in un `ConstraintLayout`. Quando applichi un `ConstraintSet` a un elemento `ConstraintLayout` visualizzato, il layout aggiorna i vincoli di tutti gli elementi secondari.

Per creare un'animazione utilizzando `ConstraintSet`, specifica due file di layout che fungono da fotogrammi chiave di inizio e fine per l'animazione. Puoi quindi caricare un elemento `ConstraintSet` dal secondo file del fotogramma chiave e applicarlo al file `ConstraintLayout` visualizzato.

**Nota:** le animazioni `ConstraintSet` animano solo le dimensioni e la posizione degli elementi secondari. Non animano altri attributi, come il colore.

Il codice di esempio seguente mostra come animare lo spostamento di un singolo pulsante nella parte inferiore dello schermo.

```
/ MainActivity.kt
```

```
fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.keyframe_one)
    constraintLayout = findViewById(R.id.constraint_layout) // member variable

    animateToKeyframeTwo() {
        val constraintSet = ConstraintSet()
        constraintSet.load(this, R.layout.keyframe_two)
        TransitionManager.beginDelayedTransition()
```

```
constraintSet.applyTo(constraintLayout)
```

---

```
/ layout/keyframe1.xml
/ Keyframe 1 contains the starting position for all elements in the animation
/ as well as final colors and text sizes.

?xml version="1.0" encoding="utf-8"?>
androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/a
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <Button
        android:id="@+id/button2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Button"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
/androidx.constraintlayout.widget.ConstraintLayout>
```

---

```
/ layout/keyframe2.xml
/ Keyframe 2 contains another ConstraintLayout with the final positions.

?xml version="1.0" encoding="utf-8"?>
androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/a
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <Button
        android:id="@+id/button2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Button"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintBottom_toBottomOf="parent" />
/androidx.constraintlayout.widget.ConstraintLayout>
```

---

**ota:** con Android Studio 3.6 e versioni successive, la funzionalità [Visualizza associazione](#)

<https://developer.android.com/topic/libraries/view-binding?hl=it>) può sostituire le chiamate `findViewById()`. Fornisce inoltre la sicurezza del tipo in fase di compilazione per il codice che interagisce con le viste. Valuta la possibilità di utilizzare l'associazione di viste anziché `findViewById()`.

## Risorse aggiuntive

`ConstraintLayout` viene utilizzato nell'app demo [Sunflower](https://github.com/android/sunflower) (<https://github.com/android/sunflower>).

I campioni di contenuti e codice in questa pagina sono soggetti alle licenze descritte nella [Licenza per i contenuti](https://developer.android.com/license?hl=it) (<https://developer.android.com/license?hl=it>). Java e OpenJDK sono marchi o marchi registrati di Oracle e/o delle sue società consociate.

Ultimo aggiornamento 2024-06-27 UTC.