This Screentop help doc has moved to a **Notion site**.

Please visit there for the latest and greatest.

Screentop.gg WikiHelp

We ❤️ Screentop.gg, and we want to help more game designers use it.

Use the outline on the left to navigate.

This help guide is created and maintained by the community. If you want to add content, please go ahead and edit the document directly. If adding new headings, please choose **Format > Paragraph Styles** and select the appropriate heading level to apply, in order to have the headings included in the outline on the left and keep this doc tidy!

Getting Started 🚀

Video Tutorials

- Screentop 101 by Jack Rosetree - 10 videos, around 5 minutes each, covering key concepts
- Guided Prototyping on Screentop.gg by Xoe Allred - 2 hour walkthrough tutorial to create a prototype from scratch

Basic Concepts & Navigation

See the official Basic Concepts.

From Screentop.gg, access your games by selecting your avatar in the upper-right corner.

Create a new game by selecting **Create**.

When you select an existing game, you can choose:

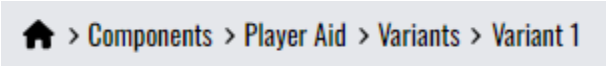
- **Edit** to define a template that sets the available components, behaviors and start state for when you choose to **Play** a game.
- **Play** to create a new instance of the game (a “Room”) that can be shared with other players. Changes made while playing the game do not affect the template at all.

In Edit mode, these are the most common items that you’ll need to update in the left page:

- **Components**: Add and define the elements of your game, including boards, cards, dice, and tokens. See Adding Components 🎲.
- **Assets**: Upload image files that can be used to set the appearance of components. See Adding Assets & Variants 🖼️.
- **Seats**: Add seats to increase the maximum number of players and/or add a seat in an Admin role.

After selecting any of the options above, select the + button to add a new item.

Use the breadcrumbs at the top of the left pane to navigate back to a previous page:



Components are the templates for the individual objects that players will manipulate in your game. Once you’ve defined a component, you can add one or more instances of that component to the game; these are called Objects in Screentop. Select the **Treasure Chest** icon in the upper-right corner to add objects to your playing surface.

My First Screentop.gg

Official Tutorial: Create a deck of cards and a hidden player hand.

See Recipes 📝 for additional walkthroughs.

Migrating from Tabletop Simulator

(Needs content)

Adding Components 🎲

Add and define templates for the elements of your game, including boards, cards, dice, and tokens. To add a component:

1. In the left pane, select **Components**.
2. Select the + button.
3. Choose the component type.
    a. Typically, you'll want to choose one of the 4 main types of components:
        i. Tiles: Double-sided. Useful for cards and tokens.
        ii. Containers: Can include Dropzones containing places where other components can snap to, stack on, and spread along. Useful for boards, decks, discard piles, player hands, and player mats.
        iii. Dice: Randomizer. Useful for dice and spinners.
        iv. Counters: Track numbers (or graphical changes) in the game. Useful as score and resource trackers, as well as game-phase indicators, reference booklets, etc.
    b. Alternatively, you can choose a pre-configured component, when you need that exact component type, such as a deck of playing cards, a set of meeples, a d6 die, or a chess board.
4. In the left pane, at the bottom of the list of Components, select the new component.
5. After configuring it, select **Save**.

Tiles

Use tiles to create the cards, tokens, and other pieces in your game that might need to be double-sided but do not need to contain other game pieces (i.e., they do not need to have their own Dropzones).

Follow the steps above under Adding Components 🎲 to create a new tile.

Here are the most common attributes to configure:

- **Name**
- **Hidden Style**: Controls what the tile looks like to players who should not be able to see it, because it's in a container with a View Policy that restricts its visibility (such as when the tile is a card in a player's hand). Options include:
    - **Opaque**: The tile appears dark gray (or another color you define).
    - **Up**: The front of the tile is shown.
    - **Down**: The back of the tile is shown. This is a typical option in many games.
    - **None**: The View Policy of the container is ignored and the tile shows normally to all players.

- **Orientations**: If this is greater than 1, players can rotate the tile using the **q** and **e** keys or by right-clicking and selecting **Rotate**. If set to 2, the tile will rotate 180°. If set to 4, the tile will rotate 90°, and so on.
- **Shape**: Define the shape of the tile. If you select **Polygon**, you can input the number of sides (for example, you can input 6 to create a hexagon). If you select **Path**, you can input SVG commands or select from a small library of predefined shapes.

- **Height** & **Width**: Define the dimensions of the tile, in pixels. For a poker-sized playing card, for example, you might set Width: 125 and Height: 175. If you are using Assets for your components, you can enter the exact height and width needed, or let Screentop scale the images to fit the dimensions you provide.

To control the look of your tiles, see Adding Assets & Variants 🖼️.

To add editable text to your tiles, see Fields.

Containers

Use containers to define areas of the table or game components that other components will be placed on top of. Common uses of containers include boards, player hands, player mats, and zones for decks or resources.

Follow the steps above under Adding Components 🎲 to create a new container.

Here are the most common attributes to configure:

- **Name**
- **Orientations**: If this is greater than 1, players can rotate the container using the **q** and **e** keys or by right-clicking and selecting **Rotate**. If set to 2, the container will rotate 180°. If set to 4, the container will rotate 90°, and so on. When a container rotates, everything inside it rotates together (even if the contained items do not have the same defined orientations).
- **Height** & **Width**: Define the dimensions of the container. For example:
    - A 20” by 30” game board could be 1000 x 1500, if you have decided that your components and assets will be based on a resolution of 200 pixels per inch.

    - A horizontal hand for 10 playing cards (sized 125 x 175) could be 1270 x 195, if you wish to have 10 pixels of “margin” around the cards.

To control which components can be placed in containers and in what locations, see Dropzones.

To hide the objects in a container from some players, see View Policies.

To control the look of your containers, see Adding Assets & Variants.

To add text to your containers, see Fields.

Dropzones

Use dropzones to define which components can be added “into” a container and in what location in the container. When a container is picked up and moved or rotated, all of the objects in the container move with it.

By default, new containers have a grid dropzone that “snaps” all objects placed into the container on a grid. Often, you’ll want to delete this default dropzone:

1. With the container open in the left pane, select **Dropzones** near the bottom.
2. Select the **3-dots** on the right of **Dropzone 1** and select **Delete**.

To add a new dropzone, select the + button and select the type of dropzone:

- **Grid**: An invisible grid of squares or rectangles that pieces will be placed onto. This is useful for boards similar to chess or Scrabble boards, as well as containers in which you don’t need variability for where objects go.
- **Anchor**: This allows for point anchors, in which objects will be placed on one or more precise points (such as a snap-point for a deck or a stack of coins), as well as line anchors, in which objects will be evenly spaced across a line (such as a hand of cards).

After creating the dropzone, select it to configure. These are the most common attributes to update:

- **Name**
- **Components**: Leave blank to accept all components onto the dropzone, or select one or more components to limit the dropzone to only interact with those components.

  NOTE: When selecting only some of the boxes, the **index** in the code is a relative reference to the items selected, not its absolute place in the list. The first item you

manually selected will be index #1, regardless of where it falls in the list. *[not sure where this paragraph was meant to be, but it doesn't make sense right here]*

- **Grid-only attributes:**
    - **Grid**: The default is **Rectangular** which allows you to set the **Width** and **Height** of the grid. You may choose **Pixel** to allow complete freedom in where objects are placed in the container.
- **Anchor-only attributes**: After configuring the main attributes of the dropzone, select **Anchors** to add one or more anchors by selecting the + button. Choose either **Point** or **Line**, and then select the anchor to configure it. These are the most common attributes to configure:
    - **Name**
    - **Point-only attributes:**
        - **X** & **Y**: The position of the point relative to the container. The position 0,0 is the center of the container; positive X values are to the right; positive Y values are *downward*. Hover your mouse over the container shown in the middle pane to see the X/Y coordinates.
    - **Line-only attributes:**
        - **X0, Y0**, **X1** & **Y1:** The start and end of the line relative to the container. The position 0,0 is the center of the container; positive X values are to the right; positive Y values are *downward*. Hover your mouse over the container shown in the middle pane to see the X/Y coordinates.
        - **Gap**: Maximum spacing between each object when placed on the line. For example, in a hand of cards, you can set the maximum spacing to be the width of the card component, or make it smaller if you want the cards to overlap each other. Note that Screentop will squeeze objects closer together when there are too many of them to use the full gap size.
        - **Alignment**: When the number of objects does not fill up the line, this determines whether they shift toward the line's Start (X0,Y0), End (X1,Y1), or Center.
    - **Capacity**: Define the number of objects that can be placed onto this specific anchor. Objects in excess of this value will spill over to other anchors, if available, or will simply not "snap" to this dropzone. Leave it blank to allow unlimited objects.
    - **Auto Rotate**: Leave it blank to not alter an object's rotation when it is placed onto this anchor. Set to 0 to automatically rotate the object to align with this container.

- **Set Tile**: Select Up or Down if you would like to automatically flip a tile to its front or back side.

Containers may have multiple dropzones, and dropzones may have multiple anchors.

View Policies

By default, all players can see all objects. To hide objects from some players, such as cards in players' hands, you'll need to configure the View Policy of each of those containers. This must be done not to the component template that defines the container, but rather to that component's individual *objects* in the game.

1. Right-click the container object showing on the surface and select **Seats**.
2. Select the player seat that is allowed to see the objects in that container. If you've added an Admin seat, also select that seat to also allow the Admin to see the objects.
3. Select **View Policy** in the right-click menu and select **Allow Seats**.

Dice

Use the die component to generate a random output from a single object. In addition to simulating dice of various sizes, this could also be used for a spinner.

Follow the steps above under Adding Components 🎲 to create a new die. If you just need a six-sided die, select **D6 Die** on the Create Component screen. For any other type of die, select **Die** on the Create Component screen and follow these steps:

1. Create an asset that has a grid of the different outputs starting in the upper-left corner. See Adding Assets & Variants 🖼️ for details.
2. Select **Components** in the left pane, and then select the + button.
3. Input a Name.
4. In the Sides box, input the number of different possible outputs and select Save.
5. Select **Variants > Variant 1**.
6. Select the asset that you uploaded.

You don't need to create more variants. Screentop will automatically randomize the output across the different images in the asset's grid from the first image up to the number entered in the Size box.

Counters

Use counters to track numbers, steps, phases, etc. in a game, such as score or resource counts.

Follow the steps above under Adding Components 🎲 to create a new counter.

Here are the most common attributes to configure:

- **Name**
- **Size**: This is the number of settings the counter will have, starting at zero. The counter's values will range from 0 to Size-1, so if you need to track and show numbers from 1 to 10, you may need to set your Size to 11.

To change the colors of the counter, see Variants. Note that the number will be the same color as the **Stroke Color**.

Adding Assets & Variants 🖼️

Assets

Assets are image files that can be used to define the appearance of your components. Assets typically include several images arranged in rows and columns in a grid, such as:

- playing-card-001.png is a grid of 5 rows and 13 columns
- d6-die-001.png is a grid of 2 rows and 6 columns

To create a new asset:

1. Select **Assets** In the left pane and select the + Button.
2. Input a Name.
3. Select the Image box and upload your image.
4. Input numbers for Rows and Columns (see sample images linked above for guidance).
5. Select Save.

Preparing Your Images

Design software enables you to make the best-looking visuals, as well as adjust the size and resolution of your artwork. Some options include:

- Gimp (free)

- Google Slides (free)
- Affinity Designer (paid, but cheaper than Adobe)

Images uploaded to assets can be JPG or PNG files. PNG files can be useful when you need a component that has transparent areas.

Images cannot be larger than 4096x4096 pixels. Setting the image resolution to 150 DPI is pretty common. Higher resolutions (DPI) will make components sharper and more readable. Compressing your images will reduce the file size of a file without changing the pixel count. Most design tools can perform this task, or you can try a web-based tool such as ResizePizel.

A deck building tool, such as nanDECK, Multideck, or Tabletop Creator, can be useful for creating large numbers of cards or other components that have repeating layout. These programs enable you to define values in a spreadsheet and then generate cards automatically in the grid layout needed for importing the image as an asset.

Variants

Use variants to define what a component should look like. Variants can be simple definitions of color, or they can display an image from an asset. Some components have a single variant (such as a board, for example), whereas others have multiple variants (such as a deck of cards or asymmetric player mats).

To define a variant for an existing component:

1. If you want to display an image, create an asset first (see Assets).
2. In the left pane, select **Components**, and select the component.
3. Select **Variants** and select **Variant 1**.
4. Edit attributes as needed and save.

The attributes vary slightly based on each component type, but most include a fill color, a stroke color, and the ability to select an image from an asset. Tiles include another set of attributes for the back of the tile.

Create additional variants for that component if needed. You can clone an existing variant by selecting the **3 dots** to the right of it and selecting **Clone**. Enter in the number of clones you'd like.

To

Recipes 📝

Steps for creating common objects used in many games.

Deck of Custom Cards

(Needs content - Create asset, create component, create variants, use bulk editor to assign asset index to variants)

Player Hands

To create containers for the player hands:

1. Select **Components** in the left pane and then select the + button.
2. Select **Container** and then **Create.**
3. Select the newly created **Container** at the bottom of the list.
4. Set the **Name** to **Player Hand**.
5. Set the **Width** to the width of your card component times the number of cards you'd like to display, plus 20 (to add a little margin on the edges). For example, if your cards are 125px wide and you want to show 5 cards in the hand, set it to 125 * 5 + 20 = 645.
6. Set the **Height** to the height of your card component plus 20. For example, if your cards are 175px, set it to 195.
7. Select **Save**

Next, to create a dropzone for cards in the container:

1. Select **Dropzones**.
2. Delete the default **Dropzone 1** by selecting the 3-dots on its right and selecting **Delete**.
3. Select the + button and select **Anchor** and **Create**.
4. Select the newly created dropzone and under **Components**, select your card component select **Save**.
5. Select **Anchors**.
6. Select the + Button and select **Line** and **Create**.
7. Select the newly created anchor.
8. Set **X0** to negative half of the width of the player hand container. For example if your width is 645, set it to -322.
9. Set **X1** to positive half of the width of the player hand container. For example if your width is 645, set it to 322.

10. If you want to limit the number of cards that can ever be placed in the hand, enter that number under **Capacity**. Note that if players are temporarily able to go above their hand limit (such as during their turn), then the **Capacity** should either be left blank or allow for this overage.
11. Set **Auto Rotate** to **0**, so that cards added to the hand match the rotation of this container.
12. For **Set Tile**, choose **Up** to make it so cards added to the hand are automatically face-up for the player.
13. Select **Save**.

Next, add the player hands to the surface:

1. Select the **Treasure Chest** icon in the upper-right corner.
2. Select the player hand container, select Variant 1 and select **Create**.
3. Copy-and-paste the player hand container on the surface by selecting it, pressing **Ctrl+c**, and then pressing **Ctrl+v** until you have enough hands for the number of players.
4. Click-and-drag containers to orient them around the surface for each player's area. You can rotate them by pressing **e** or **q**, but consider whether or not this is actually needed–for many games, having all the hands oriented in the same direction makes it easiest for players.

Finally, hide cards from all other players by performing these steps for each player hand:

1. Right-click a player hand container on the surface and select **Seats**.
2. Select the player seat that is allowed to see the cards for that container. If you've added an Admin seat, also select that seat to also allow the Admin to see the objects.
3. Select **View Policy** in the right-click menu and select **Allow Seats**.

After configuring this, you can try out your configuration:

1. Add one or more cards to each hand.
2. Select your name in the upper-left corner.
3. Select the leave icon to the right of your name.
4. Select **Join** and select a different player seat.
5. Repeat steps 2-5 to test each player's seat.

Optionally, players can pick up their containers, so that each container is only visible to that player and they "float" above the surface at the bottom of the screen. To hold a container, either:

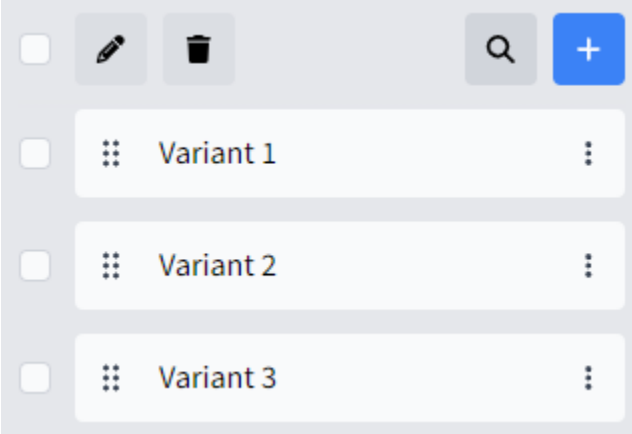
- Right-click the container and select Hold
  OR
- Select the container and press **h**.

Deck and Discard Containers

(Needs content - create container with point anchor, set up/down state, appropriate sizing)

Bulk Editing 🤖

You can apply bulk edits to any items in Screentop that have the following type of interface. Some of the most common items to bulk edit include Variants and Anchors:



Follow these steps to edit items in bulk:

1. Ideally, configure the first item with all the common attributes that you'd want to copy to all the other items.
2. Select the **3-dots** to the right of the item (as shown above), select Clone, and enter the number of clones to make.
3. Set the checkboxes on the items to bulk edit and then select the **pencil** button:
    a. **Select all**: To select all the items, select the checkbox to the left of the pencil button (as shown above)
    b. **Select some**: Manually check the boxes that you'd like to edit.

    NOTE: When selecting only some of the boxes, the **index** in the code is a relative reference to the items selected, not its absolute place in the list. The first item you manually selected will be index #1, regardless of where it falls in the list.

4. You can apply bulk edits in 2 different ways:
    a. **Code**: Update the Javascript code on the left side and select **Save** to apply.
    b. **Manual**: Manually edit the values of each attribute on the right side and select **Save** to apply.

Understanding the Code (for Non-Coders)

Each item in the list has a number and then a bunch of attributes between brackets { }.

The attributes generally correspond to the fields that you can configure when you edit that item manually. Each attribute is listed on one line with the name of the attribute on the left, followed by a colon, followed by the current value of that attribute.

When making bulk edits manually, simply edit the values that you'd like to change on the right side of the colons.


```
  "1": {
    "kind": "TILE",
    "name": "Variant 1",
    "labels": [],
    "frontFillColor": "#ffffff",
    "backFillColor": "#8e7cc3",
    "strokeColor": "#000000",
    "frontAsset": "Action Cards",
    "frontAssetIndex": 1,
    "backAsset": null,
    "backAssetIndex": 1
  },
  "2": {
    "kind": "TILE",
```


Notes for Coders

The index value starts with 1, not 0.

Code Snippets

See the official Code Snippets.

Please copy/paste this first code snippet when adding additional ones to keep formatting consistent.

Apply images to all the variants of a tile component

```
function updateVariant(variant, index) {
  return {
    frontAssetIndex: index,
  };
}
```

Apply images to all the variants of a tile component, when there are more cards that can fit within one asset

Assume you have two assets named “Asset 1” and “Asset 2” with up to 100 card images each.

```
function updateVariant(variant, index) {
  const imagesPerAsset = 100;
  return {
    frontAsset: "Asset  " + Math.floor((index - 1) / imagesPerAsset) +
1,
    frontAssetIndex: (index - 1) % imagesPerAsset + 1,
  };
}
```

Create a hex grid of point anchors in a container

Adjust the constant values for tileWidth, tileHeight, countAcross, and yAdjust to fit your particular grid. For the yAdjustEven and yAdjustOdd constants, one of them needs to be 0 and the other 0.5.

```
function updateAnchor(anchor, index) {
  const tileWidth = 45;
  const tileHeight = 51;
  const countAcross = 7;
  const yAdjust = -37.5;
  const yAdjustEven = 0
  const yAdjustOdd = 0.5;

  let xvalue = -(tileWidth * (countAcross/2-.5)) + (tileWidth *
((index-1) % (countAcross)))-1;
  let yvalue = 0;
  if (((index % countAcross) !== 0) && ((index % countAcross) % 2 ==
0)) {
```

```
    yvalue = -(tileHeight * (countAcross/2-yAdjustEven)) - yAdjust +
(tileHeight * Math.floor((index-1)/(countAcross)))
  } else {
    yvalue = -(tileHeight * (countAcross/2-yAdjustOdd)) - yAdjust +
(tileHeight * Math.floor((index-1)/(countAcross)))
  };

  return {
    "x": xvalue,
    "y": yvalue
  }
}
```

Advanced Techniques 🔥

Screentop Tips & Tricks, an interactive tutorial for experienced users built in Screentop.gg by Jack Rosetree.

Keyboard Shortcuts

See the official list of Keyboard Shortcuts.

Edit Mode Shortcuts	
Edit Object	o
Edit Component	c
Edit Variant	v

Performance Tips

See the official Performance Tips.

Fields

(Needs content)

You can accelerate prototyping by adding emojis to fields in place of adding some assets.

Hold Objects

(Needs content)

Bulk Edit / Delete Objects

In the left pane, select a surface or a container that contains the objects. Select Objects at the bottom of the pane. Use cases include:

- Deleting all objects from a surface no longer in use
- Bulk updating the X, Y coordinates of many objects
- Bulk updating other object attributes, including: Locked, View Policy, Interact Policy, and Variant.

Control Z-Order Layering

When multiple objects of different component types are placed in the same container, you can control which objects appear on top of each other by reordering dropzones in the list for that container.

Use the 6 dots to the left of the name to click-and-drag a dropzone. Objects on dropzones that are higher up in the list of dropzones for a container appear behind objects on dropzones that are lower in the list.

For this to work well:

- Each dropzone need to limit to one or more types of components
- No two dropzones should limit to the same types of components

Hidden Movement

To prevent opponents to see objects that are selected or moved within a container:

1. Edit the container.
2. Set the Hidden Style to be **Closed**.
3. Update the Cover Fill Color of each variant of the container to be a solid color (remove the **80** at the end of the default color hex code).