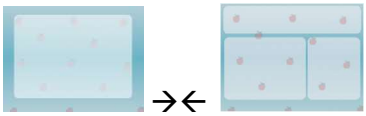
VARIABLES DIALOGUE

TUTORIAL

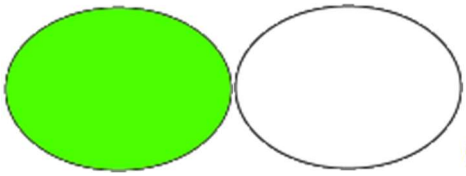
Top bgs:

Bottom bgs:



Assets used:

resized



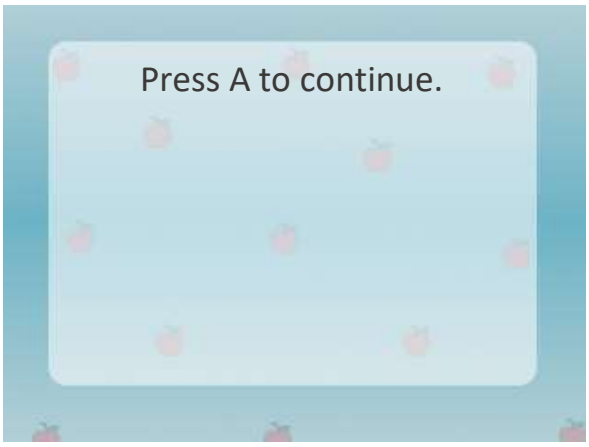
resized

resized

resized

PANEL TOP.1; BOTTOM.1

TEXT:



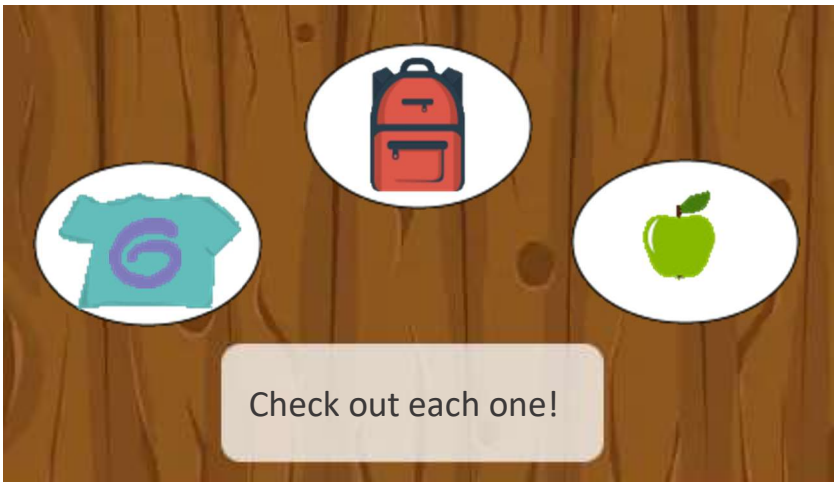
PANEL 2: TOP.1; BOTTOM.1 (recycled; same as before)

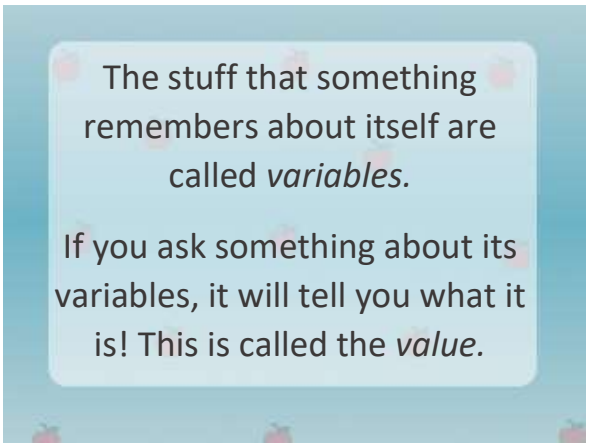
NEXT SCENE:

Assets: 3 highlighting circles with (1) Shirt, (2) backpack, and (3) apple inside them. User can cycle through the objects, click 'A', and access data.

TEXT: on object highlight

PANEL 3: TOP.2; BOTTOM.1

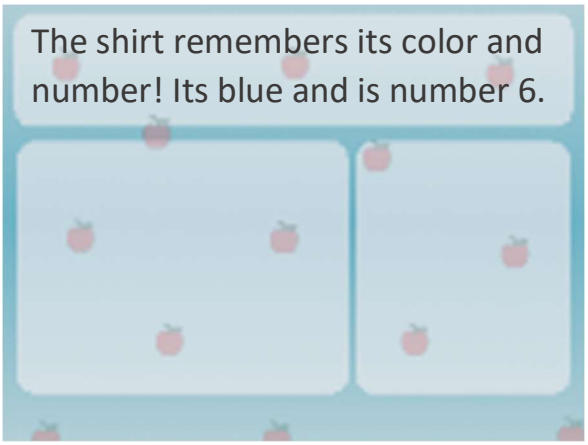




First bottom screen; press 'A' to progress to next set

PANEL 3.2: TOP.2; BOTTOM.2;

Asset change: highlighting first object



On highlight: display object intro info

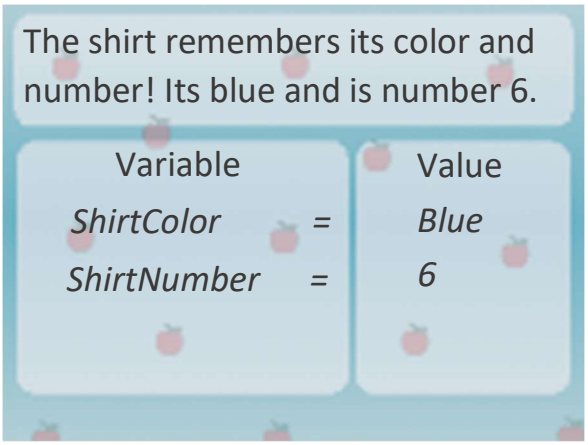
On ‘A’ press of highlighted object: display data (new text only)

Once they’ve pressed ‘A’, change text in box to indicate they should use D-pad to navigate.

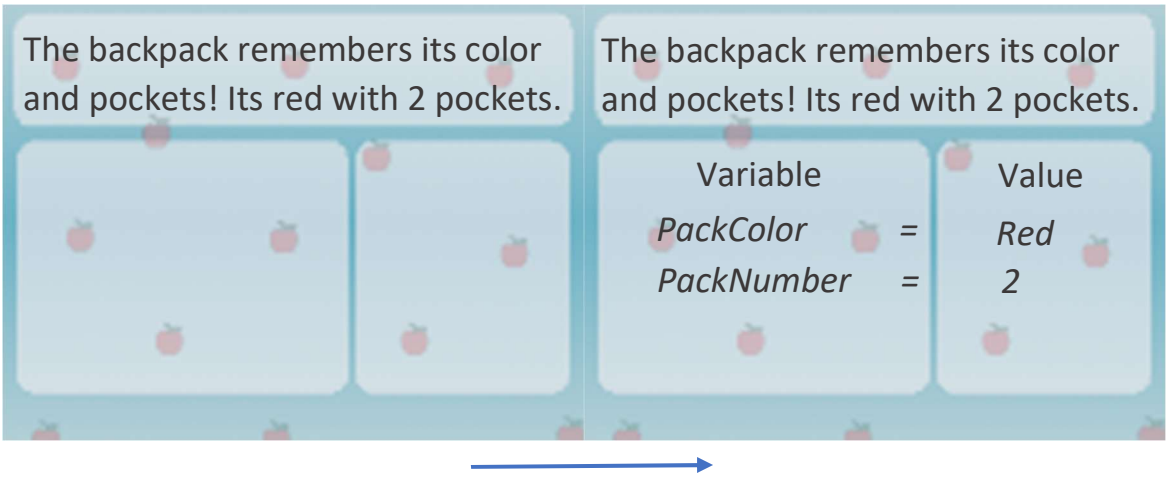
Would follow logic like:

If (object is not visited) : text = “[name]: press ‘A’!””

Else: text = “[name]: press ->!”



On D-pad press: clear data text and update with intro info



On 'A' press of highlighted object: display data (new text only).
Population of data can stay per object permanently once visited = true.

SCENE: user cycles through each object – 3;

Last Object to be marked:

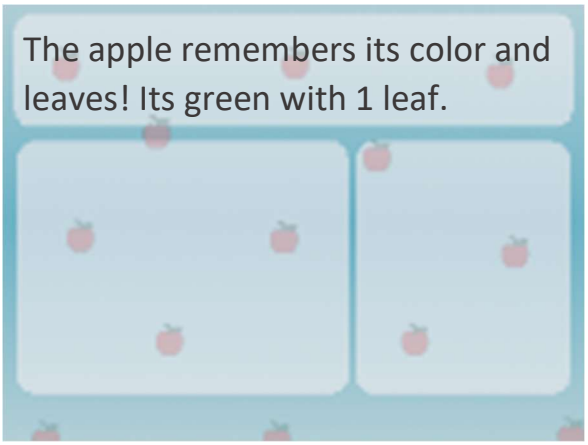
The apple remembers its color and leaves! Its green with 1 leaf. *(not shown)*

PANEL 3.3: TOP.2; BOTTOM.2

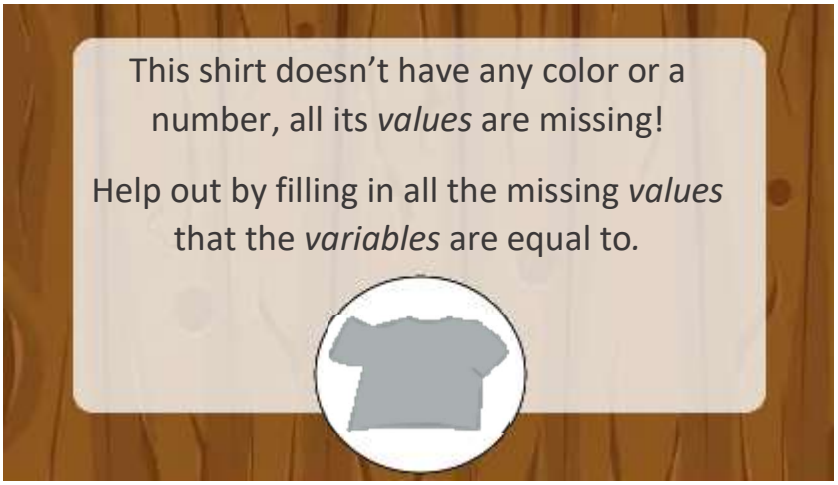
Meeting requirements to move onto next scene: to allow them to press A on the apple again without accidentally moving on, change trigger to 'start' button.

Will help with introduction of start being a functional press for future gaming.

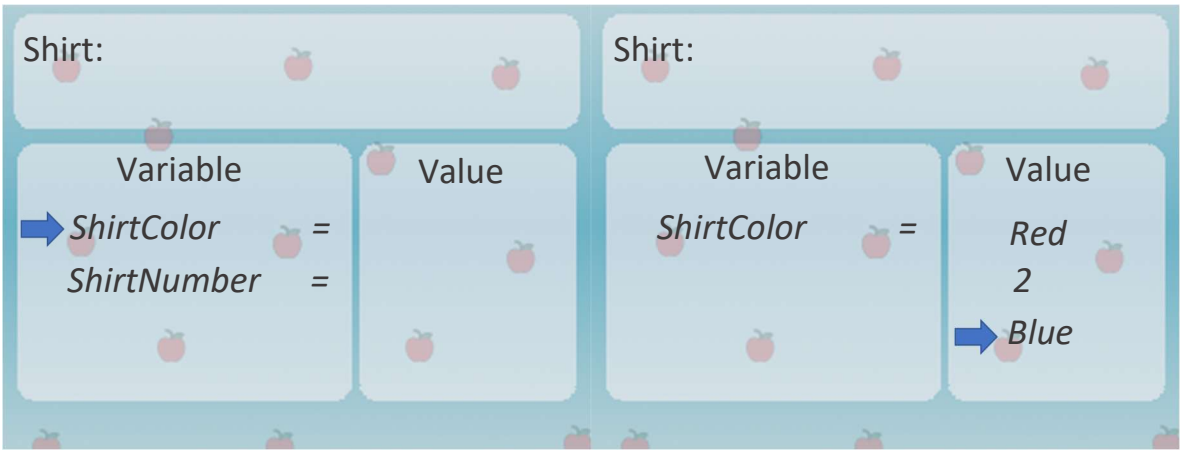
Bottom screen can either revert or stay with data populated.



PANEL 4: TOP.3; BOTTOM.1;



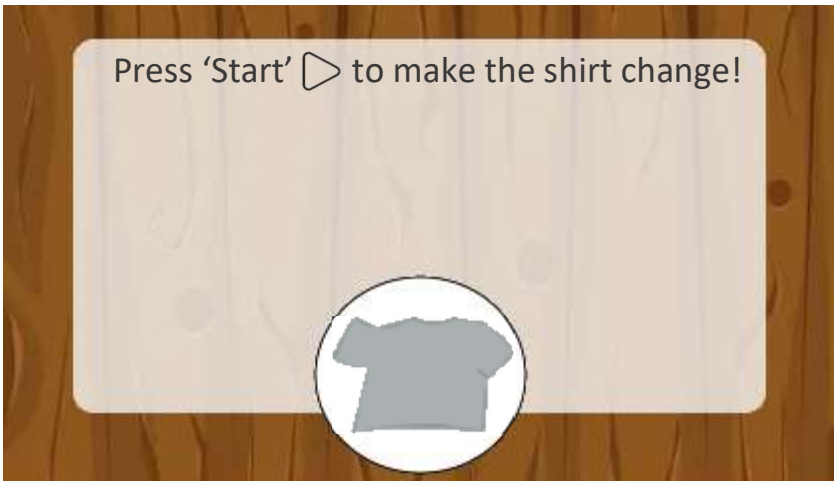
PANEL 5: TOP.3; BOTTOM.2



On 'A' press of highlighted object: open second value menu and offer interaction.

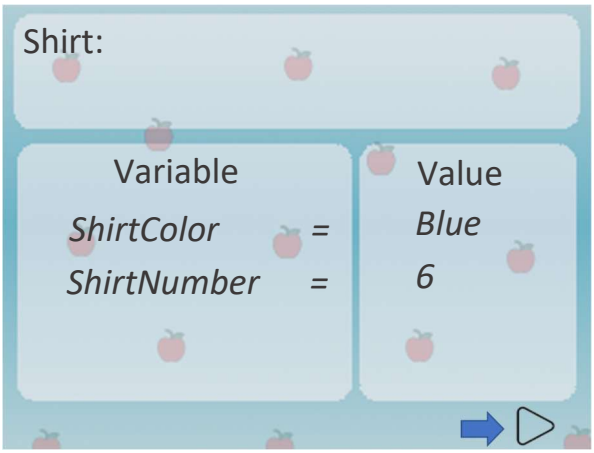
***NOTE: not imperative that the second variable (see second bottom screen) disappear. Would be ideal but not necessary.*

PANEL 6: TOP.3; BOTTOM.2



On BOTH values assigned, top screen update.

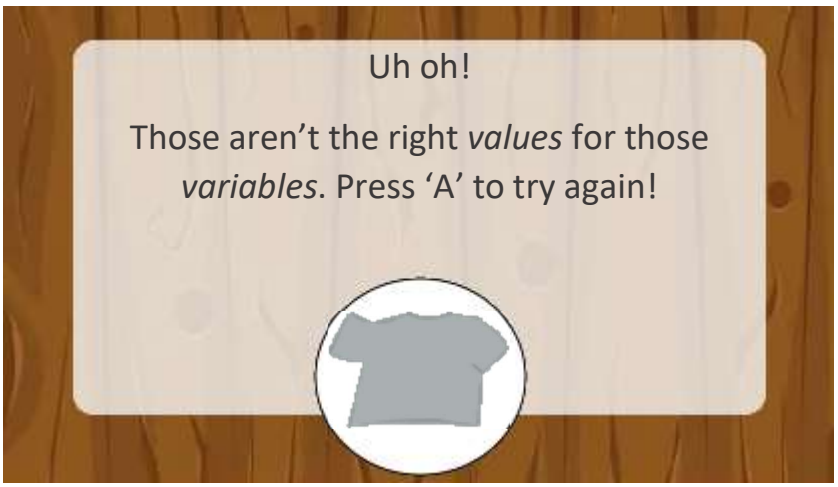
The start asset in the middle of this text will require some manual whitespace inserted in text inside code.



Populate “run” button for first time. Indicate (this time only) with arrow.

Actual minigame will not restrict the start asset population, or indicate with an arrow.

PANEL 7 ERROR: TOP.3; BOTTOM.1 *unchanged bottom screen*

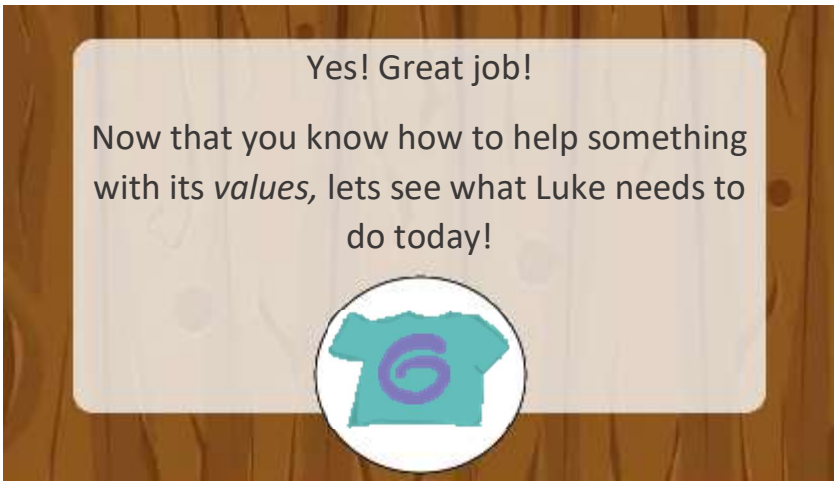


ERROR: user picks wrong input.

Leave bottom screen as is, or blank it to a bottom bg with no text – whichever is easier.

On ‘A’, return to PANEL 5 starting scene.

PANEL 7: TOP.3; BOTTOM.1



SUCCESS: final scene

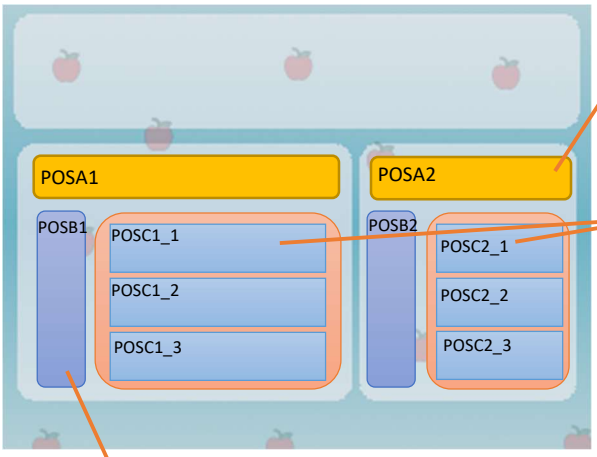
Replace gray shirt asset with blue + 6 asset combo.

Tutorial conclusion.

Pressing ‘A’ may lead immediately into active minigame.

Some notes:

- Is italicizing even possible? If not, leave as normal and brainstorm a fix later.
- Batching text and assigning it to (x, y) coordinates might be best
    - Ex: “Press A to continue” reused a lot
- Boxing on BOTTOM.2 might have some very specific coordinates to keep spacing.
    - See below notations:



Need consistent space for table headers.

Segment the remaining space here into 3 even sections. Both sides of table should have matching y-values. The '=' sign is just a part of the text with suitable whitespace separating it.

Need consistent space for arrow indicator asset.

Can have coordinates matching POSA1, POSA2, POSB1, POSB2, POSC1_1, POSC1_2, POSC1_3, POSC2_1, POSC2_2, POSC3_3

Where

POSA1.y = POSA2.y

POSB1.y = POSB2.y

POSC1_1.x = POSC1_2.x = POSC1_3.x

POSC2_1.x = POSC2_2.x = POSC2_3.x

POSC1_1.y = POSC2_1.y

POSC1_2.y = POSC2_2.y

POSC1_3.y = POSC2_3.y