4 Steps to APP CREATION:

Step 1: Define a Purpose: What the app should do.

Step 2: Design the App: We select the design components we want in our app.

Step 3: Code the App: Use required code blocks.

Step 4: Run the App: Run the app and check if it works.

**Links to be shared with students:**

COLOR CHANGER APP:

<https://appetize.io/app/p53f9844wu5m2gq98mgaebnbbw?device=nexus5&scale=100&orientation=portrait&osVersion=8.1>

PAGER CHAT APP:

<https://appetize.io/app/2qfdcpaf71za9p16mnwphq8ew8?device=nexus5&scale=100&orientation=portrait&osVersion=8.1>

SUN-EARTH SYSTEM:

<https://appetize.io/app/620bw8j5137539cx74mwq1q93w?device=nexus5&scale=100&orientation=portrait&osVersion=8.1>

TIC TAC TOE CLOUD GAME:

<https://appetize.io/app/24ty0njc34ck31g2mjq9hpejh8?device=nexus5&scale=100&orientation=portrait&osVersion=8.1>

IGISHKA

<https://bot.dialogflow.com/82e2fc96-1906-4bd8-a9d2-84a17f04ea54>

hi

how are you feeling

favorite colour

do you like reading/ books

which sports do you like

what do you eat

GSLV MARK III LAUNCHER VEHICLE:

<https://www.tinkercad.com/things/lriqDFOnu7A>

Coding Tricks:

Rgb(r, g, b, a)

, where a is the opacity of the color

a=0 is transparent; a=1 is opaque

0.5 is the medium value, 0.2 shows transparent/ see through/ pastel shades, 0.9 shows opaque shades

Higher the transparency, darker the shades

If randomNumber is used for all three primary colors with range 0 to 255, all shades of all colors will be displayed randomly.

onEvent("button1", "click", function() {

setProperty("screen1", "background-color", rgb( randomNumber(5, 245), randomNumber(0, 255), randomNumber(0, 255), 1.0) );

});

To make the button circular:

1. Height=width=radius= 90 (radius equal to the size of button)
2. Height=width= 100; radius= 50 (half of size of button)
3. Height and width should be same, the value of radius should be half or more than half, even if radius is 1,00,000, it will be circular

To keep button square:

Size of button height and width = equal; radius = 0 (increase value to get curved corners)

To change color of screen every second:

onEvent("button3", "click", function( ) {

timedLoop(1000, function( ) {

setProperty("screen1", "background-color", rgb(randomNumber(0, 255), randomNumber(0, 255), randomNumber(0, 255), 1.0) );

});

});

To add turtle to screen:

If same value of forward and backward movement, it will make a circular pattern of lines like sun rays and turtle will keep at one central point and move forward and backward.

Whichever has the higher value, turtle will move in that direction and will go out of the screen then.

onEvent("button3", "click", function( ) {

timedLoop(1000, function( ) {

setProperty("screen1", "background-color", rgb(randomNumber(0, 255), randomNumber(0, 255), randomNumber(0, 255), 1.0) );

penWidth(3);

moveForward(25);

penColor("red");

turnRight(randomNumber(1,50));

});

});

Screen color change by click of button, turtle added for 1000 milliseconds, hide elements, screen color changed every, millisecond;

onEvent("button5","click",function() {

penColor("red");

penWidth(3);

timedLoop(1000, function(){

setProperty("screen2","background-color", rgb(randomNumber(0, 255), randomNumber(0, 255), randomNumber(0, 255), 0.5));

turnRight(randomNumber(1,50));

moveForward(25);

hideElement("button5");

hideElement("button6");

hideElement("label1");

hideElement("text\_input1");

});

});

onEvent("button6", "click", function(){

setProperty("screen2","background-color", getText("text\_input1"));

});

MoveForward(25);

turnRight(45); = makes octagonal

(25,45) = octagonal

(25,30) = bigger 12 sided hexagonal

(25,40) = 9 sides

(25,45) = 8 sides octagonal

(25, 60) =6 sides hexagonal

(25,90) = 4 sides square

(50,90) = 4sides bigger square

(100,90) = even bigger square

(200,90) =biggest square