



Topic	APPLICATION DESIGN		
Class Description	<b>We start by de-constructing advanced app creation into sequential process elements. Kids create the first real-world app using code and design to gain creation confidence.</b>		
Class	<b>ADV-T</b>		
Class time	<b>50 mins</b>		
Goal	<ul style="list-style-type: none"> <li>● Understand the process of creating real world apps</li> </ul>		
Resources Required	<ul style="list-style-type: none"> <li>● <b>Tips for Teachers</b> <ul style="list-style-type: none"> <li>○ If the student seems quick you can ask the kid to switch to TEXT MODE.</li> <li>○ Invite Parents in the class to showcase the products.</li> </ul> </li> <li>● Teacher Resources <ul style="list-style-type: none"> <li>○ Code.org login: It is advisable for teachers to create 2 dummy gmail ids of their choice beforehand and share the same with trial class students for smooth code.org login</li> <li>○ Earphones with Mic</li> <li>○ Notepad and Pen</li> </ul> </li> <li>● Student Resources <ul style="list-style-type: none"> <li>○ Code.org login</li> <li>○ Earphones with Mic (Optional)</li> <li>○ Notepad and Pen</li> </ul> </li> <li>● Hygiene Factors <ul style="list-style-type: none"> <li>○ Have a clean and clear background</li> <li>○ Always keep your camera on</li> <li>○ Sufficient Lighting on your face (light should be facing your face and not your back)</li> <li>○ Keep your phone on silent (Refer How To Ace Trial Class Section for more details)</li> </ul> </li> </ul>		
Class structure	<b>Warm Up</b> <b>Teacher-Led Activity</b> <b>Student-Led Activity</b> <b>Wrap-Up</b>		<b>10 Mins</b> <b>15 Mins</b> <b>15 Mins</b> <b>10 Mins</b>
<b>Start the video call from H2H</b>			

Class Steps	Say	Do
Ask the student to get into Fullscreen mode.		
<b>Step 1:</b> Warm up (10 minutes)	<p>Hi, my name is _____. May I know your name?</p> <p>I am going to be your instructor in this session. Let's get to know each other a little bit before we start. First I will share a brief introduction about myself then I would love to know more about you.</p> <p>I have done &lt;XYZ certification&gt; and my hobbies are &lt;Teacher's interest areas&gt;</p> <p>&lt;Student Name&gt; Please tell me about which grade are you in, which areas interest you?</p> <p>&lt;Wait for their response&gt;.</p> <p>So &lt;student's name&gt;, why do you want to learn to code? and &lt;if parents are present, ask them&gt; what are your expectations from this class?</p>	
!! IMPORTANT SOP !!		
<u><b>CHECK THE TRIAL STUDENT Ad type displayed SOURCE ON THE DASHBOARD</b></u> <ul style="list-style-type: none"> <li>If the Ad type displayed = Roblox, then check whether the student has installed Roblox already. If installed then do the Roblox activity in the Green section below.</li> <li>Else, skip the Green section &amp; carry on with the activity in the Blue Section.</li> </ul>		
<b>Step 2.1:</b> <b>Teacher Activity 2</b>	<u><b>Set the Agenda</b></u> Let me quickly tell you <b>What we are going to do in this Trial Class:</b>	Play the Roblox Game yourself while Sharing your screen.

 <b>Roblox Activity</b>	<ol style="list-style-type: none"> <li>1. I will be teaching you the steps to build a “BASIC MOBILE APP”</li> <li>2. You will then be building the Mobile App yourself with some guidance from me</li> <li>3. Towards the end I will be presenting the course structure and student outcomes which will help you understand exactly what we offer</li> </ol> <p>I know you are interested in Roblox, So we will do a fun activity. This game is built in 3D using Roblox.</p>	
<b>Teacher Initiates Screen Share</b>		
	<ul style="list-style-type: none"> <li>• <u><a href="#">Blazing Obby ( Roblox Game)</a></u>  <u><a href="#">[Teacher Activity 2]</a></u></li> </ul>	Play the game and explain the instructions after you have played, ask the child if they would like to beat you in the game.
<b>Teacher Stops Screen Share</b>		
	Now, would you like to play this game and beat me in it?	Share the link with the child.
<b>Ask Student to Share their Screen</b>		
<b>Step 2.3: Student led Activity</b>	Let's Start playing the game In the Student Panel, click the link for Student Activity 5.	Help the student in creating an account and complete all necessary steps you have done to play the game.

 <b>Roblox Activity</b>	<p>Would you like to create games of this kind?</p> <p>ESR: Yes</p> <p>We will create 3D games and publish them during the course. Are you excited about this?</p> <p>ESR-Yes</p>	 <b>Roblox Activity</b>
<b>Student Stops Sharing their screen</b>		
<b>Agenda Setting</b>	<p>Let me quickly tell you <b>what we are going to do next in the Trial Class:</b></p> <ol style="list-style-type: none"> <li>4. I will be teaching you the steps to build a “BASIC MOBILE APP”</li> <li>5. You will then be building the Mobile App yourself with some guidance from me</li> <li>6. Towards the end I will be presenting the course structure and student outcomes which will help you understand exactly what we offer</li> </ol>	<p>In case the parents are not there please Invite the parents to join the Class. If they are busy, ask them to join in the last 10 minutes if possible.</p>
<b>IMPORTANT</b>	<p>Before going ahead, I want to ask you something.. Let me share my screen.</p>	<p><i>Share your screen and open <a href="#">[Teacher Reference]</a> (<b>Did you Know?</b>) and ask questions present in the slides to build the excitement.</i></p>
<b>Teacher Initiates Screen Share</b>		
	<p>Yes, a Satellite.</p>	<p><i>ESR: “With the help of Satellite.”</i></p>

	<p>And you know what? In our Space explorer program, You will be able to communicate &amp; code a real satellite, click a picture of the earth, predict weather or natural disasters and you can even track the position of the satellite. Sounds like a superpower, isn't it?</p> <p>Today, We will start with building a simple app and then progressively build complex games, apps &amp; websites after a few classes.</p>	
--	--	--

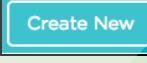
### Teacher Stops Screen Share

	<p><b>Today we will learn to make a basic mobile app.</b></p> <p>What do you typically do using your parents phone?</p> <p>Yeah! You are using many apps on the phone.</p> <p>For example: Camera is an app, which lets you take pictures. Music is an app which lets you play music.</p> <p>We create <b>APPS</b> for a purpose.</p> <p>Today, we will create an app to change the color of a screen when a button is pressed.</p>	<p>Encourage students to share examples. <b>Sample Answer:</b> <i>Youtube video, Games, Music, Click pictures</i></p>
--	---	---

<b>Step 2.2: Teacher Activity 1</b> (15 minutes)	<p>One of our big goals is <b>APP CREATION</b>. There are 4 steps we will follow to do this.</p> <p><b>Step 1: Define a Purpose:</b> What the app should do.</p> <p><b>Step 2: Design the App:</b> We select the design components we want in our app.</p> <p><b>Step 3: Code the App:</b> Use required code blocks.</p> <p><b>Step 4: Run the App:</b> Run the app and check if it works.</p> <p>Now, let's discuss these steps for the app that we want to create right now.</p> <p><b>Step 1: Define a Purpose:</b></p> <p><b>Q</b> What did we say is our purpose for the app?</p> <p><b>A</b> When I click the button the color of the screen should change to my favorite colour “green”.</p> <p>Which means:</p>	 <p>Click this icon in the bottom left of the screen. This is the Chat Window.</p> <p>Copy Paste the following lines <b>one by one</b> into the Chat Window (if you copy paste all the four steps together, they will be sent as a paragraph):</p> <p><b>Step 1: Define a Purpose:</b> What the app should do. <b>Step 2: Design the App:</b> We select the design components we want in our app <b>Step 3: Code the App:</b> Use required code blocks <b>Step 4: Run the App:</b> Run the app and check if it works</p> <p>Encourage the student to reply with the purpose of the app.</p>
---	---	--

	<p><b>A: When I click the button</b>  <b>B: Color of the screen should change to green.</b></p> <p><b>Q</b> Now can you give me some examples of things that work by the click or press of a button?</p> <p><b>A</b> To switch on/off a light, we press a switch button. Similarly, for TV switch on off, we use the power button on Remote/TV</p> <p>You did really well and you deserve a <b>big round of applause!</b></p> <p>So, <b>Step 1</b> is now complete.</p> <p>I will share my screen and show you the rest.</p>	<p>Encourage the student to respond.</p> <p>On the top left corner, press the  icon It will open up a tray of emojis</p>  <p>Press the Clap Emoji</p>
--	--	--

### Teacher Initiates Screen Share

- When you open the Activity link, it will prompt for sign-in.
- After login, press  in the top menu bar.

	<p><b>Step 2: “Design the App</b>  Choose the design elements we want.”</p>	<p><a href="#"><u>Teacher Activity 1: APP LAB</u></a></p> <p>Go to <b>Design Mode</b> on top left.</p> <p><b>Design Mode</b></p>
--	---	--

This is our **design** environment.

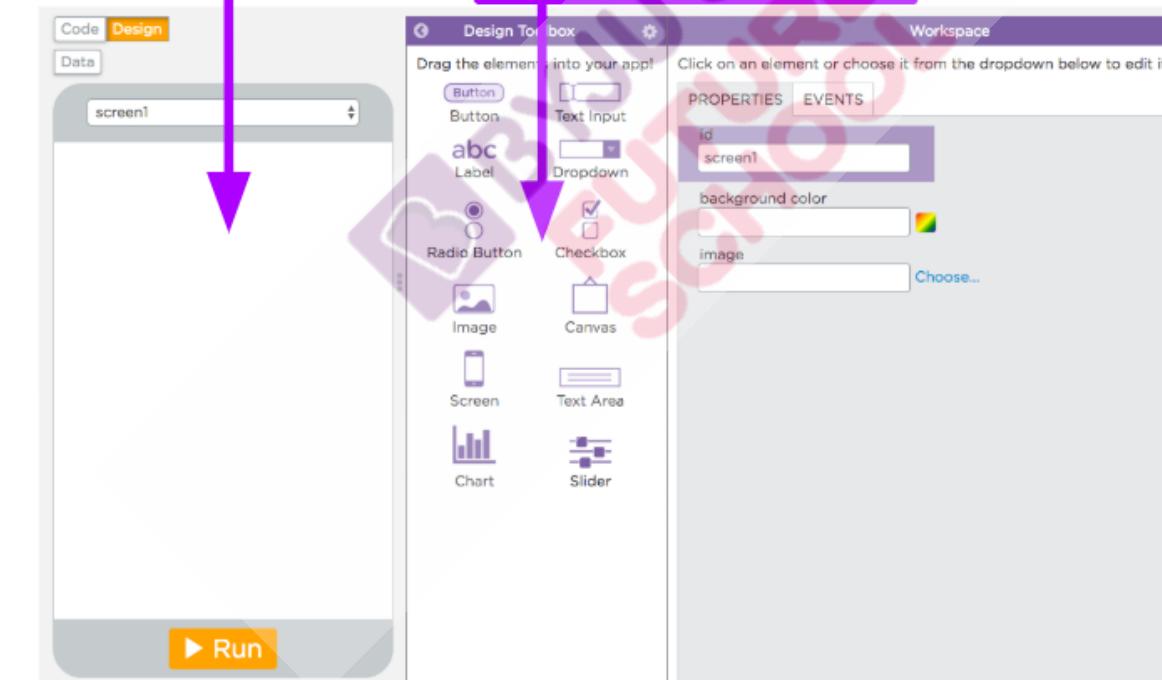
Code   Design   Data

Explain panels as shown in the image below.

- **Mobile Screen**
- **Design Toolbox**

**Mobile Simulator**  
**This shows what you will see on your mobile screen when you run the app**

**Design Toolbox**  
**All the app design components are here to drag and drop on the mobile screen**



The design elements we will use are **SCREEN** and **BUTTON**.  
 We already have a screen which we will use.

**Click on Screen**

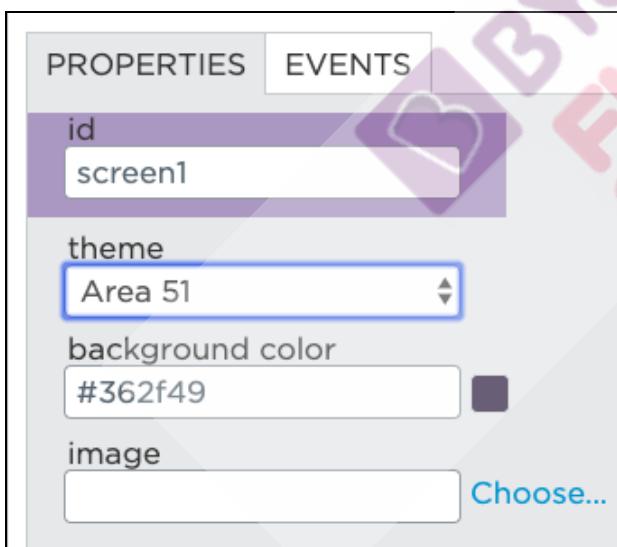
	<p>Let's <b>set a theme</b> for our Screen.</p> <p><b>Themes</b> are prebuilt design templates that get applied to your screen.</p> <p>Now from the <b>DESIGN TOOLBOX</b>, I will select and drag <b>BUTTON</b> and drop it into the <b>MOBILE SCREEN</b>.</p>	
--	--	--

### Let's Design The Screen

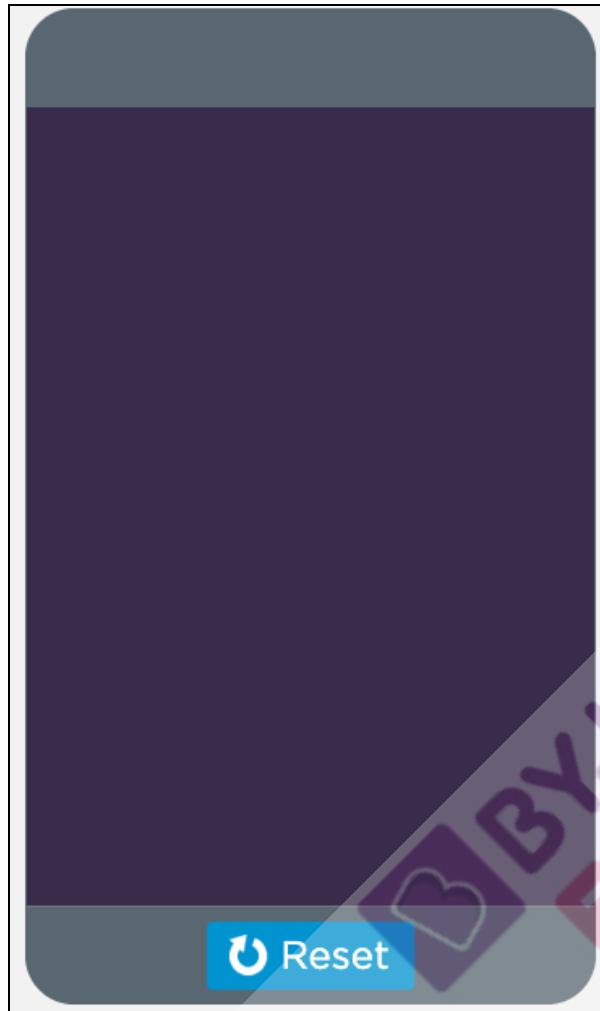
1. Toggle to 'Design'  and click on Screen1.



2. On the right, in the Properties section, select any THEME of your choice.



Output with Theme applied on the Screen:



BYJU'S  
FUTURE  
SCHOOL

Now let's **add Elements** on Screen.

**1. Goto Design Mode**

Code    **Design**    Data

screen1

**2. Drag Button**

x: 115, y: 325  
id: button1

Button

▶ Run

Design Toolbox

Drag the elements into your app!

- Button
- Text Input
- Dropdown
- Label
- Radio Button
- Checkbox
- Image
- Canvas
- Screen
- Text Area
- Chart
- Slider

Workspace

Click on an element or choose it from the dropdown below

PROPERTIES EVENTS

**id**: button1

**text**: Button

**width (px)**: 80

**height (px)**: 30

**x position (px)**: 115

**y position (px)**: 325

**text color**: #ffffff

**background color**: #1abc9c

**font size (px)**: 14

**text alignment**: center

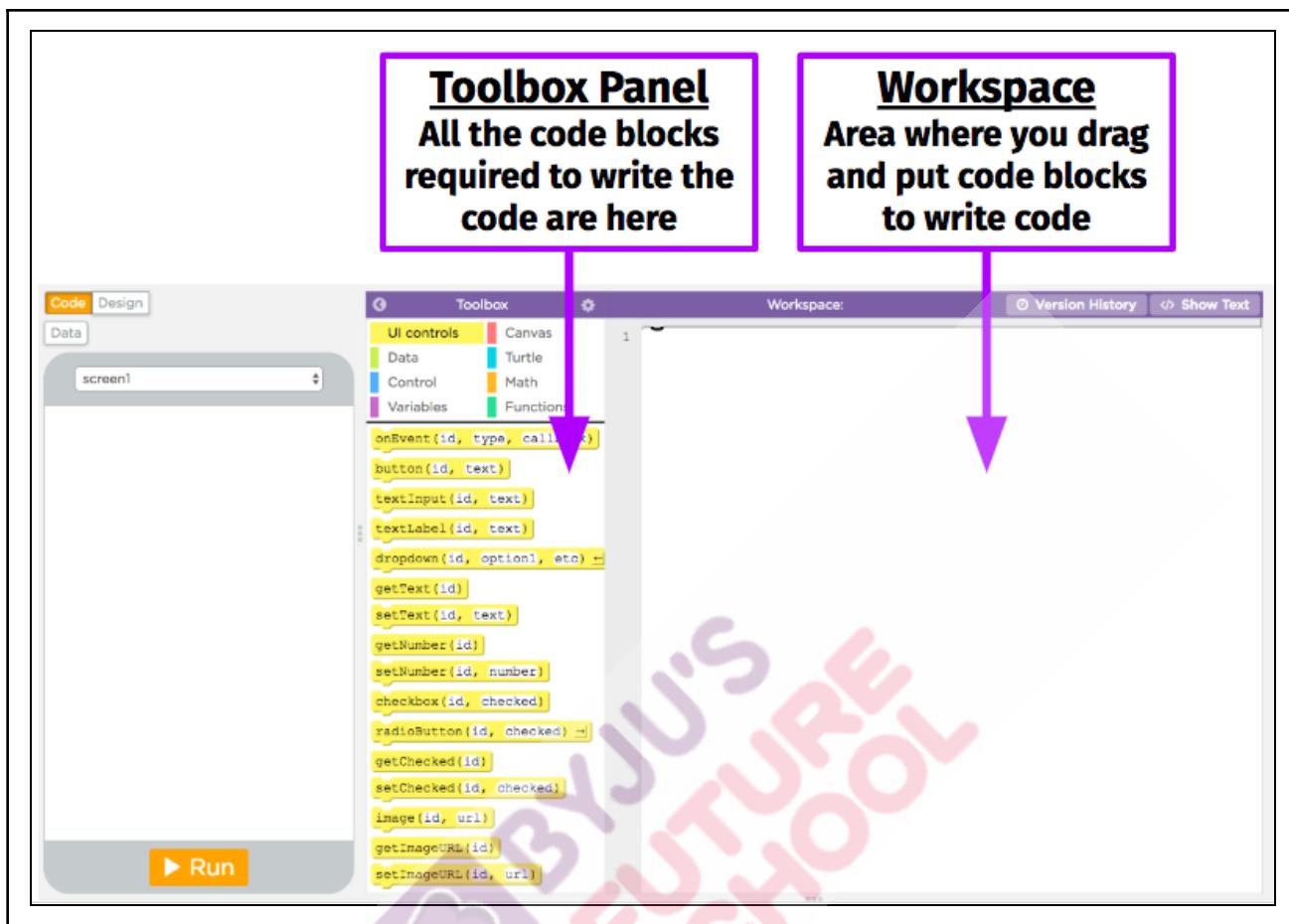
You can change the text of the button in the Properties section.

PROPERTIES EVENTS

**id**: button1

**text**: Click Me!

	<p>Now we will discuss the <b>3rd step</b> which is to <b>Code the App</b>.</p> <p>We will use code blocks to code our app.</p> <p>The Computer Language we are using to make this app is called <b>JAVASCRIPT</b>.</p> <p>This programming language is used to build apps and websites. Nearly 99% of all internet websites use <b>Javascript</b>.</p> <p>This is what our <b>CODING ENVIRONMENT</b> looks like.</p>	<p>Go to <b>Code Mode</b>.</p>  <p>Explain panels as shown in the image below:</p> <ul style="list-style-type: none"><li>• <b>ToolBox</b></li><li>• <b>Workspace</b></li></ul>
--	---	--



	<p>Now there are 2 ways in which we can write our program or code.</p> <ol style="list-style-type: none"> <li><b>1. Block Based Coding:</b> In this type of coding the commands are already present in the form of blocks. You just have to drag and drop these commands. This is an easier and faster way to code.</li> <li><b>2. Text Based Coding:</b> In this type of coding, you have to type the commands. This is how most programmers program - by typing the program.</li> </ol> <p>In this course we will be learning both the methods of coding.</p> <p>We will start with block coding and then when you feel confident we can switch to text so that you can code like professional programmers.</p> <p>Let me show you how we can switch between Block Mode and Text mode.</p> <p>I am going to first code in blocks and then in text.</p>	
		<p>Select required code blocks from the <b>TOOLBOX PANEL</b> and drag them into the <b>WORKSPACE</b> area.</p> <p>To see the full list of all events, scroll in the <b>Toolbox area</b>.</p> <p>Drag the <b>onEvent</b> block.</p>

Drag the **setProperty** block.

**1.Goto Code Mode**

Code Design

screen1

x: 148, y: 34  
id: screen1

Button

Run

Toolbox

UI controls

- Canvas
- Data
- Turtle
- Control
- Math
- Variables
- Functions

```

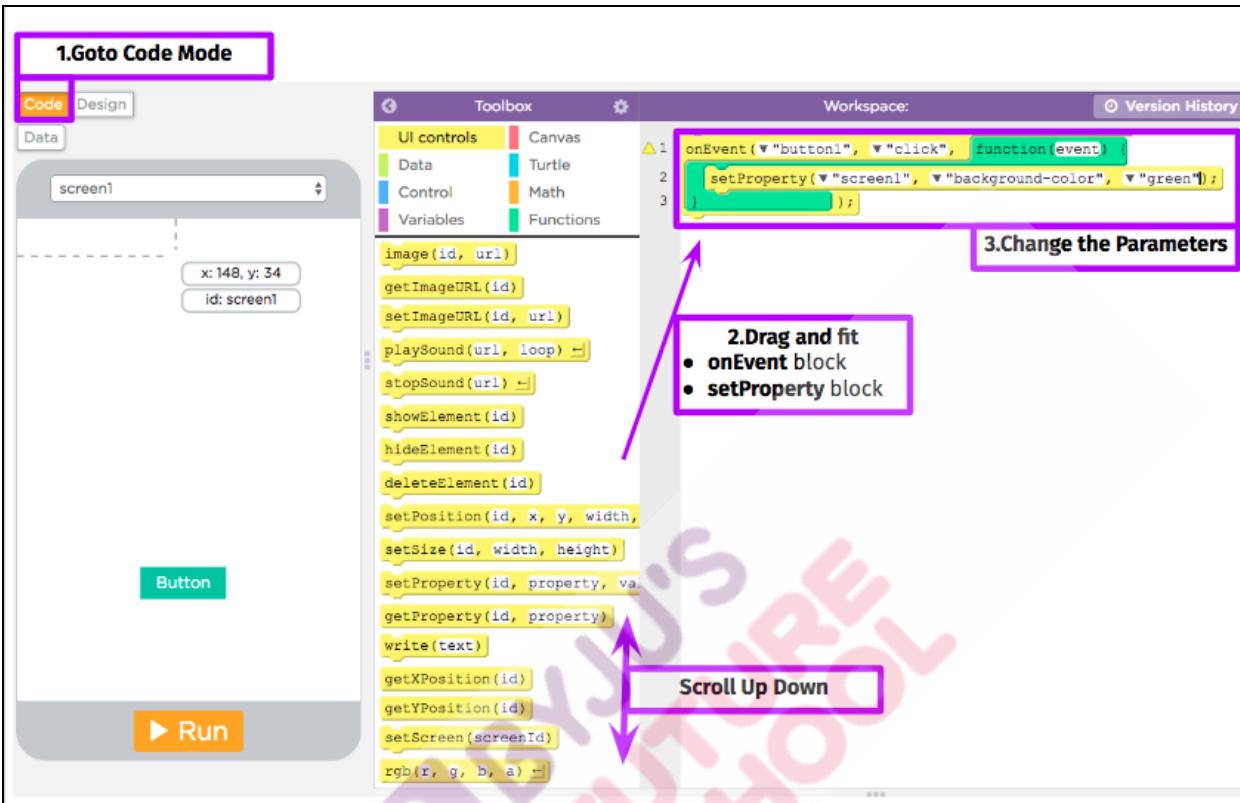
1 onEvent("button1", "click", function(event)
2     setProperty("screen1", "background-color", "green");
3 )

```

**3.Change the Parameters**

**2.Drag and fit**  
 • **onEvent** block  
 • **setProperty** block

Scroll Up Down



Based on our purpose,

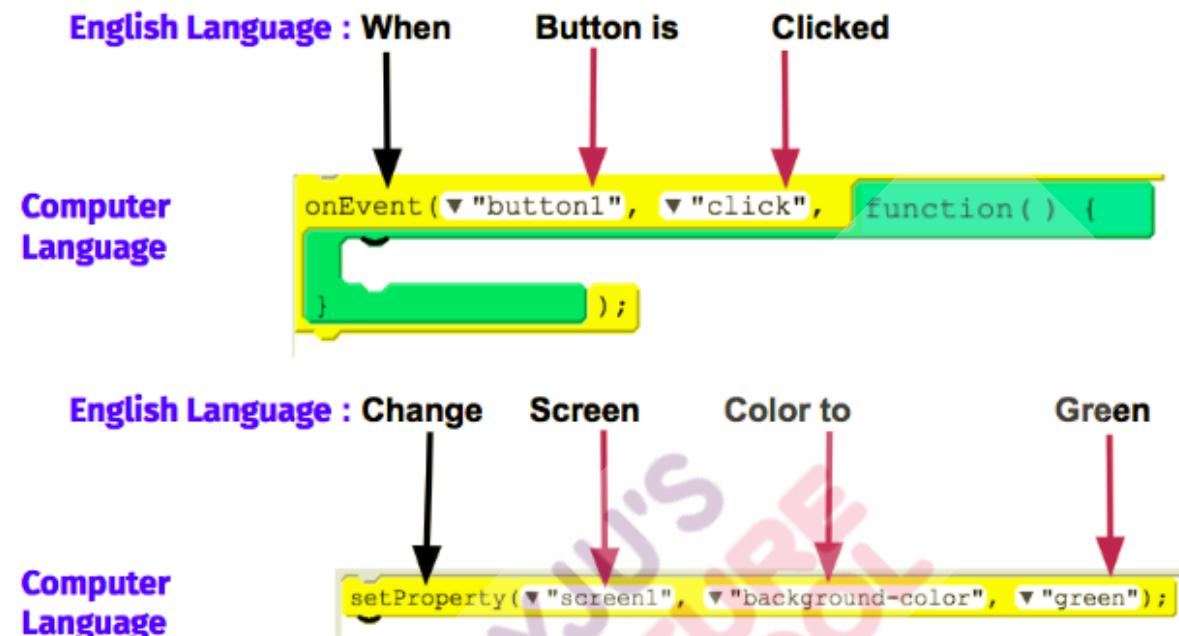
- A: When I click the button  
 B: Color of the screen should change to green.

For a computer to understand it, we need to translate the purpose of the app into computer code.

**“When”** in english translates into **“OnEvent”** in coding language.

**“Change”** in english translates into **“SetProperty”** in coding language.

## Code the app: English to Computer Language



So, we have dragged exactly these two code blocks.

Now, let's set proper values in the blocks.



```
onEvent(▼ "button1", ▼ "click", function(event) {
   setProperty(▼ "screen1", ▼ "background-color", ▼ "green");
});
```

	<p>“When I click the button”          Now reads as  <b>“onEvent, button1 is click”</b></p> <p>“Change color of the screen to green”          now reads as  <b>“setProperty to screen1          background-color to green”</b></p>	<p>Change the values of parameters from the drop-down list as shown in the screenshots below.</p>
	<p>We are done with Step 3 which is the coding part.</p> <p>Now we will do the fourth and the final step - Run the App.</p> <p><b>Step 4: Run the App</b>          Now, I will click the <b>BUTTON</b> on screen to see if it works.</p>	<p>Click on </p> <p>Press the <b>Button</b> on the mobile screen to see the code working.</p>

Code Design

screen1

Button

Run

4.Click Run and Click the Button

Toolbox

UI controls Data Control Variables Canvas Turtle Math Functions

onEvent(id, type, callback)  
button(id, text)  
textInput(id, text)  
textLabel(id, text)  
dropdown(id, option1, etc) ~  
getText(id)  
setText(id, text)  
getNumber(id)  
setNumber(id, number)  
checkbox(id, checked)  
radioButton(id, checked) ~  
getChecked(id)  
setChecked(id, checked)  
image(id, url)  
getImageURL(id)  
setImageURL(id, url)

Workspace:

```
1 onEvent("button1", "click", function(event){  
2     setProperty("screen1", "background-color", "green");  
3 })  
4
```

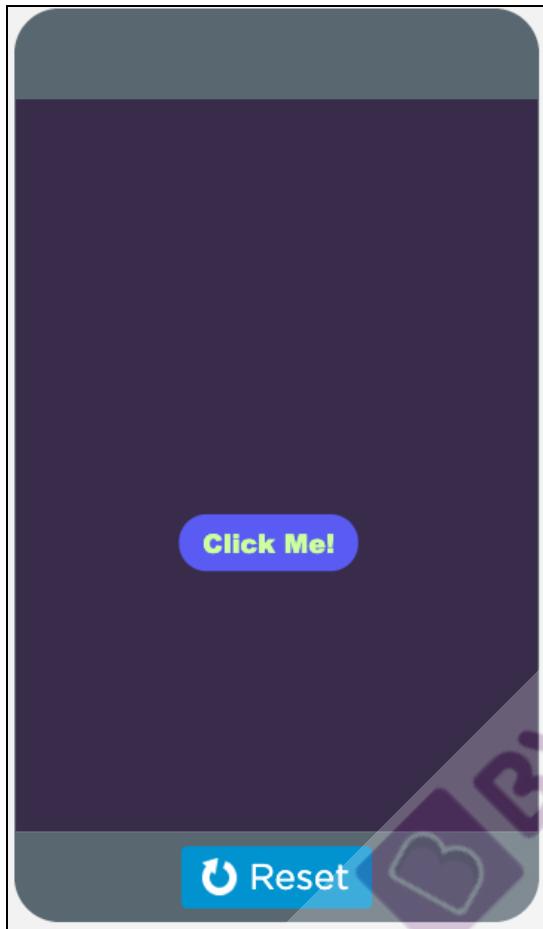
Version History

Output:

© 2022 - BYJU'S Future School.

Note: This document is the original copyright of BYJU'S Future School.

Please don't share, download or copy this file without permission.



BYJU'S  
FUTURE  
SCHOOL

Wow! Did you see that the app works and it changes the screen color.

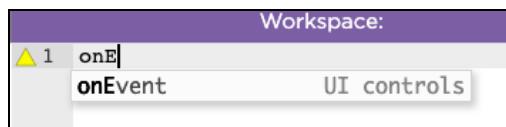
Now let me delete all the blocks and write the code in text. Let's see how it works.

You see as I am typing, the code shows me options. This feature is

Drag all the blocks back to **ToolBox** to clear the workspace and click on  on the top right corner to switch to text mode.

Type the same code.

called autocomplete. This helps programmers to program faster. You can press the down arrow key to choose from the options.



Since textmode is more strict than block mode you have to follow the rules of the javascript language.

See here - the yellow triangles indicate a warning. Here it is showing the warning because we did not put a semicolon.

**A semicolon tells the computer that the code line ends here as the computer executes the code line by line.**

Exactly like how a period or a full stop indicates the end of a sentence in english language, similarly even computer languages have grammar rules which are called Syntax.

We will learn more of this in detail as the course progresses.

Okay now lets run and see if the app works.

```
onEvent("button1", "click",
function() {
    setProperty("screen1",
    "background-color",
    "green");
});
```

Point to the Yellow triangle.

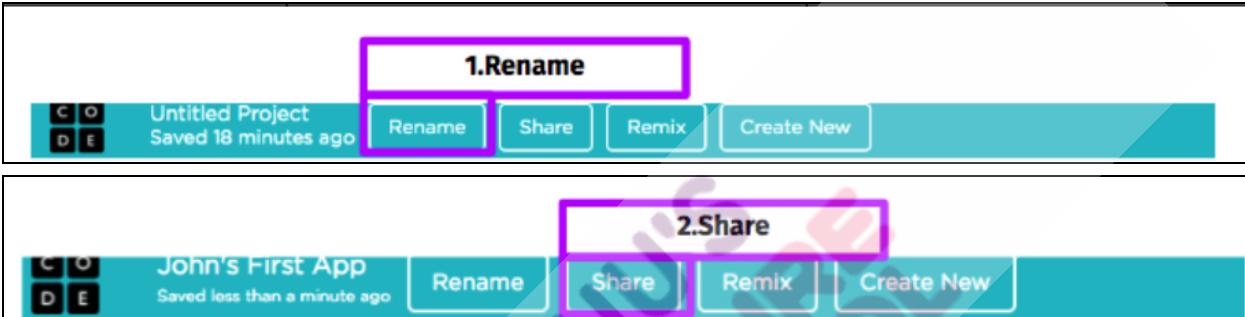
Click **Run** and press **Button** on the mobile screen to see the code working.

	<p>Did it work? What colour did the mobile screen turn into? Green? Yes! It worked.</p> <pre>1 - onEvent("button1", "click", function(){ 2     setProperty("screen1", "background-color", "green"); 3 }) </pre>	
	<p>Let me explain the code to you again.</p> <p>In coding “Function” means “do this”. It is a way of telling the computer to do something.</p> <p>So  <b>OnEvent</b> means When  <b>Function</b> means Do this  <b>setProperty</b> means Change.</p> <p>This code in text form reads exactly what our purpose is -  <b>When</b> button1 is clicked <b>Do This.</b>  <b>Change</b> screen1 background-color to green.</p>	
<b>Teacher Stops Screen Share</b>		
	<p>Now, please do all this at your end and change the colour of a screen to one of the following colors - Red, or Blue or Yellow.</p> <p>Key points to remember:</p> <p><b>What was our PURPOSE?</b></p>	

	<p>When the button is clicked, change screen colour.</p> <p><b>How to DESIGN?</b> Drag and drop the button on screen</p> <p><b>What is our CODE?:</b> When <b>button1</b> is <b>clicked</b>; change <b>screen1 background-color</b> to <b>COLOR</b></p> <p>Now reads as: <b>"onEvent, button1 click"</b></p> <p><b>"setProperty, screen1 background-color green"</b></p> <p>You did really well and you deserve a <b>big round of applause!</b></p> <p>Finally <b>Run</b> to see it work</p> <p>Now it's your turn.</p>	<p>On the top left corner, press the  icon It will open up a tray of emojis</p>  <p>Press the Clap Emoji</p> <ul style="list-style-type: none"> <li>Ask the student to press <b>ESC</b> key to come back to panel</li> <li>Guide the student to start Screen Share</li> <li>Teacher gets into Fullscreen</li> </ul>
--	---	--

<b>Step 3:</b> <b>Student led</b> <b>Activity</b> (15 minutes)	<p>In the Student Panel, click the link for Activity 1.</p> <p>Use the email and password I shared with you in the chat window to login.</p> <p>Also open <b>Student Activity 2</b> for Reference.</p> <p><b>I want you to code in block mode for now as this is your first class and as we progress we will learn to code in text.</b></p>	<p><a href="#"><u>Student Activity 1 - APP LAB</u></a></p> <p>It will prompt for sign-in</p>  <p>Copy Paste the Login and Password in the chat window.</p> <p><a href="#"><u>Student Activity 2 - CODE DIAGRAM</u></a></p> <p><b>Let the student code in Block mode only.</b></p> <p>As the student does the activity, use encouraging remarks like “Great”, “You are doing good” “Good Job”.</p> <p>Student has to repeat all <b>4 steps</b> and do the entire activity with their selected colour</p> <p>Guide the student as required using the steps above.</p>
	<p>Great Job! Now that you have built an app, give it a name and Share it.</p> <p>I want you to come back to the BYJU'S Future School Screen now!</p> <p><b>Note to teachers:</b> Make sure that the kid is on BYJU'S Future School screen during the Emoji reward!</p>	<p>On the top left corner, press  the  icon</p> <p>It will open up a tray of emojis</p>  <p>Press the Clap Emoji</p>

	<p>You did really well and you deserve a <b>big round of applause</b>.</p> <p>Click <b>Rename</b> and name your project. Give it a special name as this is the first app that you have created.</p> <p>Click <b>Share</b> to share your project.</p>	
--	--	--



**1.Rename**

Untitled Project  
Saved 18 minutes ago

**Rename** Share Remix Create New

**2.Share**

John's First App  
Saved less than a minute ago

Rename **Share** Remix Create New

	<p>Copy the project link and paste it in the <b>Project Link Field</b> on the student panel and click <b>Submit Button</b>.</p> <p><b>This app is not downloaded. To convert this app into a downloadable app we need to perform a few more steps which you will learn later in this course.</b></p> <p><b>However, the link of your app will be shared with your parents via email.</b></p> <p><b>Don't forget to open the link on any mobile to see how it appears.</b></p> <p>Excellent! Your app is working.</p> <p>I want you to come back to the BYJU'S</p>	
--	---	--

On the top left corner, press the  icon

	<p>Future School Screen now!</p> <p><b>Note to teachers:</b> Make sure that the kid is on the BYJU'S Future School screen during the Emoji reward!</p> <p>You deserve a chocolate for doing so well</p>	<p>It will open up a tray of emojis</p>  <p>Press the Chocolate Emoji</p>
<b>Call parents, introduce yourself and celebrate the kid's accomplishment</b>		
<p><b>Teacher-</b> "Hi, I feel proud to say that today, &lt;child's name&gt; created his/her first app. &lt;child's name&gt;, please show the app you created today."</p> <ul style="list-style-type: none"> <li>• Students show what they created and explain the concepts learned to the parents.</li> <li>• Appreciate and compliment the student for trying and making strong progress to learn coding.</li> <li>• Get to know how they liked the session.</li> <li>• Review and check their understanding.</li> </ul>		
<b>Wrap Up</b> (10 minutes)	<p>So, how did you like the class today?</p> <p><b>Q</b> Can you tell me what are the 4 steps to build an app successfully?</p> <p><b>A</b></p> <ol style="list-style-type: none"> <li>1. Define the purpose of the app.</li> <li>2. Design the app.</li> <li>3. Code the app.</li> <li>4. Run the app to see it working.</li> </ol> <p>You did really well and you deserve a <b>big round of applause!</b></p> <p>You completed all the activities with great creativity and you deserve a <b>Hats Off</b> which gives you 10 Points.</p>	<p>On the top left corner, press the  icon</p>

	<p>You also get a Hats Off for being well behaved in class and asking/answering some great question which gives you another 10 Points.</p> <p>You did really well in class today doing all the puzzle challenges with great concentration. So you get another Hats Off which gives you 10 more Points.</p> <p>Congratulations! You are among the exceptional students who get 3 out of 3 hats off in the trial class!</p> <p>You are really bright and I hope you join the full course where you'll create a full entrepreneur-ready app that thousands across the world may download.</p> <p>Now, Let me give you a quick summary of what we will be learning in the course.  <a href="#">Please click on Student Activity 3</a></p>	<p>It will open up a tray of emojis</p>  <p>Press the Clap Emoji</p> <p>Press the Hats Off Icon for <b><i>Great Persistence</i></b></p>  <p>Press the Hats Off Icon for <b><i>Great Question</i></b></p>  <p>Press the Hats Off Icon for <b><i>Strong Concentration..</i></b></p>  <p><b><a href="#">Student Activity 3 - Curriculum Overview</a></b>  <b><a href="#">START THE SLIDE SHOW</a></b></p>
	<p>To provide you with more details about the course as well as answer any questions that you may have our academic counselor will get in touch with you. <b>If they are not able to reach out to you, do you have a preferred time?</b> My schedule is almost full but</p>	

	<p>I'd love to have your kid as my student since your kid is exceptionally bright with true entrepreneurship potential!</p> <p>Congratulations &lt;kid name&gt; again on being awarded 3 hats off! You are truly exceptional. Hope you had fun.</p> <p>Thank you for your time today. Kindly stay on the panel and do not close this page when I end the class--our entire curriculum along with the details will be displayed on the panel.</p>	<p><b>ADVANCED Curriculum Vision</b> is provided below for Teacher reference if required</p> <p><a href="#"><u>Student Activity 4-COURSE PDF</u></a></p> <p>Ask the parent to refer to this link to know more about the <b>ADVANCED</b> curriculum.</p>
--	--	---

## Course Objectives



# ADVANCED

Grade 4-Grade 5-Grade 6

 <b>JAVASCRIPT</b> <b>TRACK</b> <b>144 Classes</b>	<span style="font-size: 2em; color: #4CAF50;">+</span>	 <b>PYTHON 3</b> <b>TRACK</b> <b>156 Classes</b>
--	--	--

The Advanced Curriculum is World's First Comprehensive Coding Curriculum Crafted for kids Age 9+ to prepare them for Future Digital Leadership. The Curriculum is focused on:

- 1. **Introducing** Fundamental Coding Concepts to kids through Block Based Coding with Javascript.
- 2. **Mastering** Web Development Through Syntax Programming in CSS HTML and Javascript
- 3. **Deepening** Algorithmic Thinking with Applied Space Technology and Arduino Programming with C++
- 4. **Specialising** in 29+ Industry Grade Technologies + Platforms like Artificial Intelligence, NLP, Realtime Database, Machine Learning
- 5. **Inventing** the Digital Future by stretching kid's intelligence by having them work at frontiers of Technologies like Internet of Things(IoT), Blockchain, Self-Driving Cars, Quantum Computing.

## Course **Modules**



# ADVANCED

Grade 4-Grade 5-Grade 6



**JAVASCRIPT**  
TRACK  
**144**



**PYTHON 3**  
TRACK  
**156**

The Advanced Course is further divided into following 21 Major Modules Leading to highly advanced Tech Products covering entire Spectrum of Computer Technology

### **JAVASCRIPT TRACK** **144**

1. Coding Basics
2. Procedural Programming
3. Native App Development
4. Functional Programming
5. AI Chatbots
6. Cloud Apps|Games
7. Web Development
8. UI/UX Design
9. Space Tech
10. Natural Language Processing
11. Neural Networks
12. Deep Learning
13. Machine Learning
14. AI Game

### **PYTHON TRACK** **156**

15. Python Applications
16. Internet of Things(IoT)
17. Big Data
18. Programming Self-Driving Cars
19. Blockchain
20. Cloud Computing
21. Quantum Computing

## Learning **Outcomes**



# ADVANCED

Grade 4-Grade 5-Grade 6



**JAVASCRIPT**  
TRACK  
**144**



**PYTHON 3**  
TRACK  
**156**

The Advanced Curriculum is World's First Comprehensive Coding Curriculum Crafted for kids Age 9+ to prepare them for Digital Future, Teaching kids to develop products like:

### **JAVASCRIPT TRACK** **144**

- Web Apps
- Native Apps
- Cloud Games
- Space Simulations
- CAD 3D Models
- Advanced Websites
- Arduino Circuits
- AI Chatbots
- UI/UX Design
- Cloud Apps
- Cloud Databases
- Artificial Intelligence

### **PYTHON TRACK** **156**

- Python Application
- Machine Learning
- Self-Driving Cars Logic
- Cryptocurrency
- BlockChains
- Smart Contracts
- Data Visualisation
- Cloud Repositories
- Cloud Functions
- Internet of Things
- Smart Devices
- Quantum Circuits
- Quantum Programs

## Course Summary



# ADVANCED

Grade 4-Grade 5-Grade 6



**JAVASCRIPT**  
TRACK  
144



**PYTHON 3**  
TRACK  
156

### 48|FOUNDATION [ Block Based Coding ]

Kids start with learning the basics of Coding while developing Animations , Stories, Games, WebApps, Native Apps, Chatbots.

### 49-96|EXPLORATION [ Block+Text Based Coding ]

Kids learn Web Programming While Building Websites and Sophisticated Industry Grade UI/UX Space Tech Apps thus Stretching their Imagination to comprehend and Solve Complex Space Problems with Simulations and Robotic Rovers

### 97-144|SPECIALISATION [ Text Based +Frameworks ]

Kids Specialise in Artificial Intelligence Concepts and their applications through building smart Apps and Websites. This prepares them to take on the future with confidence in their creativity.

### 145-300|INVENTION [ Libraries+Frameworks ]

Kids learn to work right at the frontiers of future technologies like Self Driving Cars, Cryptocurrencies, Smart Contracts, Internet of Things, Quantum Computing transforming them from experimenter to inventor digital world of the future.

## 8|BASIC



# ADVANCED

Grade 4-Grade 5-Grade 6



**JAVASCRIPT**  
TRACK  
144

### MODULES

- Coding Basics

### OUTCOMES

- Web App
- Art
- Animation
- Stories
- Games

## 48|STANDARD



# ADVANCED

Grade 4-Grade 5-Grade 6



### MODULES

- Procedural Programming Concepts
- Native App Development
- Functional Programming Concepts
- AI Chatbots
- Cloud Apps|Games

### OUTCOMES

- Web Apps|Native Apps|AI Apps
- Communication Apps eg.Chat App
- Health Apps eg.Heath Calculator
- Utility Apps eg.Password Generator
- AI Chatbots

## 144|PREMIUM



# ADVANCED

Grade 4-Grade 5-Grade 6



### MODULES

- Front End Development|Bootstrap
- UI/UX Design
- Space Tech
  - Space Simulations
  - Space CAD 3D Design
  - Space Data Analysis
  - Space Robotics
- NLP
- Neural Networks
- Deep Learning
- Machine Learning
- AI Game

### OUTCOMES

- Advanced Websites
- Advanced Blog
- 3D Models
- Arduino Circuits & Programming
- Space Simulations
- e-Portfolio Website
- AI Board Game
- Space Data Visualisation

# 300|FRONTIERS



## ADVANCED

Grade 4-Grade 5-Grade 6



### MODULES

- Python Applications
- Internet of Things(IoT)
- Big Data
- Programming Self-Driving Cars
- Blockchain
- Cloud Computing
- Quantum Computing

### OUTCOMES

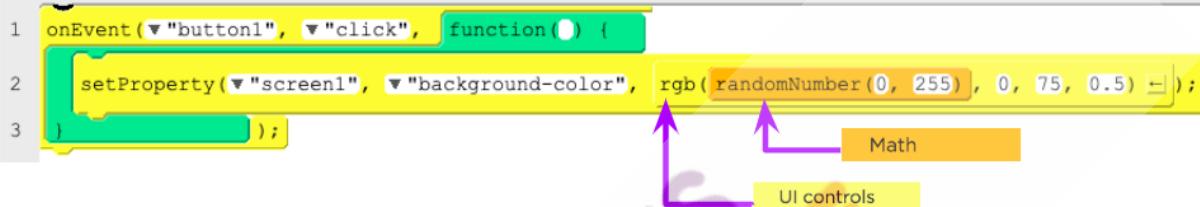
- Application Softwares
- Autonomous Car Simulations
- Advanced Data Visualisations
- Arduino Smart Devices
- Smart Homes
- Cryptocurrency
- Smart Contract
- Cloud Applications
- Quantum Programs

**x End Class**

**Teacher Clicks**

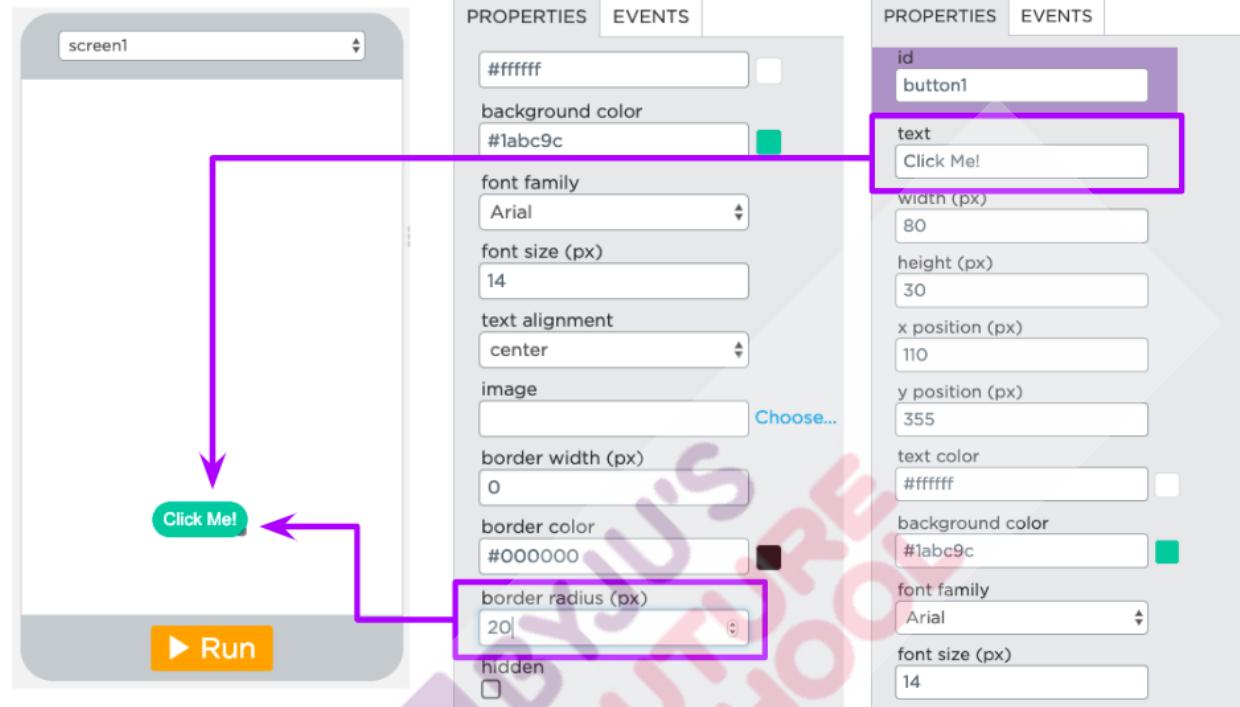
<b>Additional Activity 1</b>	<p>Okay Great!</p> <p>We could successfully change the color of the screen when the button is pressed.</p> <p>What if I want to change the screen color to any random color every time the button is pressed.</p> <p>I will guide you to do the same.</p> <p>1. Go to <b>UI controls</b> and Drag <b>rgb(r, g, b, a) ←</b> block and drop it at the place</p>	<p>Ask the student to follow your instructions as the student makes the following changes.</p>
------------------------------	---	--

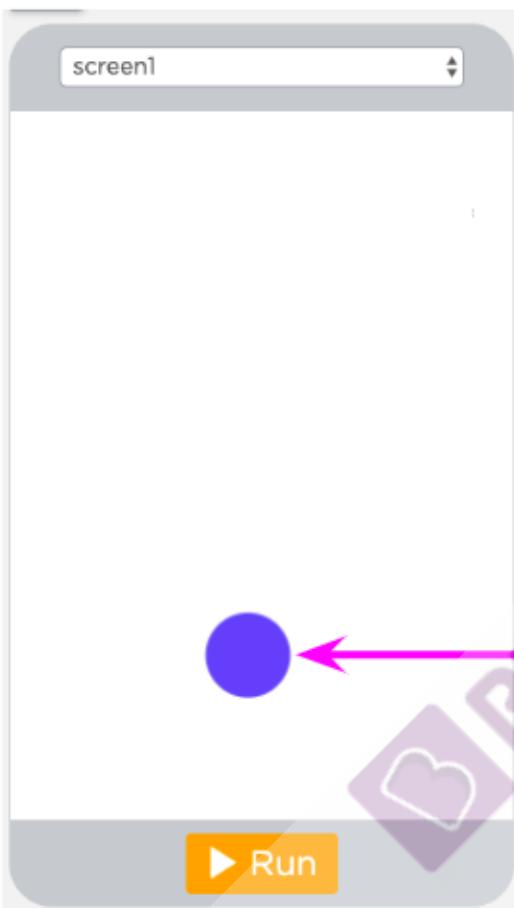
	<p>where you selected color from in <b>set Property</b> block.</p> <p>2. Now go to <b>Math</b> and drag <b>randomNumber(0, 255)</b> and drop it at <b>r</b> in the <b>rgb</b> block.</p>	
--	--	--



	<p>Let me explain what we did.</p> <ol style="list-style-type: none"> <li>1. The <b>rgb</b> block stands for a mix of RED+GREEN+BLUE shades. The values we put are shades of red green blue. The last number indicates transparency.</li> <li>2. The <b>randomNumber</b> lets the computer select any number within the range you provided.</li> </ol> <p>In computer every color has 256 shades so here what we did is we let the computer choose any shade of red from 0 to 255 using <b>randomNumber</b> block so we get random shade of RED everytime we click the button.</p> <ol style="list-style-type: none"> <li>3. We are keeping shades of green and blue the same</li> </ol>	
--	--	--

	<p>default values and also the transparency is set default 0.5.</p> <p>4. Now let's click Run and click the button as many times as possible to see what happens.</p>	<p>Click on</p> 
Additional Activity 2	<p>You are doing great!</p> <p>To make the button look great in your app let's make some design modifications to the button.</p> <p>Go back to <b>Design</b> Mode and click on the button to see its properties.</p> <p>Make the following changes to the button properties.</p> <p>text Click Me!</p> <p>1. Change the button text to <b>Click Me!</b> to make it appealing. This will make the user curious to press it to see what happens when the button is pressed.</p> <p>border radius (px) 20</p> <p>2. Change the border radius to 20. This will give your button a circular look to make it look more advanced when you open the app on mobile phone. OR You can keep the button square but make sure radius should be half of the size of the button.</p>	<p>Ask the student to follow your instructions as the student makes the following changes.</p>

	<p>3. Now click on <b>Run</b> and press <b>Button</b> on mobile screen to see the code working.</p>	<span>Click</span> 
		
	<p>You can also make the button completely circular by changing the border radius to 100 and removing the button text. Make sure you set the button dimension as a square.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> width (px) 40 height (px) 40 </div>	



PROPERTIES EVENTS

id	button1
text	
width (px)	40
height (px)	40
background color	#536cea
font family	Arial Black
font size (px)	15
text alignment	center
image	<input type="button" value="Choose"/>
border width (px)	0
border color	#4d575f
border radius (px)	100

Additional Activity 3	<p>Great!</p> <p>Now let's say we want to enhance the features of this app.</p> <p>Let's say I want to change the color of the screen to any random every second.</p> <ol style="list-style-type: none"> <li>1. First go to Code and locate <b>Control</b> Section.</li> </ol>	
-----------------------	--	--

2. Lets drag the

`timedLoop(ms, callback)`

function.

This code block will execute all the code blocks we insert into it with some time interval. This is a kind of a forever loop function which runs forever.

3. So drag the block and let's drop it above the set property block.

```

1 onEvent("button1", "click", function() {
2     timedLoop(1000, function() {
3         // ...
4     });
5     setProperty("screen1", "background-color", rgb(randomNumber(0, 255), 0, 75, 0.5));
6 });
7

```

Now let's drag the **set Property** block inside the **timed loop** block.

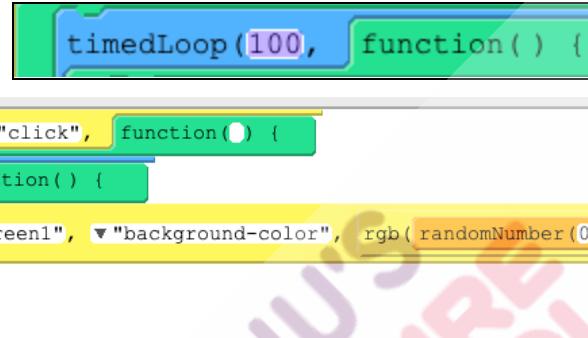
```

1 onEvent("button1", "click", function() {
2     timedLoop(1000, function() {
3         setProperty("screen1", "background-color", rgb(randomNumber(0, 255), 0, 75, 0.5));
4     });
5 });
6

```

When the button is clicked, the **timed loop** block will loop forever or repeat forever what is inside this block.

Since we have the **set Property** block inside the **timed loop** block the screen

	<p>color will change randomly every 1000 milliseconds or 1 second as you know 1 second = 1000 milliseconds.</p> <p>You can change the time interval and make it smaller to have rapid color change effect. Let's try and set it to 100 milliseconds. See what happens.</p>	
	 <pre> 1 onEvent("button1", "click", function() { 2     timedLoop(100, function() { 3         setProperty("screen1", "background-color", rgb(randomNumber(0, 255), 0, 75, 0.5)); 4     }); 5 }) </pre>	
<b>Additional Activity 4</b>	<p>Now let's add some animation to this App. What if I want the computer to draw random shapes on the screen by itself as the background color of the screen changes?</p>	
<ol style="list-style-type: none"> <li>1. So let's go to  and add turtle to our app inside the timed loop. This will make the turtle draw randomly on the screen. Turtle is used to draw on the screen.</li> <li>2. Now add the following code in the <b>timed loop</b> function. So every 100 milliseconds or 0.1 seconds or 1/10th of second these 5 blocks will get executed.</li> </ol>		

```
2  timedLoop(100, function() {  
3      setProperty("screen1", "background"  
4          penColor("red");  
5          penWidth(3);  
6          moveForward(25);  
7          turnRight(randomNumber(1, 50));  
8      });  
});
```

- **set Property** block to change the screen color to random shades of red, green, blue.
- Turtle **pencolor** block to set the turtle pen color to red. You can change it to any color.
- Turtle Pen or brush width is set to 3 pixels. You can change this as well.
- Turtle will move forward by 25 pixels: Pixels are little dots of color any computer screen or mobile screen is made up of.
- Turtle will turn right by a random angle between 1 degree to 50 degree. you can play around with this range too.

Also since we are keeping the color and width constant, we can take it out of the timed loop to keep it. That's how we make our code accurate. As we want the turtle to move forward every 100 milliseconds and turn right between 1 and 50 every milliseconds.

### Modified Code:

```

1 onEvent("button1", "click", function() {
2   penColor("red");
3   penWidth(3);
4   timedLoop(100, function() {
5     setProperty("screen1", "background-color",
6     moveForward(25);
7     turnRight(randomNumber(1, 50));
8   });
9 });

```

3. As the Turtle moves on the screen, let's hide other elements so that we can see what shape the turtle is able to draw.

Go to **UI controls** and use **hideElement Function**

```
9 hideElement("button1");
```

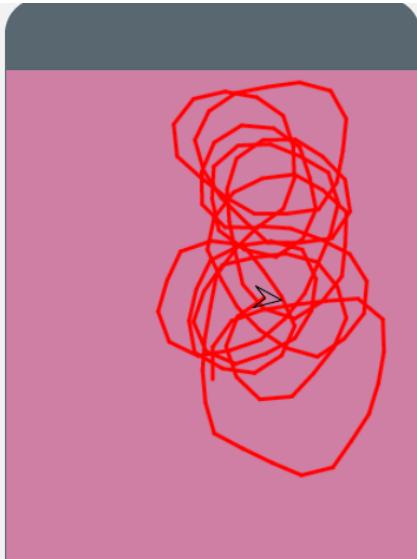
### COMPLETE CODE:

```

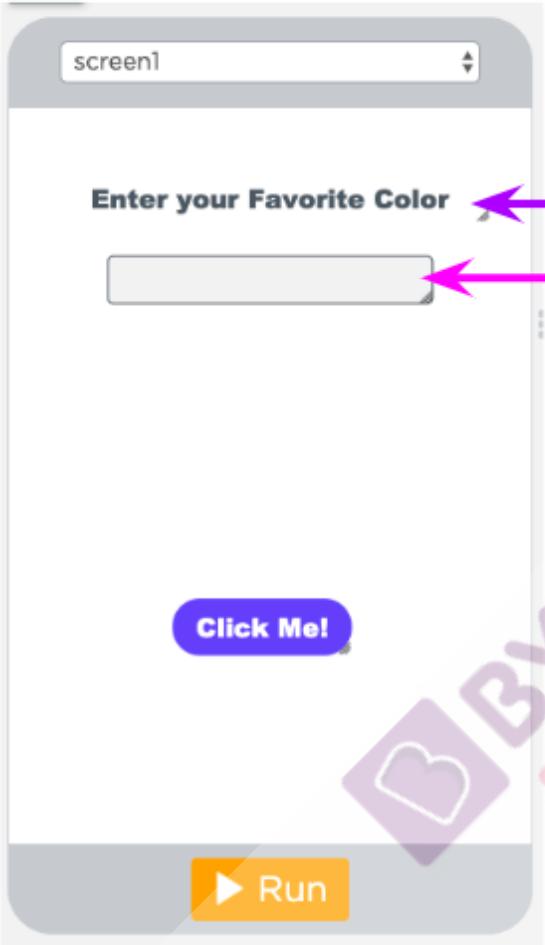
1 onEvent("button1", "click", function() {
2   penColor("red");
3   penWidth(3);
4   timedLoop(100, function() {
5     setProperty("screen1", "background-color", rgb(randomNumber(0, 255), 0, 75, 0.5));
6     moveForward(25);
7     turnRight(randomNumber(1, 50));
8   });
9   hideElement("button1");
10 });

```

**Wow! You just taught a computer to draw by itself through Code.**



Additional Activity 5	<p>Now let's extend the app to make it more interactive.</p> <p>Let's set the screen color to the app user's favorite screen color. So we will need a text input box where the user can type his/her favorite color</p> <p>We will also have to add another button. This button will be clicked to change the background colour according to the input from the person.</p> <p>Let's go to Design mode and add a <b>textinput</b> box to get the user input and a button.</p> <p>Drag a <b>label</b> and a <b>textinput</b> box. The label is used to display some text on the app and the textinput box is used to type in text.</p>	
-----------------------	---	--



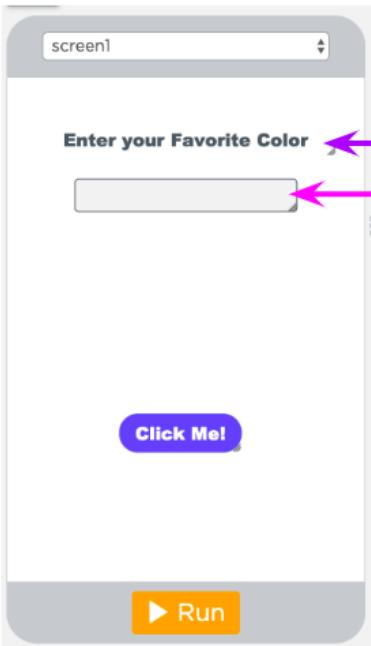
The screenshot shows a mobile application interface titled "screen1". It contains a text input field with the placeholder "Enter your Favorite Color" and a label "abc" below it. A "Click Me!" button is at the bottom left, and a "Run" button is at the bottom right. A large watermark "BYJU'S FUTURE SCHOOL" is diagonally across the screen.

### Design Toolbox

Drag the elements into your app!

 Button	 Text Input
 Label	 Dropdown
 Radio Button	 Checkbox
 Image	 Canvas
 Screen	 Text Area
	

	Now let's set the properties of label1 and textinput1.	
--	--	--



Properties for 'label':

id	label
text	Enter your Favorite Color
width (px)	260
height (px)	25
x position (px)	35
y position (px)	45
text color	#4d575f
background color	rgba(0, 0, 0, 0)
font family	Arial Black
font size (px)	13

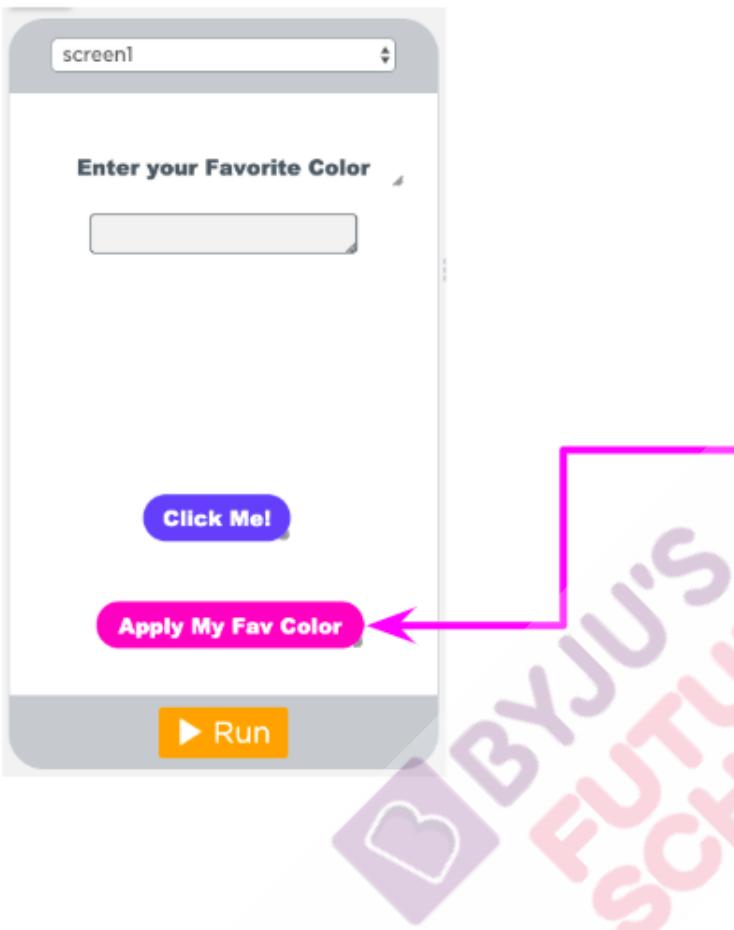
Properties for 'text\_input1':

id	text_input1
placeholder	
width (px)	200
height (px)	30
x position (px)	60
y position (px)	90
text color	#4d575f
background color	#f2f2f2
font family	Arial
font size (px)	13

Click on the **Click Me!** button and click on the **Duplicate** button on the right.

This will create another button exactly like the first one.

Now drag this new button and place it on the screen and set its properties as shown below.



PROPERTIES EVENTS

id	button2
text	Apply My Fav Color
width (px)	200
height (px)	35
x position (px)	65
y position (px)	380
text color	#ffffff
background color	#ea53be
font family	Arial Black
font size (px)	15

Remember the id of this button is button2.

From **UI controls** add another **onEvent** block for this button and select button2.

```
onEvent("id", "click", function() {  
    "button1"  
    "button2"  
    "label1"  
    "screen1"  
    "text_input1"
```

Use the `getText(id)` block to get the text entered by the user and drop it in the color parameter.

```
onEvent("button2", "click", function() {  
    setProperty("screen1", "background-color", getText("id"));  
});
```

"button1"  
"button2"  
"label1"  
"screen1"  
"text\_input1"

```
onEvent("button2", "click", function() {  
    setProperty("screen1", "background-color", getText("text_input1"));  
});
```

## COMPLETE APP CODE

```

1 onEvent("button1", "click", function() {
2     penColor("red");
3     penWidth(3);
4     timedLoop(100, function() {
5         setProperty("screen1", "background-color", rgb(randomNumber(0, 255), 0, 75, 0.5));
6         moveForward(25);
7         turnRight(randomNumber(1, 50));
8     });
9     hideElement("label1");
10    hideElement("button1");
11    hideElement("text_input1");
12    hideElement("button2");
13 });
14 onEvent("button2", "click", function() {
15     setProperty("screen1", "background-color", getText("text_input1"));
16 });

```

So here we have added the **hideElement** function to button1 event, so that these elements hide making space for turtle to draw on the screen.

```

hideElement("label1");
hideElement("button1");
hideElement("text_input1");
hideElement("button2");

```

So when the button2 is clicked, the app will get the text from textinput block and change the color to the color entered by the user in the text input block.

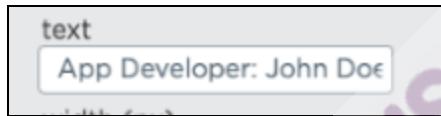
So if the user types “John” in the textinput block nothing will happen as “John” is not a color. Computer knows all the color names. So once it detects

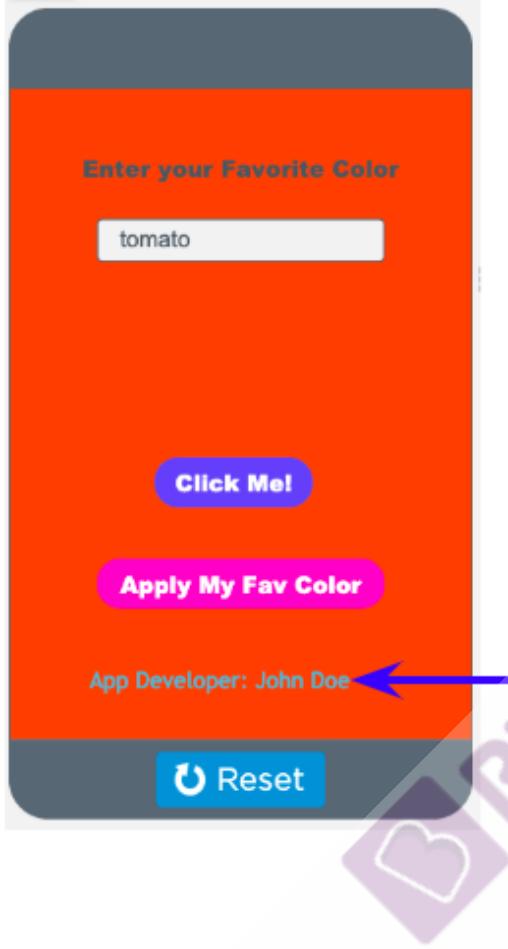
the color name it will apply it to the screen.

For example: light green, tomato, skyblue, brown, plum.

Great!

You can add your name to the app as a label, as this app is developed by you. So let's go to design and add your name in the Label2 text column.





The screenshot shows a mobile application interface. At the top, there is a header with the text "Enter your Favorite Color". Below it is a text input field containing the text "tomato". In the center, there is a blue button labeled "Click Me!". Below the button is another button labeled "Apply My Fav Color". At the bottom, there is a "Reset" button with a circular arrow icon. To the right of the application, a large red watermark with the text "BYJU'S FUTURE SCHOOL" is visible. A blue arrow points from the text "App Developer: John Doe" in the application to a properties panel on the right.

**PROPERTIES**   **EVENTS**

**id**: label2

**text**: App Developer: John Doe

**width (px)**: 250

**height (px)**: 20

**x position (px)**: 40

**y position (px)**: 400

**text color**: #72c2e2

**background color**: rgba(0, 0, 0, 0)

**font family**: Trebuchet

**font size (px)**: 13

Activity No.	Name of the Activity	Link
Teacher Reference	Did you know?	<a href="https://s3-cdnwhjr.whjr.online/18747c68-eb2e-471f-9f08-dfe88fa99b5f.html">https://s3-cdnwhjr.whjr.online/18747c68-eb2e-471f-9f08-dfe88fa99b5f.html</a>
Teacher Activity 1	APP LAB	<a href="https://studio.code.org/projects/applab/new">https://studio.code.org/projects/applab/new</a>
Teacher Activity 2	Blazing-OBBY	<a href="https://www.roblox.com/games/7351712185/Blazing-OBBY">https://www.roblox.com/games/7351712185/Blazing-OBBY</a>
Teacher Activity 3	Setup Roblox on your Computer	<a href="https://s3-whjr-curriculum-uploads.whjr.online/27cc2a7f-2b00-479f-9be5-59eb5850af6f.pdf">https://s3-whjr-curriculum-uploads.whjr.online/27cc2a7f-2b00-479f-9be5-59eb5850af6f.pdf</a>
Student Activity 1	APP LAB	<a href="https://studio.code.org/projects/applab/new">https://studio.code.org/projects/applab/new</a>
Student Activity 2	CODE DIAGRAM	<a href="https://docs.google.com/document/d/e/2PACX-1vQmBKkhzfPi1SjrhlfnfF4-5bdCsImRMUooGupScr84uFx9kV0OQiQriQ77kjUp6df_IGLZDTuSSyu/pub">https://docs.google.com/document/d/e/2PACX-1vQmBKkhzfPi1SjrhlfnfF4-5bdCsImRMUooGupScr84uFx9kV0OQiQriQ77kjUp6df_IGLZDTuSSyu/pub</a>
Student Activity 3	CURRICULUM OVERVIEW	<a href="https://s3-whjr-curriculum-uploads.whjr.online/4f588242-262b-41f0-bc45-ba204428a086.html">https://s3-whjr-curriculum-uploads.whjr.online/4f588242-262b-41f0-bc45-ba204428a086.html</a>
Student Activity 4	COURSE PDF	<a href="https://s3-whjr-curriculum-uploads.whjr.online/fade702a-5c60-4656-8064-0160afdb5435.pdf">https://s3-whjr-curriculum-uploads.whjr.online/fade702a-5c60-4656-8064-0160afdb5435.pdf</a>
Student Activity 5	Blazing-OBBY	<a href="https://www.roblox.com/games/7351712185/Blazing-OBBY">https://www.roblox.com/games/7351712185/Blazing-OBBY</a>

### Glossary for Teacher Reference -

S. No.	Abbreviations / Technical terms	Explanation
1	UI	The user interface (UI) is the point of human-computer interaction and communication in a device. which we as an user use to see on the screen and use them for interacting with our system like buttons, text input, images, dropdowns etc.
2	UX	User Experience. It is the way of design for the end users to experience the app or website. It helps us to know what will be the user's experience, his point of view for our product here in this case may be our mobile application, mobile game etc.
3	Algorithms	It is a set of rules that must be followed when solving a particular problem
4	Binary	Binary means something close to dual or double. binary is a code of zeros and ones (computer programming) also known as base two.
5	Variables	A variable is a quantity that may change within the context of a mathematical problem or experiment.
6	Conditional Loop	Conditional loops are a way to repeat something while a certain condition is satisfied, or True.
7	Function	A function is a block of organized, reusable code that is used to perform a single, related action.
8	API	API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other
9	URL	URL stands for Uniform Resource Locator. A URL is nothing more than the address of a given unique resource on the Web
10	IDE	An integrated development environment (IDE) is software for building applications that combines common developer tools into a single graphical user interface (GUI).
11	AI	Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions.
12	ML	Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed
13	NLP	Natural Language Processing, or NLP for short, is broadly defined as the automatic manipulation of natural language, like speech and text, by software.
14	GUI	A graphical user interface (GUI) is a type of user interface through which users interact with electronic devices via visual indicator representations.
15	Database	A database is an organized collection of structured information, or data, typically

		stored electronically in a computer system
16	Native Apps	A native app, or native application, is a software application built in a specific programming language, for the specific device platform, either iOS or Android.
17	Github	GitHub is a code hosting platform for collaboration and version control. GitHub lets you (and others) work together on projects.
18	Capstone Class	These are revision classes placed after some classes such as C8, which has the intention to revise all the concepts which has been taught in previous classes
19	Report card	Report card is something which we use to provide after analysing the kid on various parameters like his creativity, Logic building, Concentration, Coding Proficiency.
20	Space tech	Space tech classes & projects are usually those classes in which the activity is based on the concepts and facts of space technology. Example: Sun Earth Simulation etc
21	Simulator	A simulation is an approximate imitation of the operation of a process or system; that represents its operation over time
22	APK	APK stands for Android Package Kit and it's the file format that Android uses for its apps; much like Windows EXE files.
23	Code.org	Code.org is a programming platform which is used to create interactive animations, games, mobile applications and many more within built labs which are available like Game lab, App lab, Sprite lab, Artist lab etc.
24	Arduino	Arduino is an open-source electronics platform based on easy-to-use hardware & software.
25	HTML	HTML is the standard markup language for creating Web pages. HTML describes the structure of a Web page
26	CSS	CSS stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
27	JS	JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else.
28	Backend	The "backend" refers to any part of a website or software program that users do not see. In programming terminology, the backend is the "data access layer,".
29	Python	Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991
30	Rover Circuits	We will be using Rover Circuits for circuit simulations to simulate component behavior & to use it to measure circuits and to evaluate them. New designs can be tested, evaluated & diagnosed without actually constructing the circuit or system.
31	Circuit	It is a technique where computer software simulates the behavior of an electronic

	Simulation	circuit or system.
32	Flappy Bird	This game uses the concepts which are taught in the previous classes, where we have to prevent the bird to hit any object to touch the ground
33	Bouncing Ball	"In this game we Integrate all coding concepts--Sequence, Variables, Events and Animations and create a bouncing ball game
34	Health App	Build a health app to calculate various parameters which help people track their health.
35	Chat App	Students learn to integrate industry-grade database service to their app, learning to code with a focus on privacy protection and UI/UX design, building towards creating their own chat app.
36	School Blog	Students work on building a responsive website by applying bootstrap classes to all the elements of the website.
37	Kwitter	Here we apply every concept of front end development to build a social website while integrating google firebase in the backend.
38	School App	Kids learn to make the e-portfolio website applying bootstrap classes to all the elements of the website thus developing advanced UX design Skills.
39	Rocket Launcher	Here we learn to create a 3D model of ISRO PSLV using the tinkercad website
40	Solar system	We learn to simulate the effect of astrophysical concepts of solar distance and to express it in terms of light minutes and light years and create a scale model of our solar system.
41	Sun-Earth System	Here we learn to simulate the effect of astrophysical concepts like Astronomical Unit( Solar Distance) and orbital speed on the Sun-Earth System through coding.
42	My Selfie App	In this app we will make a native application which converts speech to text, and take a selfie if we have said "take my selfie"
43	Image filter app	kids learn to import pretrained net models to perform real time image processing leading to better understanding of efficient models
44	AI-DJ App	In this we will learn , making an AI DJ Web App using this posenet model of ml5.js, ml5.js library, and p5.js library.
45	AI-Ninja Game	Apply all the concepts of machine learning for creating an AI Ninja Game and making it work on neural networks thus learning to apply AI in real life.
46	Self Driving Car	In this students will learn various attributes considered while programming a self driving car, incorporating the sensors which are used in self driving cars, applying advance logic for following the traffic rules, object detection, and making the car smart enough to drive on its own for conducting a self driving car simulation.





## How to ace a trial class



**Connectivity and Power**

- ▶ Internet speed 20 Mbps
- ▶ Laptop always charged
- ▶ Power backup for router



**Camera and Headsets**

- ▶ Camera on while teaching
- ▶ Use a separate headset with inbuilt microphone
- ▶ Microphone 3 inches away from mouth



**Class personalisation**

- ▶ Address students by their name
- ▶ Give examples related to their hobbies and interests
- ▶ Engage them in class



**Communicate better**

- ▶ Moderate Speed of speech
- ▶ Facial Expressions and hand gestures
- ▶ Modulate tone to stress on important things



**Use Table and Chair**

- ▶ Be seated on a chair
- ▶ Laptop 1.5 feet away on the table
- ▶ Laptop Camera level just above eyes



**Background and Ambience**

- ▶ Clean and light colored background
- ▶ Well lit room
- ▶ Light source in front