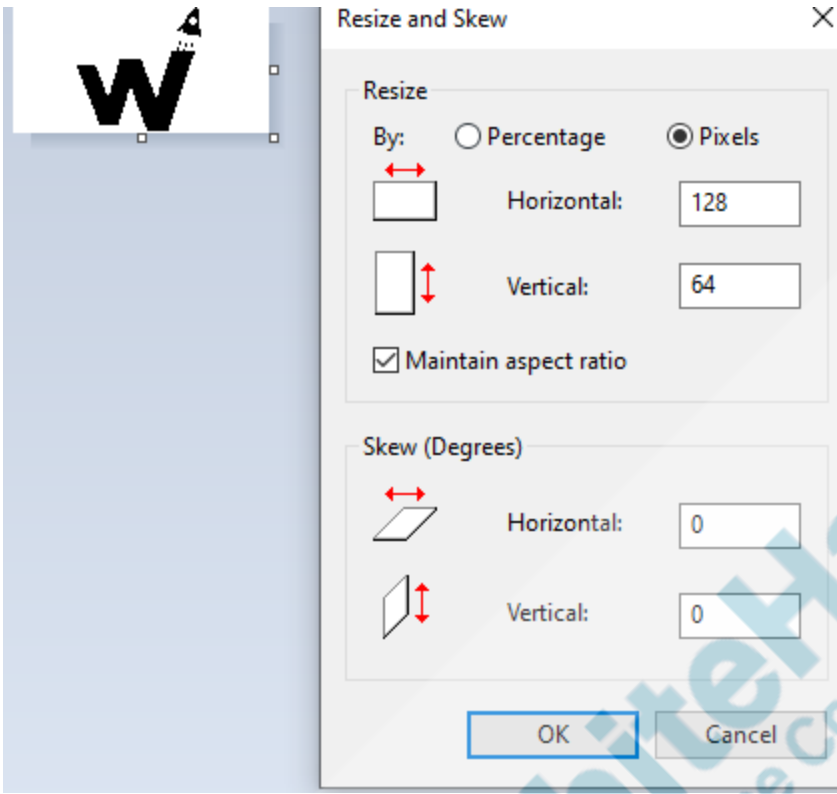


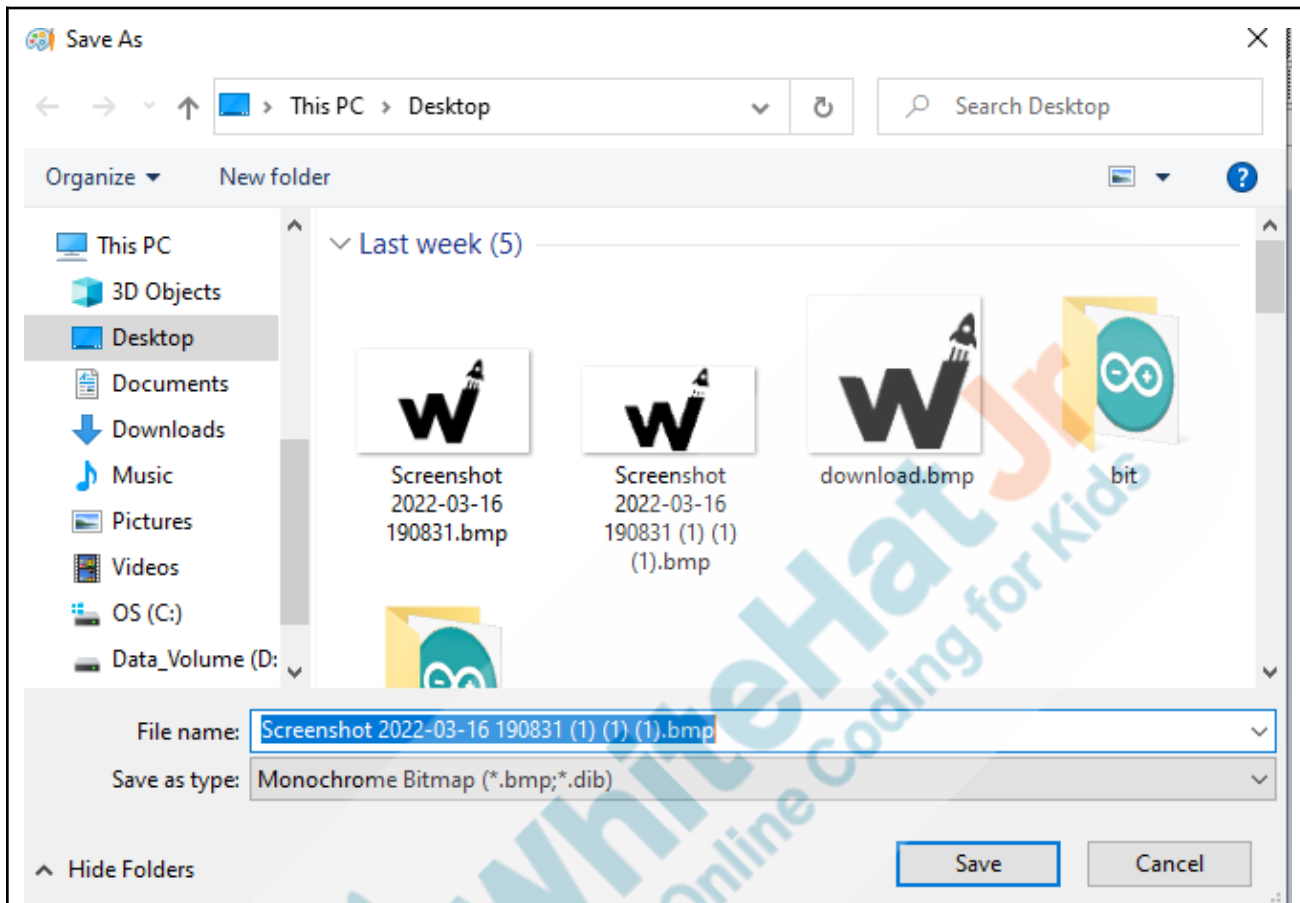
Topic	OLED BITMAP ARRAY	
Class Description	Students will be introduced to the concept of displaying logos and Bit Maps on OLED. They will learn how to convert images into Array and using Array convert them into the image.	
Class	PRO C256	
Class time	50 mins	
Goal	<ul style="list-style-type: none"> • Displaying WHJR Logo • Displaying Picture 	
Resources Required	<ul style="list-style-type: none"> • Teacher Resources: <ul style="list-style-type: none"> ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen ○ Smartphone • Student Resources: <ul style="list-style-type: none"> ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen 	
Class structure	Warm-Up Teacher-Led Activity -1 Teacher-Led Activity -2 Wrap-Up	10 mins 15 mins 15 mins 10 mins
Credit & Permissions:	Code samples used for Firebase-Google Authentication are licensed under the Apache 2.0 License . Expo documentation used from - https://expo.io Note: Keep this row section only if applicable	

WARM-UP SESSION - 10 mins	
Teacher Action	Student Action
<p>Hey <student's name>. How are you? It's great to see you! Are you excited to learn something new today?</p> <p>Following are the WARM-UP session deliverables:</p> <ul style="list-style-type: none"> Greet the student. Revision of previous class activities. Quizzes. 	<p>ESR: Hi, thanks! Yes, I am excited about it!</p> <p>Click on the slide show tab and present the slides</p>
WARM-UP QUIZ Click on In-Class Quiz	
<p>Activity Details</p> <p>Following are the session deliverables:</p> <ul style="list-style-type: none"> Appreciate the student. Narrate the story by using hand gestures and voice modulation methods to bring in more interest in students. 	
STUDENT-LED ACTIVITY-1 - 15mins	
Teacher Initiates Screen Share	
•	
Teacher Action	Student Action
<p>So, in the last class, we learned about OLED</p> <p>Any doubts from the last class,</p> <p><i>If the student has any doubts, clarify the doubts</i></p>	

<p>So last class we covered simple Shapes and text on OLED.</p> <p>Do you think it's possible to display images on the OLED?</p> <p>Yes, It is</p> <p>So in today's class we will display some graphics or pictures on OLED</p> <p>It's very simple!</p> <p>The only thing is we need to follow some steps to convert any graphics/Logo/Image/Picture onto OLED.</p> <p>But to make this happen we must understand about BITMAP</p> <p>BITMAP is an array of binary data representing the values of pixels in an image or display. A bitmap is a file format or memory organization of rows and columns of bits (or pixels) that collectively display a graphical representation. Most graphic images contain thousands of bits. Pixels are larger squares made up of bits.</p> <p>TYPES OF BITMAP FILE FORMATS</p> <ul style="list-style-type: none"> • BMP: Bitmap • GIF: Graphics Interchange Format • JPEG: Joint Photographic Expert Group • PNG: Portable Network Graphics <p>HOW DO BITMAPS WORK?</p>	<p>ESR: Varied!</p>
---	----------------------------

<p>Bitmaps contain data that organize thousands of squares by position and assigned color into a single, viewable graphic image.</p> <p>The resolution of the bitmap is the density of these squares. A resolution of 800×600 for a computer monitor means the screen offers 800 pixels for 600 lines, or 480,000 pixels collectively for the display. The higher the number of pixels, the clearer (or more realistic) a graphic or digital photograph is. Fewer pixels display a blurrier representation.</p> <p>So to display images/Graphics/logos on the screen we must perform some steps:</p>	
<p>We can draw a bitmap in this OLED display with some steps:</p> <ol style="list-style-type: none"> 1. Resize your image to fit the OLED display. 2. Convert image to monochrome 3. Convert monochrome image to array 4. Copy the array to Arduino code. 	
<p>Let's learn how we can convert images into arrays first.</p> <p>Let's try with Whitehatjr Logo.</p> <p>Resize the image to fit the OLED display</p> <p>Let's try with the Whitehatjr Logo that will be used to draw in the OLED display. The resolution of our OLED display is 128×64 pixels. So the size of the image is not more than that. It should be equal to or less than 128 x 64 pixels. We can resize an image using a paint tool.</p>	

	
<p>Convert the image to monochrome</p> <p>Save the image as a monochrome bitmap. Go to the File>>Click on Save AS>> Monochrome BitMap</p>	

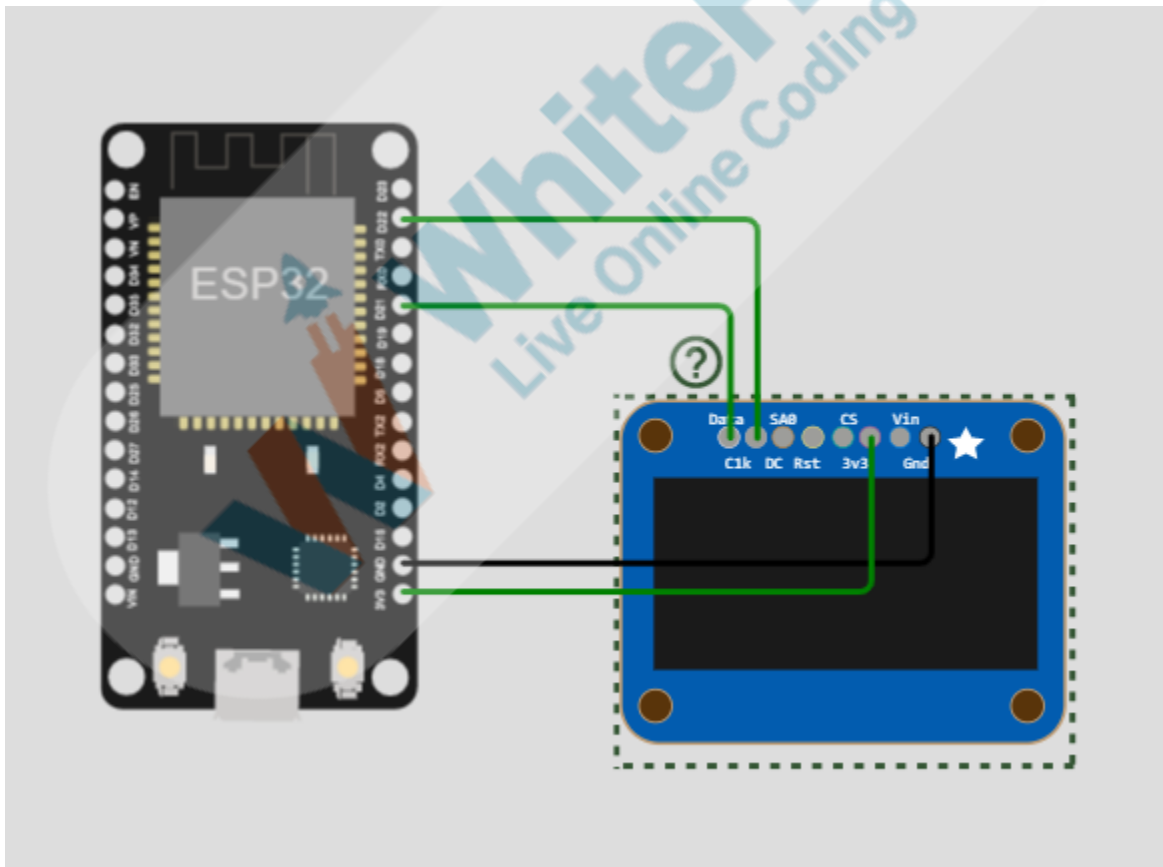


Convert monochrome image to array	
Teacher click on Teacher Activity 1	Student click on Student Activity 1
Upload the picture on the tool. Click on BitMap <input checked="" type="radio"/> Bitmap	
<div> <input type="text" value="Paste image or C/C++ here."/> <input type="button" value="Choose File"/> new1.jpg </div>	
Copy the entire content and save it, We have to use the	

[illegible]

- Insert OLED into the breadboard
- Take four jumper wires.

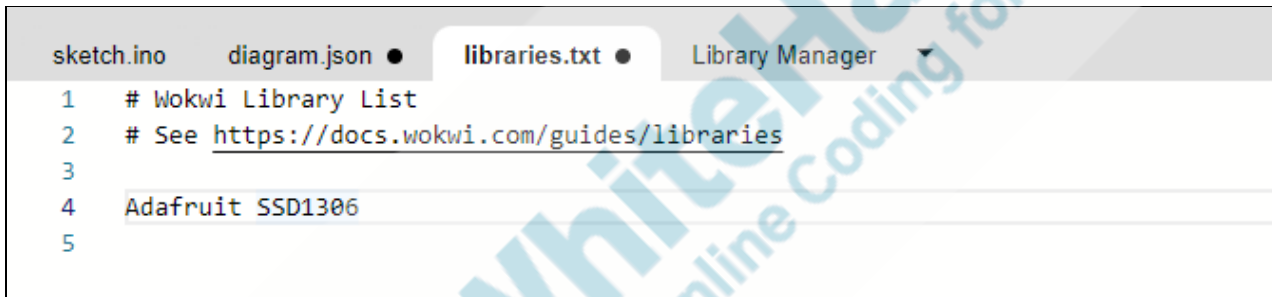
OLED PIN	ESP32 PIN
VCC	3.3 V
GND	GND
Data	GPIO 21
CLK	GPIO 22



To control the **OLED** display, we need to install libraries

1. Click on the small triangle icon ▼ next to Library Manager
2. Select New File
3. Name the file libraries.txt
4. Write down **Adafruit SSD1306**

Note: Follow the same step at student systems too. For reference, everything is mentioned again in the student activity.



```

sketch.ino  diagram.json  ●  libraries.txt  ●  Library Manager ▼
1  # Wokwi Library List
2  # See https://docs.wokwi.com/guides/libraries
3
4  Adafruit SSD1306
5

```

Import Libraries:

- **Wire.h** This library allows you to communicate with I2C / devices. I2C is a **serial communication protocol**, so data is transferred bit by bit along a single wire.
- **Adafruit_GFX.h:** This library offers a common graphical syntax and set of functions for all LCD displays, OLED displays, and LED matrices.
- **Adafruit_SSD1306:** This library takes care of low-level communication with the hardware.
- Define **SCREEN_WIDTH & SCREEN_HEIGHT** for OLED .Our **OLED** size is **128×64**.
- Declaration of an **SSD1306** display that connects to **I2C** communication using **Wire** Library.
- Initialize a **display** object with the **SCREEN_WIDTH**

<p>& SCREEN_HEIGHT defined earlier with the I2C communication protocol.</p> <ul style="list-style-type: none"> A value of (-1) indicates that our OLED display does not have a RESET pin. Sometimes OLED displays have a RESET pin on the OLED, in that case, we should connect it to a GPIO and should include the GPIO number as a parameter. 	
<pre>#include <Wire.h> #include <Adafruit_GFX.h> #include <Adafruit_SSD1306.h> #define SCREEN_WIDTH 128 #define SCREEN_HEIGHT 64 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);</pre>	
<p>The teacher/student must convert any picture into a BITMAP array at their end and use the same BITMAP array under this section.</p> <p>Note the name of the BITMAP array. To call it, we need to use the same name.</p> <p><i>Note: Copy the Bit Map array code and paste it into the below section</i></p> <p>The screenshot below is just for reference.</p>	

```
const unsigned char bitmap_14v9pq[] PROGMEM = {

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111
};
```

Initialize using **void setup()** function

- Serial.begin(115200): Sets the **data rate** in bits per second (baud) for **serial** data transmission.
- Initialize the OLED display with the begin() method.
- If the OLED displays nothing, check the OLED address at **0x3C**. In our case, the address is 0x3C.
- If we are not able to connect to the display, it prints a message on the Serial Monitor.
- If something fails, don't proceed further, try to repeat the process using **for()** loop
- display.drawBitmap is used to display the picture
Write down the exact name of the BitMAP array.
Here in this case it is **bitmap_14v9q**

Note: This is a reference picture code, the student will right his BITMAP array name

```
void setup() {  
  Serial.begin(115200);  
  
  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {  
    Serial.println(F("SSD1306 allocation failed"));  
    for (;;);  
  };  
  
  delay(2000);          // wait two seconds for initializing  
  display.clearDisplay(); // clear display  
  
  display.drawBitmap(0, 0, bitmap_l4v9pq, 128, 64, 1);  
  
  display.display();  
}
```

Output:

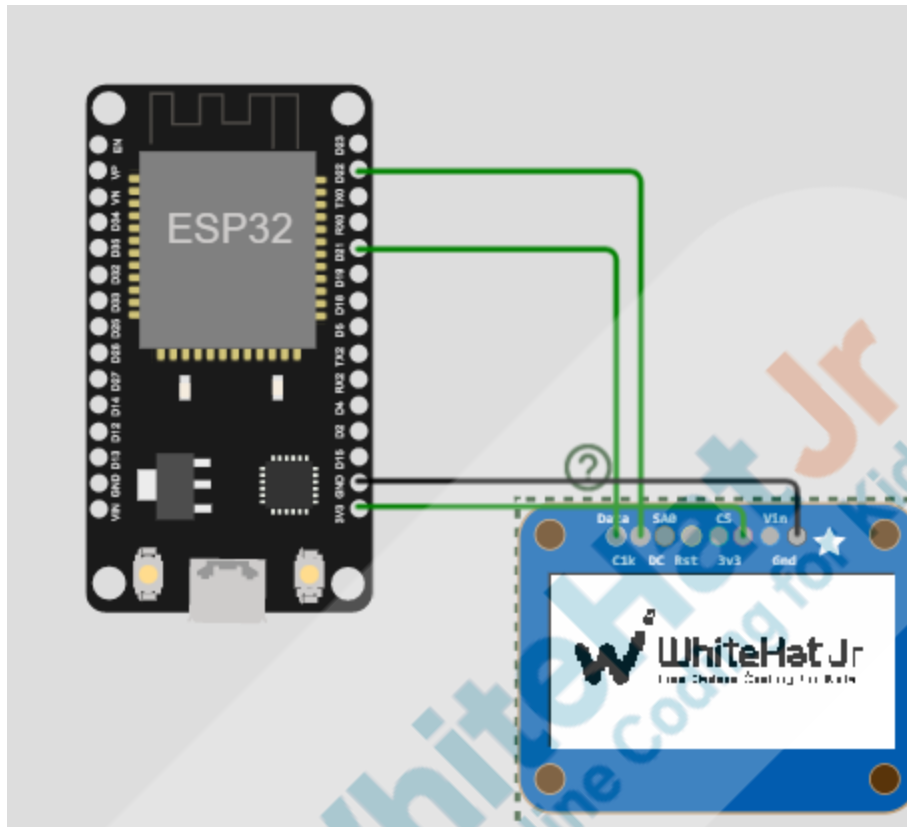
Click on the Save button and then click on the simulation button

Note:

If your OLED display is not showing anything:

Check that the OLED display is properly wired

Output will look like this:



Student Stops Screen Share

So its time to do the Activity, now you will convert your picture into a BITMAP array and display it on OLED
Please share your screen with me.

We have one more class challenge for you.
Can you solve it?

Let's try. I will guide you through it.

STUDENT-LED ACTIVITY-2 - 15 mins

- Ask the student to press the ESC key to come back to the panel.

- Guide the student to start Screen Share.
- The teacher gets into Full Screen.

Student Initiates Screen Share

ACTIVITY

- Picture OLED

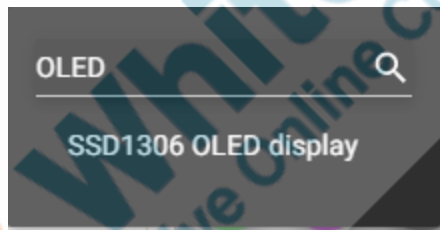
Teacher Action

Student Action

Note: The teacher will follow the same steps to convert the picture into a BITMAP array. Students can take their own pictures and generate their own BITMAP arrays.

Step -1: Select the material from the Simulator

- 1 x ESP32
- 1 x OLED Click on + Sign and selects OLED



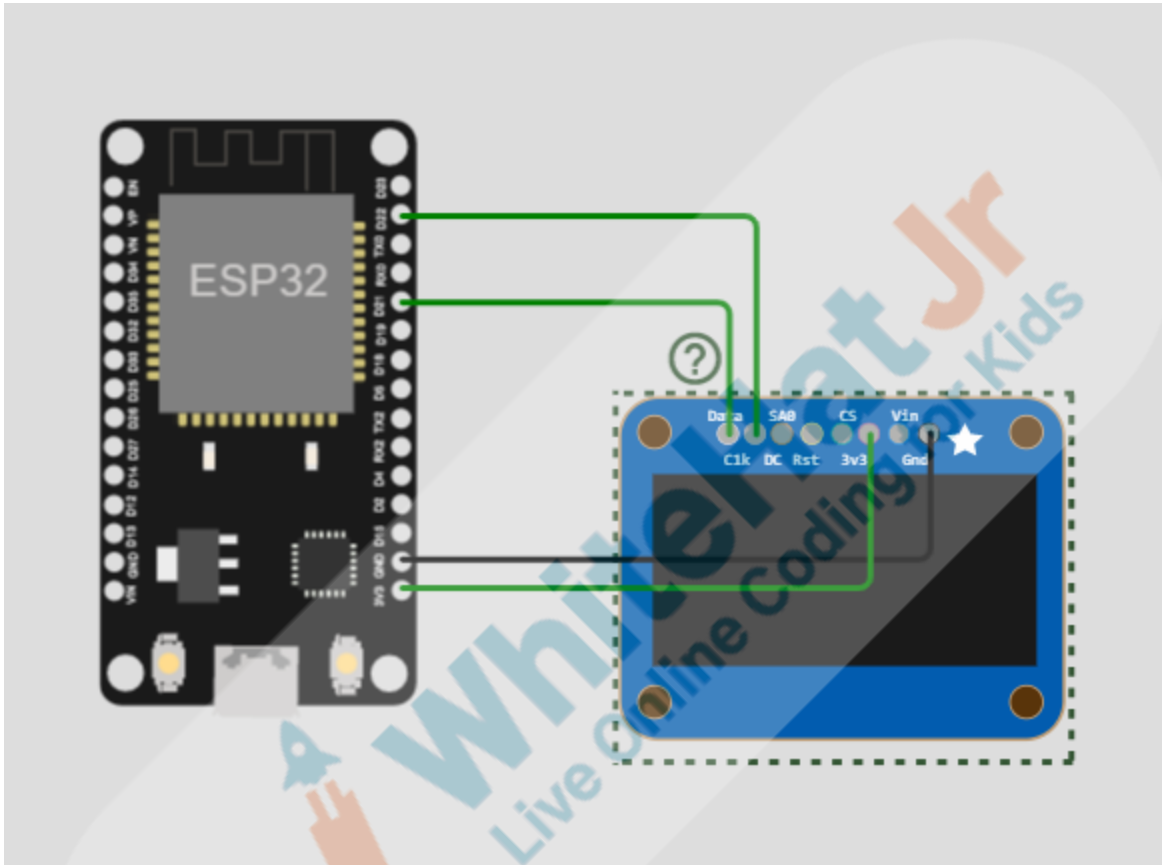
Step -2: Let's do connections:

- Insert OLED into the breadboard
- Take four jumper wires.


OLED PIN	ESP32 PIN
VCC	3.3 V
GND	GND
Data	GPIO 21

CLK

GPIO 22



To control the **OLED** display, we need to install libraries

5. Click on the small triangle icon  next to Library Manager
6. Select New File
7. Name the file libraries.txt
8. Write down **Adafruit SSD1306**

```
sketch.ino  diagram.json  ●  libraries.txt  ●  Library Manager  ▼  
1  # Wokwi Library List  
2  # See https://docs.wokwi.com/guides/libraries  
3  
4  Adafruit SSD1306  
5
```

Import Libraries:

- **Wire.h** This **library** allows you to communicate with I2C / devices. I2C is a **serial communication protocol**, so data is transferred bit by bit along a single wire.
- **Adafruit_GFX.h:** This library offers a common graphical syntax and set of functions for all LCD displays, OLED displays, and LED matrices.
- **Adafruit_SSD1306** : This library takes care of low-level communication with the hardware.
- Define **SCREEN_WIDTH & SCREEN_HEIGHT** for OLED .Our **OLED** size is **128×64**.
- Declaration of an **SSD1306** display that connects to **I2C** communication using **Wire** Library.
- Initialize a **display** object with the **SCREEN_WIDTH & SCREEN_HEIGHT** defined earlier with the I2C communication protocol.
- A value of **(-1)** indicates that our OLED display does not have a **RESET** pin. Sometimes OLED displays have a RESET pin on the OLED, in that case, we should connect it to a GPIO and should include the GPIO number as a parameter.


```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
```

The student must convert any picture into a BITMAP array at their end and use the same BITMAP array under this section.

Note the name of the BITMAP array. To call it, we need to use the same name.

The screenshot below is just for reference.

Note: This step is entirely needed to be done by the student. The teacher will guide it

```
const unsigned char bitmap_14v9pq[] PROGMEM = {

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,

B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111
};
```

Initialize using **void setup()** function

- Serial.begin(115200): Sets the **data rate** in bits per second (baud) for **serial** data transmission.
- Initialize the OLED display with the begin() method.
- If the OLED displays nothing, check the OLED address at **0x3C**. In our case, the address is 0x3C.
- If we are not able to connect to the display, it prints a message on the Serial Monitor.
- If something fails, don't proceed further, try to repeat the process using **for()** loop
- display.drawBitmap is used to display the picture
Write down the exact name of the BitMAP array.
Here in this case it is **bitmap_14v9q**

Note: This is a reference picture code, the student will right his BITMAP array name

```
void setup() {  
  Serial.begin(115200);  
  
  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {  
    Serial.println(F("SSD1306 allocation failed"));  
    for (;;);  
  };  
  
  delay(2000);          // wait two seconds for initializing  
  display.clearDisplay(); // clear display  
  
  display.drawBitmap(0, 0, bitmap_14v9pq, 128, 64, 1);  
  
  display.display();  
}
```

Output:

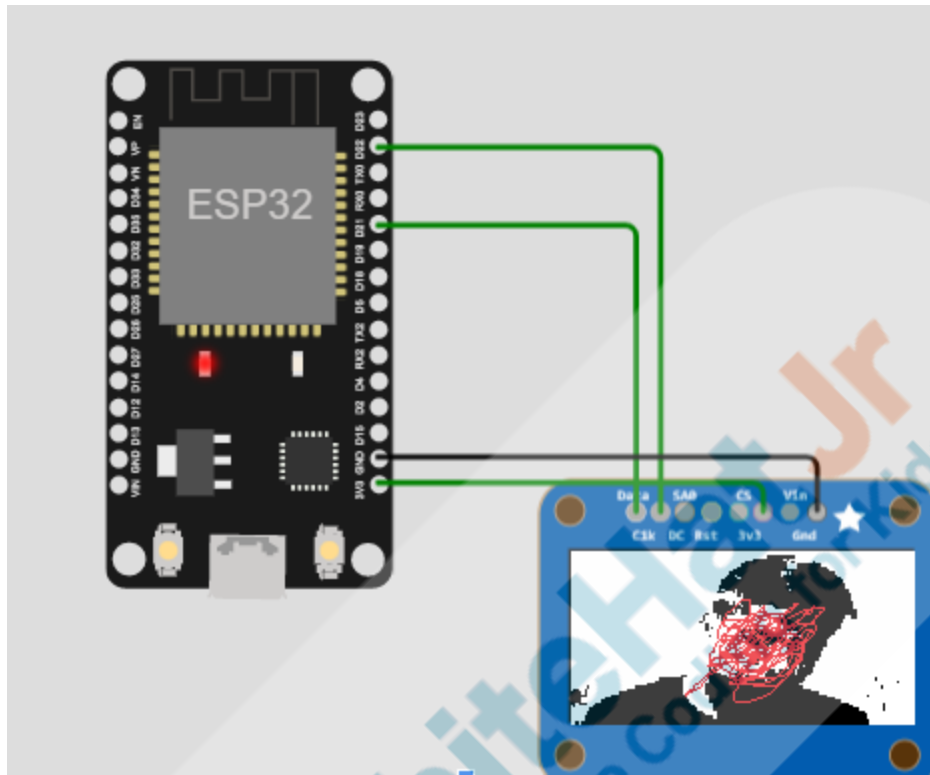
Click on the save button and then Click on restart the simulation

- If there is any error resolve it

Note:

If your OLED display is not showing anything:

Check that the OLED display is properly wired



Vow! This is how our pixels work for any picture.

That's fun!

Teacher Guides Student to Stop Screen Share

WRAP-UP SESSION - 05 mins

Activity details

Following are the WRAP-UP session deliverables:

- Appreciate the student.
- Revise the current class activities.
- Discuss the quizzes.

WRAP-UP QUIZ
Click on In-Class Quiz

Activity Details

Following are the session deliverables:

- Explain the facts and trivia
- Next class challenge
- Project for the day
- Additional Activity (Optional)

FEEDBACK

- Appreciate and compliment the student for trying to learn a difficult concept.
- Get to know how they are feeling after the session.
- Review and check their understanding.

Teacher Action

You get “hats-off” for your excellent work!

In the next class, we will learn about Electronic Voting Machine

Student Action

Make sure you have given at least 2 hats-off during the class for

Creatively Solved Activities  +10

Great Question  +10

Strong Concentration  +10

PROJECT OVERVIEW DISCUSSION

Refer the document below in Activity Links Sections

Teacher Clicks

✕ End Class

ADDITIONAL ACTIVITIES

(Optional)	
Additional Activities	

ACTIVITY LINKS		
Activity Name	Description	Links
Teacher Activity 1	Bit Map Array converter	https://marlinfw.org/tools/u8glib/converter.html
Teacher Activity 2	Reference Code -TA	https://github.com/procodingclass/Pro-C256-Reference-Code
Student Activity 1	Bit Map Array converter	https://marlinfw.org/tools/u8glib/converter.html
Student Activity 2	Reference Code -SA	https://github.com/procodingclass/Pro-C256-Reference-Code-SA