GUESS THE KEY

What is our GOAL for this CLASS?

In this class, we learned how to create a guess game using a 4X4 keypad.

What did we ACHIEVE in the class TODAY?

- We learned how to connect a keypad with ESP32.
- We used different coding concepts to create the game.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Concepts : Generating random numbers , conditional statements, infinite loops, sequencing of code, taking input from keypad, adding time interval without using delay method.
- Coding blocks : **random** method, **if-else** statements, **while** loops, keypad methods like **getKey** method, **break** command, **millis** method.

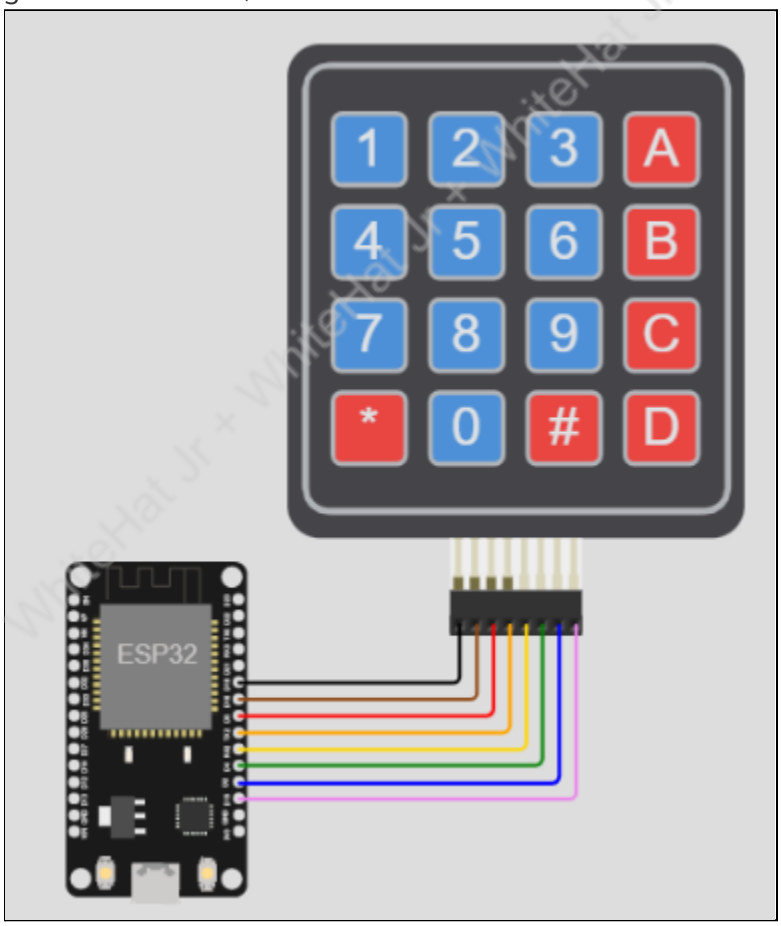
How did we DO the activities?

1. Connect the **4x4 keypad** with **ESP32** using the following instructions.

KEYPAD	ESP32 PIN
R1	GPIO 19
R2	GPIO 18

R3	GPIO 5
R4	TX2
C1	RX2
C2	GPIO 4
C3	GPIO 2
C4	GPIO 15

2. The circuit diagram will look like,



3. Define all the necessary **global** variables.

```
long int current_time = 0;
long int prev_time = 0;
int threshold = 4000;
char keypressed = '\0';
int score = 0;
```

4. Use the **random()** method to generate a **random row** and a **column number**, so that we can pick up a key code from the **keys** array.

```
byte random_row = random(1,5);
byte random_column = random(1,5);
char random_element = keys[random_row-1][random_column-1];
```

5. Create **2** hints. One which displays the **row number** and the other which displays the **column number**.

```
String hint1 = "Row : " + String(random_row);
String hint2 = "Column : " + String(random_column);
byte random_hint = random(1,3);
```

6. Use the **random()** method again to change the order of displaying the hints.

```
if (random_hint  ==  1)Serial.println(hint1 + "\t" + hint2);
else Serial.println(hint2 + "\t" + hint1);
```

7. Use an **infinite while** loop, so that you can listen for the **keypad inputs** from the user.

```
while (true){


  char key = k.getKey();


  if (key){
    keypressed = key;
  }
```

8. Start the timer using the **millis()** method.

```
current_time = millis();
```

9. Write a condition using the **if** statement which will be executed whenever we reach the **threshold** time interval.

```
if (current_time - prev_time  ==  threshold){
```

10. Write the **win** condition. Increase the **score variable** and decrease the **threshold time variable**, to make the game more interesting.

```
if (keypressed  ==  random_element){
  score++;
  if (score > 6){
    Serial.println("You WON!");
    while(1);
  }
  Serial.println("Correct guess, score : " + String(score));
  threshold = threshold - 500;
  if (threshold < 500)threshold = 500;
}
```

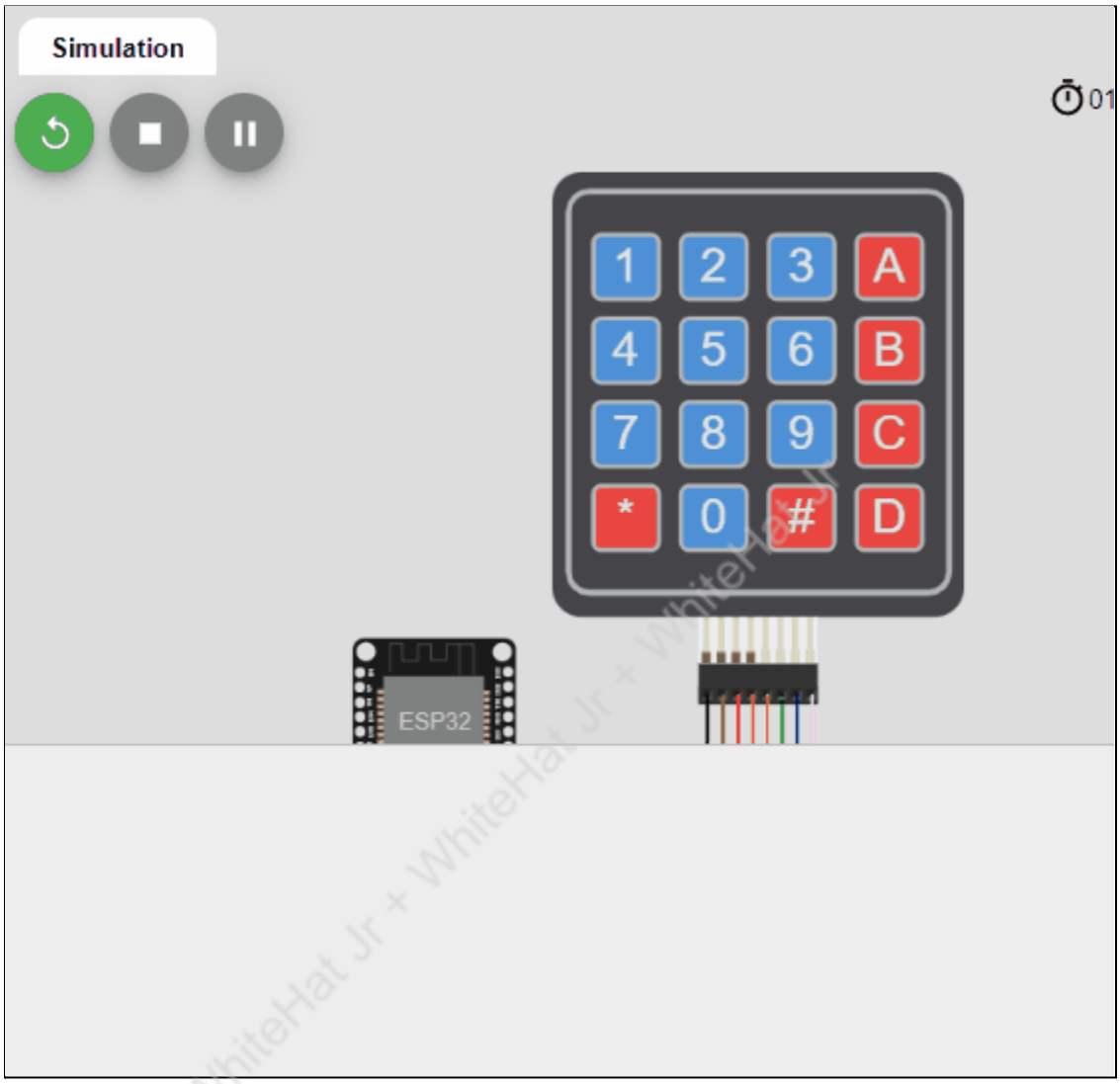
11. Write a condition if a player **loses** the game.

```
else{
  Serial.println("You LOSE!");
  while(1);
}
```

12. Finally, update the **prev_time** variable and **break** the loop.

```
prev_time = current_time;
break;
```

13. The output for the following code would look like,



What's NEXT?

In the **next class**, we will learn about **servo motors**.

Expand Your Knowledge

To know more about **keypads** on wokwi**,** click here.