

```

//Interactive LCD Display- Q & A
//Pat McMahon V1 17/10/2023
// From chatGPT Using a (PCF8574T) i2C Display advancing 10 messages on
pushing a button.
// I altered the number of message lines to [26] for my example.
// Connections- LCD to Uno- Gnd to Gnd, VCC to +5V, SDA to A4, Scl to A5.
// Push Button Switch- One terminal to +5V, Second terminal to D7, also from
the Second Terminal a second wire with a 10K pulldown resistor,connected to
Gnd.
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// The default address is usually 0x27 or 0x3F
const int lcdAddress = 0x27;

// Define LCD properties
LiquidCrystal_I2C lcd(lcdAddress, 16, 2);

// Define button pin
const int buttonPin = 7;

// Variable to store the current message index
int currentMessageIndex = 0;

// Array of messages
// Change the No [26] below to suite the number of message lines you have in
your Code.
const char messages[26][17] = {

  "Press to Start ",
  "Hello & Welcome!",
  "What's your Name",
  "Are you a      ",
  "Student       ",
  "or a Teacher? ",
  "Great!        ",
  "What school are ",
  "you from?     ",
  "Do you like STEM",
  "Activities?   ",
  "Have you checked",
  "out          ",
  "Pat's website, ",
  "patsrobots.com ",
  "Over 120 STEaM ",
  "Projects      ",
  "to share free  ",
  "with info on  ",

```

```

    "How to Build    ",
    "How to Code    ",
    "Photos, Videos",
    "etc.           ",
    "Worth checking ",
    "out, CHEERS, PAT",
};

void setup() {
{

    lcd.init();           // This initializes the LCD.
    lcd.backlight();     // This turns on the backlight of the LCD.
    lcd.clear();         //This wipes the screen.
    lcd.setCursor(0,0);  //Sets the screen to start on the top left,
top line of screen.

}

    // Set up the LCD
    lcd.begin(16, 2);
    Serial.println("Button Pressed");
    // Set up the button pin as an input
    pinMode(buttonPin, INPUT);
}

void loop() {
    // Read the state of the button
    int buttonState = digitalRead(buttonPin);

    // Check if the button is pressed
    if (buttonState == HIGH) {
        // Increment the message index
        // Change 26 below to suite the number of message lines you have above
        currentMessageIndex = (currentMessageIndex + 1) % 26;

        // Display the current message on the LCD
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(messages[currentMessageIndex]);

        // Wait for a short time to debounce the button
        delay(500);
    }
}
}

```

