```
#include <lvgl.h>
#include <TFT_eSPI.h>

/* Change to your screen resolution */
static const uint16_t screenWidth  = 320; // Landscape
static const uint16_t screenHeight = 240;

static lv_disp_draw_buf_t draw_buf;
static lv_color_t buf[screenWidth * 10];
TFT_eSPI tft = TFT_eSPI(screenWidth, screenHeight);

/* Display flushing: required by LVGL to push data to the screen */
void my_disp_flush(lv_disp_drv_t *disp, const lv_area_t *area, lv_color_t *color_p) {
   uint32_t w = (area->x2 - area->x1 + 1);
   uint32_t h = (area->y2 - area->y1 + 1);
   tft.startWrite();
   tft.setAddrWindow(area->x1, area->y1, w, h);
   tft.pushColors((uint16_t *)&color_p->full, w * h, true);
   tft.endWrite();
   lv_disp_flush_ready(disp);
}

void setup() {
   Serial.begin(115200);
// --- MANUALLY TURN ON BACKLIGHT ---
   pinMode(21, OUTPUT);      // Set pin 21 as output
   digitalWrite(21, HIGH);   // Set to HIGH to turn on the backlight
   // ----------------------------------
```

```
lv_init();
tft.begin();
tft.setRotation(1); // Landscape

lv_disp_draw_buf_init(&draw_buf, buf, NULL, screenWidth * 10);

/* Initialize the display driver for LVGL */
static lv_disp_drv_t disp_drv;
lv_disp_drv_init(&disp_drv);
disp_drv.hor_res = screenWidth;
disp_drv.ver_res = screenHeight;
disp_drv.flush_cb = my_disp_flush;
disp_drv.draw_buf = &draw_buf;
lv_disp_drv_register(&disp_drv);

// --- UI ELEMENTS ---

/* 1. Create a Label (Text) */
lv_obj_t *label = lv_label_create(lv_scr_act());
lv_label_set_text(label, "ESP32-32E Test");
lv_obj_align(label, LV_ALIGN_TOP_MID, 0, 20);

/* 2. Create a "Circle" (Using an object with rounded corners) */
lv_obj_t *circle = lv_obj_create(lv_scr_act());
lv_obj_set_size(circle, 100, 100);
lv_obj_align(circle, LV_ALIGN_CENTER, 0, 0);
lv_obj_set_style_radius(circle, LV_RADIUS_CIRCLE, 0); // Makes it a circle
lv_obj_set_style_bg_color(circle, lv_palette_main(LV_PALETTE_BLUE), 0);
```

```
}

void loop() {
    lv_timer_handler(); /* Let LVGL do its work */
    delay(5);
}
```