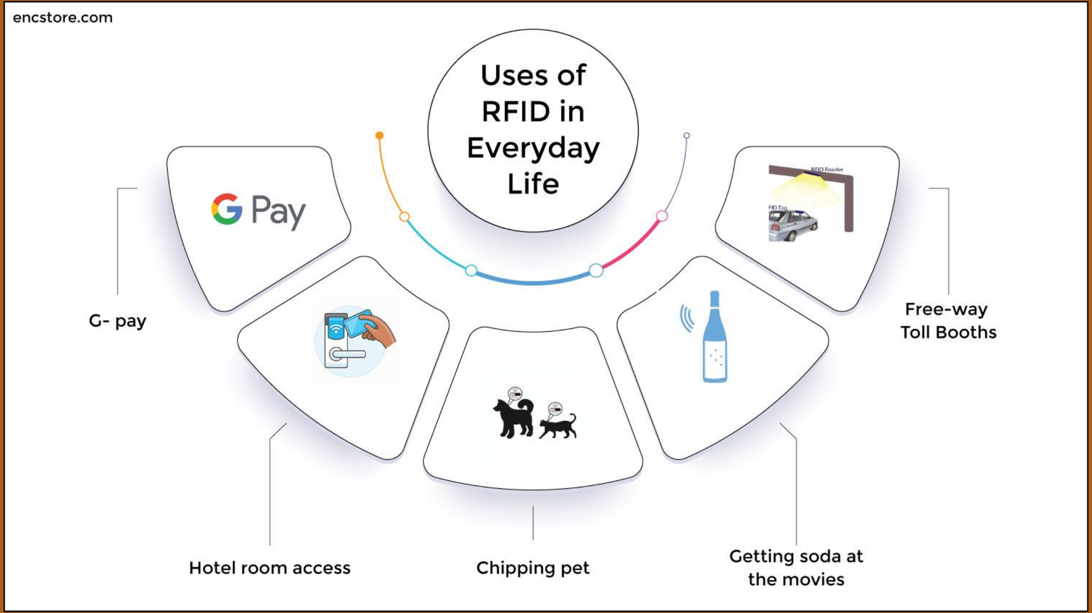
RFID Lock Box

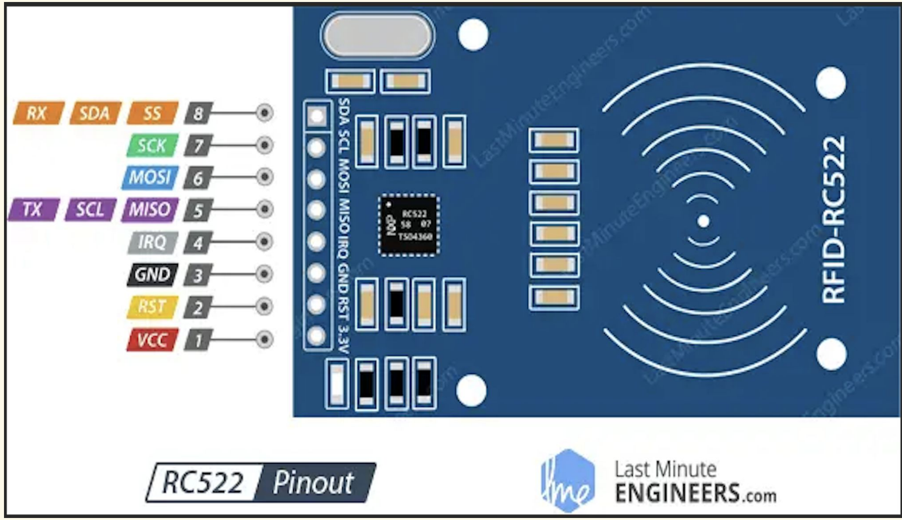
By Wyatt Bordeaux


encstore.com

FRID

- **RFID stands for Radio Frequency Identification.** It uses radio waves to transfer data wirelessly.
- **Two main parts: the tag and the reader.**
  1. **Tag:** Attached to the object; contains a tiny chip and antenna.
  2. **Reader (or scanner):** Sends out a signal to detect nearby tags.
- **How it works:**
  1. The **reader sends a radio signal**.
  2. The **tag receives the signal and sends back its data** (like an ID number).
  3. The **reader collects the data** and sends it to a computer system.
- **Used in:** inventory tracking, contactless payment, ID badges, pet microchips, etc.

PIN 13 -> SCK
PIN 12 -> SCL
PIN 11 -> MOSI
PIN 10 -> SDA
PIN 7 -> RST
3.3 V -> VCC



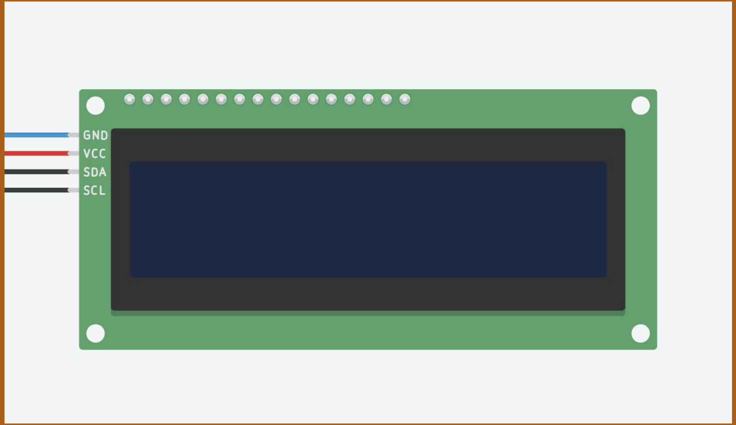
Servo and LCD

- A servo motor rotates to a specific angle based on a control signal (PWM).

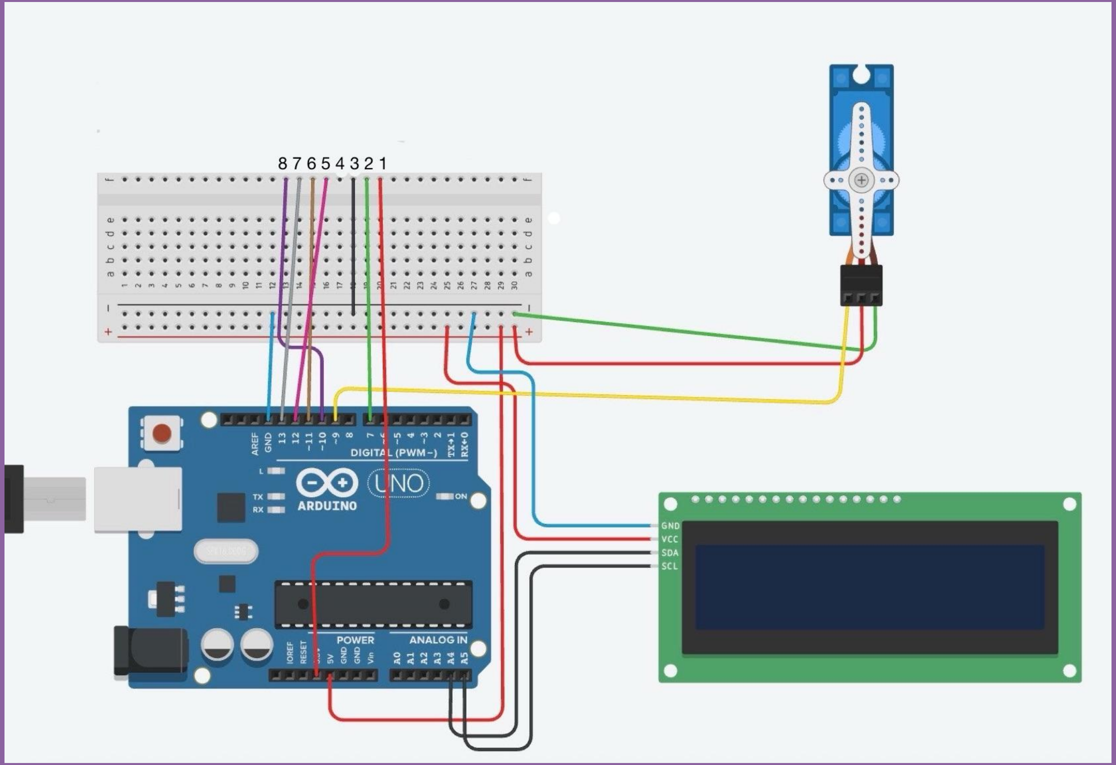
- Controlled by the Arduino, the servo turns when access is granted, acting as the electronic lock mechanism in your system.

- The **MCP23008** is an I²C I/O expander that lets the Arduino control more pins using just two wires (SDA/SCL), enabling it to run the LCD screen with fewer pins.

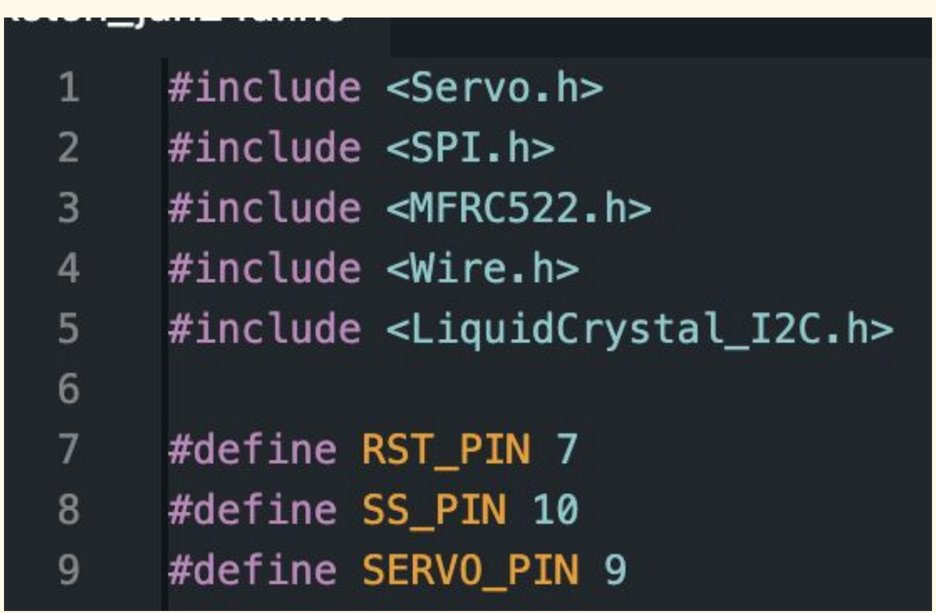
- The LCD displays the entered key and whether access is **"GRANTED"** or **"DENIED"**, giving real-time feedback to the user through text.



RFID Schematic



CODE

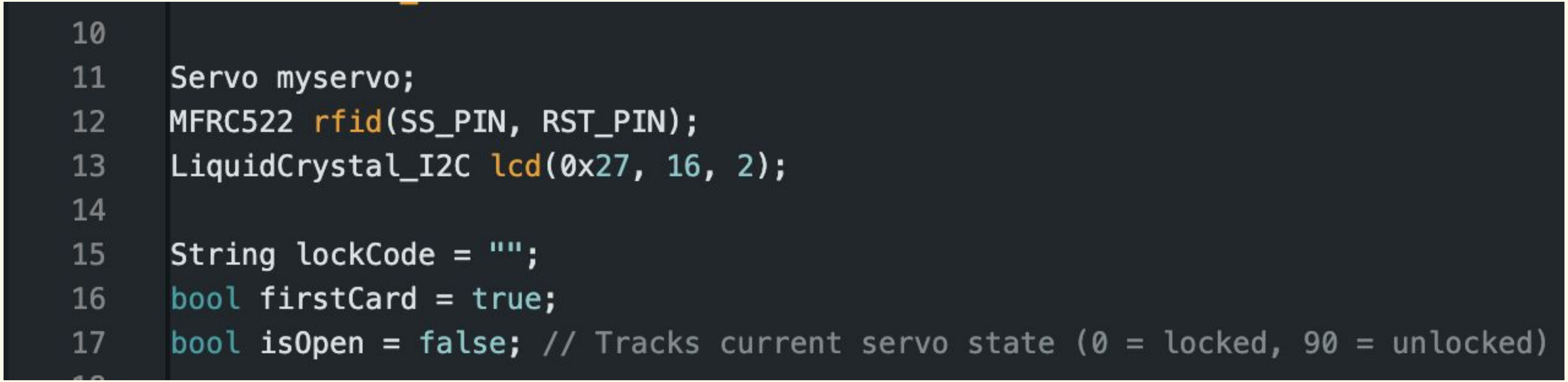


```
#include <Servo.h>
#include <SPI.h>
#include <MFRC522.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define RST_PIN 7
#define SS_PIN 10
#define SERVO_PIN 9
```

- Includes all necessary libraries for servo control, RFID communication, and I2C LCD.
- Defines which pins are used for the RFID module and the servo motor.

CODE

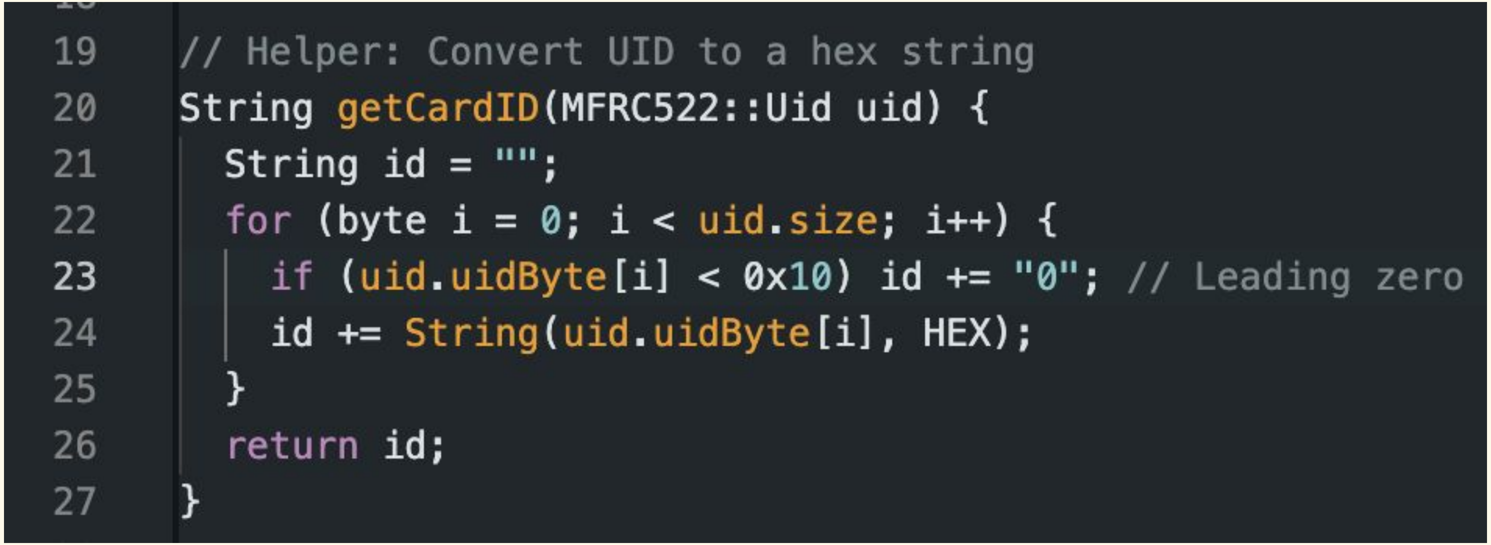
- initializes a servo object, Initializes the RFID reader with its chip select and reset pins.
- Sets up the I²C LCD at address `0x27`, with 16 columns and 2 rows.
- `lockCode`: stores the authorized card’s UID.
- `firstCard`: flag to indicate if the system is waiting for the first card to set as a key.
- `isOpen`: tracks whether the lock is open or locked



```

Servo myservo;
MFRC522 rfid(SS_PIN, RST_PIN);
LiquidCrystal_I2C lcd(0x27, 16, 2);

String lockCode = "";
bool firstCard = true;
bool isOpen = false; // Tracks current servo state (0 = locked, 90 = unlocked)
```

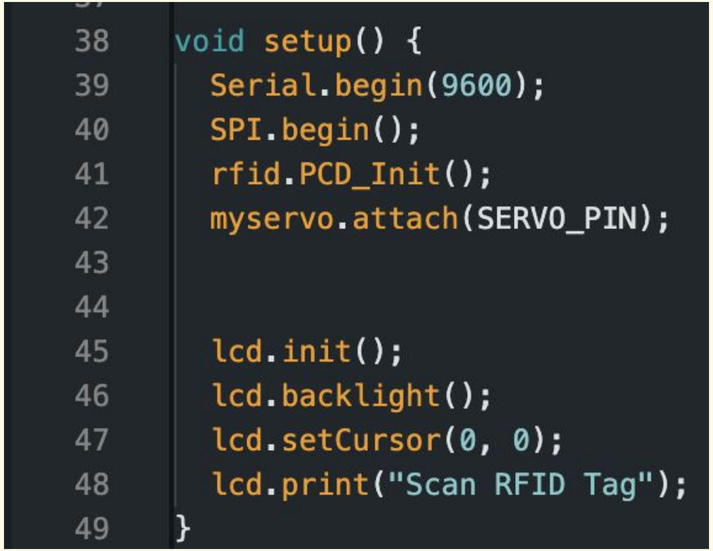
CODE

- Converts the card’s UID bytes into a readable hexadecimal string.
- Adds a leading zero to any byte less than `0x10` for proper formatting.



```
// Helper: Convert UID to a hex string
String getCardID(MFRC522::Uid uid) {
  String id = "";
  for (byte i = 0; i < uid.size; i++) {
    if (uid.uidByte[i] < 0x10) id += "0"; // Leading zero
    id += String(uid.uidByte[i], HEX);
  }
  return id;
}
```

CODE

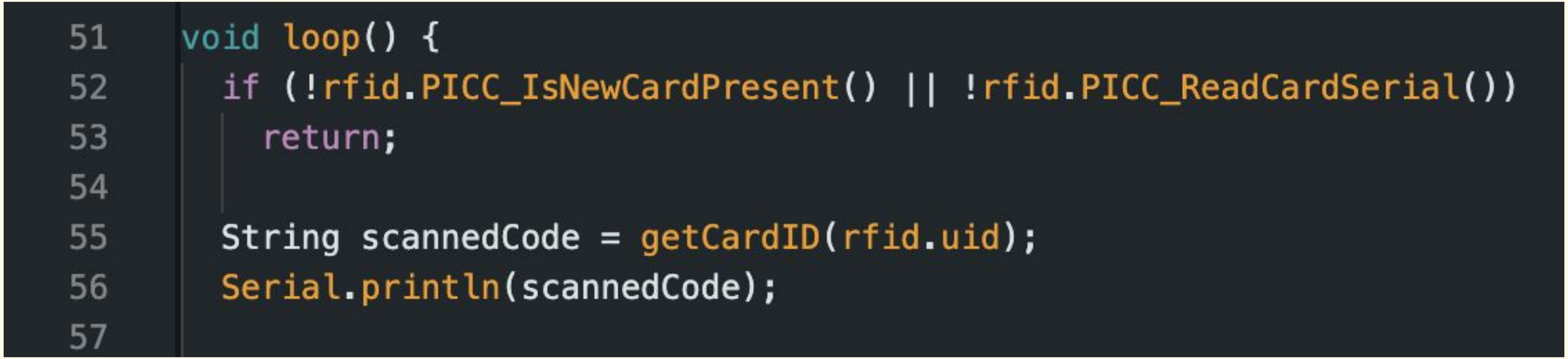


```
void setup() {
  Serial.begin(9600);
  SPI.begin();
  rfid.PCD_Init();
  myservo.attach(SERVO_PIN);


  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Scan RFID Tag");
}
```

- Starts the serial monitor and SPI communication.
- Initializes the RFID reader and servo motor.
- Initializes the LCD, turns on backlight, and prompts the user to scan an RFID tag.

CODE

- Start of the Loop logic
- Waits until a new card is presented and successfully read.
- Converts scanned UID to a string and prints it for debugging.



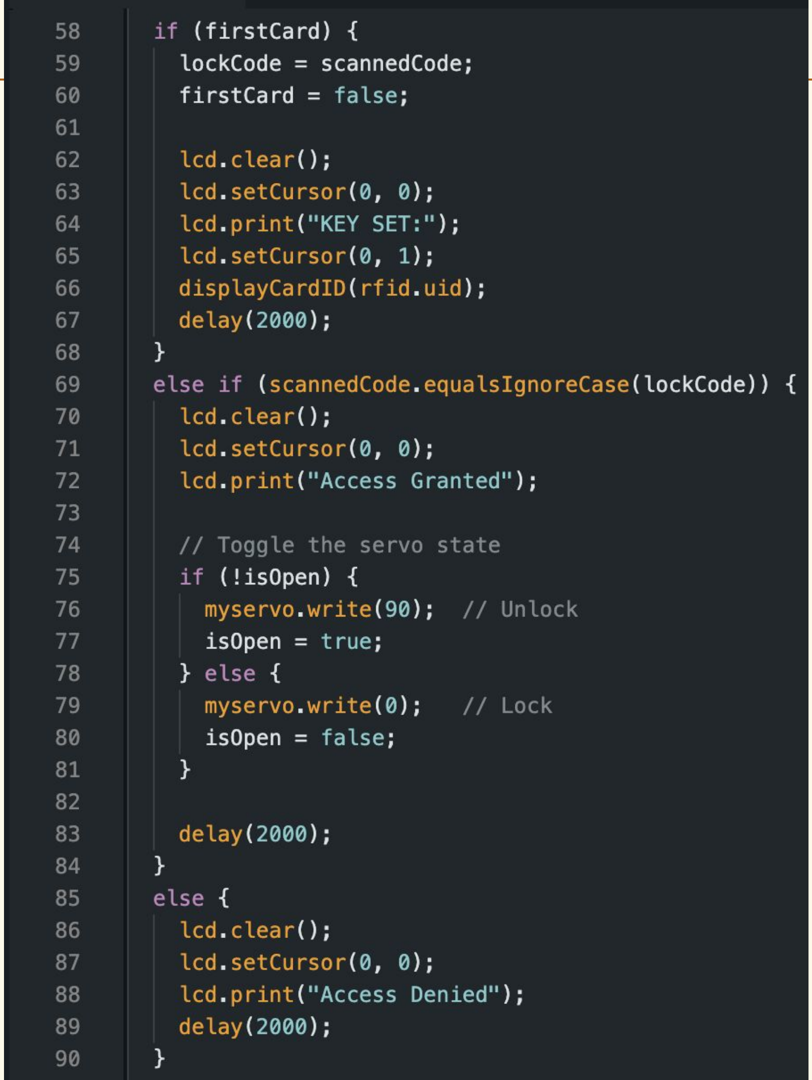
```
void loop() {
  if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())
    return;

  String scannedCode = getCardID(rfid.uid);
  Serial.println(scannedCode);

```

CODE

- The very first card scanned becomes the “key.”

- Stores its UID and displays confirmation on LCD.

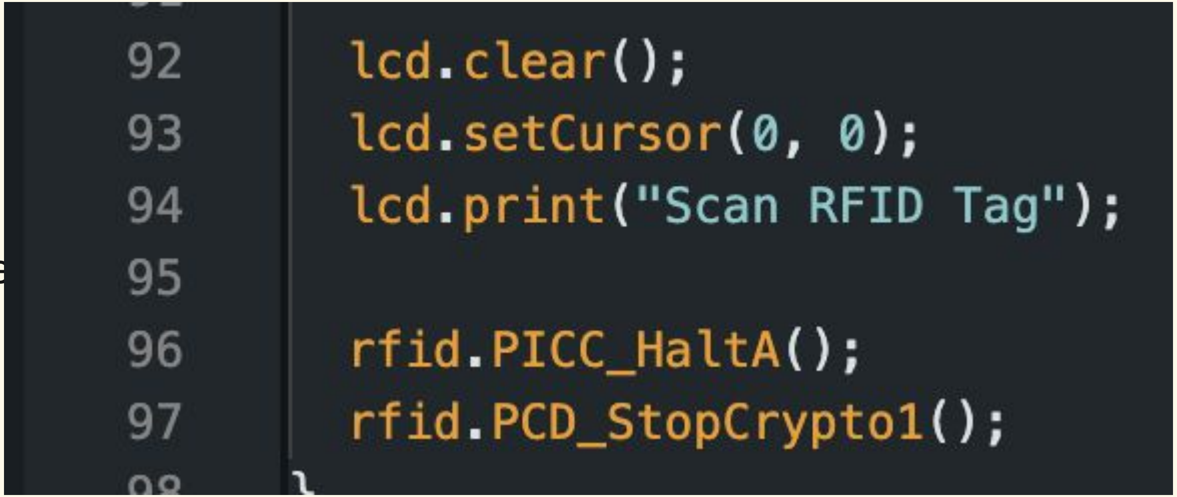
- If scanned UID matches the stored key: Displays “Access Granted” and toggles the servo (unlocks if locked, locks if unlocked)
- Else display “Access Denied” on lcd



```
if (firstCard) {
  lockCode = scannedCode;
  firstCard = false;

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("KEY SET:");
  lcd.setCursor(0, 1);
  displayCardID(rfid.uid);
  delay(2000);
}
else if (scannedCode.equalsIgnoreCase(lockCode)) {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Access Granted");

  // Toggle the servo state
  if (!isOpen) {
    myservo.write(90);  // Unlock
    isOpen = true;
  } else {
    myservo.write(0);   // Lock
    isOpen = false;
  }

  delay(2000);
}
else {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Access Denied");
  delay(2000);
}
```

CODE

- Clear and Reset the text on the LCD screen.
- Halts communication with the RFID tag to prevent repeated reads.
- Ends the secure communication session between the reader and the tag. (So the reader’s state is prepared for the next tag)



```
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Scan RFID Tag");

    rfid.PICC_HaltA();
    rfid.PCD_StopCrypto1();
}
```

THANK YOU