```
#include <Servo.h>
#include <Wire.h>
#include <ESP8266WiFi.h>
#include "ThingSpeak.h"

#include <LiquidCrystal_I2C.h>
WiFiClient client;


Servo servo1;
Servo servo2;
LiquidCrystal_I2C lcd(0x27, 16, 2);
int inputPin1 = D0; // choose input pin (for Infrared sensor)
int inputPin2 = D1;
int inputPin3 = D2;
int inputPin4 = D3;

int val1 = 0; // variable for reading the pin status
int val2 = 0;
int val3 = 0;
int val4 = 0;
const char* ssid = "IOT LAB";
const char* password = "iotlab@507";
int slot=2;
unsigned long myChannelNumber = 1;
const char * myWriteAPIKey = "7MQA95ZCS0QMCRG9";


void setup()
{
  servo1.attach(D4);
  servo1.write(0);

 WiFi.mode(WIFI_STA);
ThingSpeak.begin(client);  // Initialize ThingSpeak

  servo2.attach(D5);
  servo2.write(0);


   pinMode(inputPin1, INPUT); // declare Infrared sensor as input
   pinMode(inputPin2, INPUT);
   pinMode(inputPin3, INPUT);
   pinMode(inputPin4, INPUT);


  lcd.init();                      // Initialize the display
```

```
  lcd.backlight();                  // Turn on the backlight
  lcd.print("Smart Parking");       // Display the message on the first
line
  lcd.setCursor(0, 1);              // Move cursor to the second line
  lcd.print("Initializing...");

   Serial.begin(9600);
}

void loop()
{

  // Connect or reconnect to WiFi
if(WiFi.status() != WL_CONNECTED){
  Serial.print("Attempting to connect");
  while(WiFi.status() != WL_CONNECTED){
    WiFi.begin(ssid, password);
    delay(5000);
  }
  Serial.println("\nConnected.");
}

   val4 = digitalRead(inputPin4); // read input value
   //Serial.print(val1);

    if (val4 == 0)
   {

      servo1.write(90);
      delay(3000);
      servo1.write(0);
   }

   val1 = digitalRead(inputPin1);
   if (val1 == 0)
   {
      servo2.write(90);
      delay(3000);
      servo2.write(0);
   }

   val2 = digitalRead(inputPin2);
   val3 = digitalRead(inputPin3);
   if(val2==0 && val3==0){
     slot=0;
   }
   else if(val2==1 && val3==1){
```

```
    slot=2;
   }
   else{
    slot=1;
   }
   Serial.print("Slot Free:");
   Serial.print(slot);
   Serial.print("\n");
  lcd.setCursor(0, 1);  // Move cursor to the second line
  lcd.print("Slot Free: ");
  lcd.print(slot);       // Display the slot status on the LCD
  Serial.print(slot);
  //Serial.print("\n");



  int x = ThingSpeak.writeField(myChannelNumber, 1, slot, myWriteAPIKey);
if(x == 200){
  Serial.println("Channel update successful.");
}
else{
  Serial.println("Problem updating channel. HTTP error code " +
String(x));
}


delay(1000);
}
```