CODE

```
#include <Servo.h>

// Objects and Pin Definitions
Servo doorServo;

int pirPin = 2;              // PIR sensor pin
int lightLedPin = 3;         // LED for smart lighting
int acLedPin = 4;            // LED for AC simulation
int buttonPin = 5;           // Button for appliance control
int applianceLedPin = 6;     // LED for appliance status
int tempPin = A0;            // TMP36 temperature sensor pin

float temperature = 0.0;

void setup() {
  // Initialize components
  pinMode(pirPin, INPUT);
  pinMode(lightLedPin, OUTPUT);
  pinMode(acLedPin, OUTPUT);
  pinMode(buttonPin, INPUT);
  pinMode(applianceLedPin, OUTPUT);

  doorServo.attach(9);
  doorServo.write(0); // Door initially closed

  Serial.begin(9600); // For debugging
  Serial.println("System Initialized");
}

void loop() {
  // PIR Motion Sensor
  int motionState = digitalRead(pirPin);
  if (motionState == HIGH) {
    digitalWrite(lightLedPin, HIGH); // Turn on light
    doorServo.write(90);           // Open door
    Serial.println("Motion Detected: Door Open, Light ON");
```

```
    delay(5000);                  // Keep door open for 5 seconds
  } else {
    digitalWrite(lightLedPin, LOW); // Turn off light
    doorServo.write(0);           // Close door
    Serial.println("No Motion: Door Closed, Light OFF");
  }

  // Temperature Control (AC On/Off Simulation)
  int tempValue = analogRead(tempPin);
  temperature = (tempValue * 5.0 / 1023.0 - 0.5) * 100.0; // TMP36
conversion
  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.println(" C");

  if (temperature > 25.0) { // Threshold for AC
    digitalWrite(acLedPin, HIGH); // Turn AC on
    Serial.println("AC: ON");
  } else {
    digitalWrite(acLedPin, LOW); // Turn AC off
    Serial.println("AC: OFF");
  }

  // Voice Control (Simulated with Button)
  int buttonState = digitalRead(buttonPin);
  if (buttonState == HIGH) {
    digitalWrite(applianceLedPin,  !digitalRead(applianceLedPin));  //
Toggle appliance
    Serial.println("Appliance Toggled");
    delay(300); // Debounce
  }

  delay(200); // General delay for stability
}
```