CODE EXPLANATION

Variables and Constants:

1. **const int analogIn = A0;**
   - This line defines a constant integer 'analogIn' and assigns it the value 'A0'. This means 'analogIn' will refer to the analog pin A0 on the Arduino.
2. **int humiditysensorOutput = 0;**
   - This line initializes an integer variable 'humiditysensorOutput' to store the humidity sensor's output value.
3. **int RawValue=0;**
   - This line initializes an integer variable 'RawValue' to store the raw analog reading from the temperature sensor.
4. **double Voltage = 0;**
   - This line initializes a double variable 'voltage' to store the voltage value calculated from the raw analog reading.
5. **double tempC = 0;**
   - This line initializes a double variable 'tempC' to store the temperature in Celsius.
6. **double tempF = 0;**
   - This line initializes a double variable 'tempF' to store the temperature in Fahrenheit.

Setup Function: The setup() function runs once when the Arduino is powered on.

1. **Serial.begin(9600);**
   - Initializes the serial communication at a baud rate of 9600 bps. This allows the Arduino to communicate with the computer.
2. **pinMode(A1, INPUT);**
   - Sets the A1 pin as an input pin. This pin will be used to read the humidity sensor output.

Loop Function:

1. **RawValue = analogRead(analogIn);**
   - Reads the analog value from the temperature sensor connected to 'A0' and stores it in 'RawValue'.
2. **Voltage = (RawValue / 1023.0) * 5000;**
   - Converts the raw analog value to a voltage value in millivolts. The '1023.0' is the maximum value for a 10-bit ADC, and '5000' represents 5V in millivolts.
3. **tempC = (Voltage - 500) * 0.1;**
   - Converts the voltage to a temperature in Celsius. The formula assumes the sensor output is offset by 500 mV and each 10 mV represents 1°C.
4. **tempC = (tempC * 1.8) + 32;**
   - Converts the temperature from Celsius to Fahrenheit.