```
classdef Thermistor < handle & matlab.mixin.CustomDisplay
    %Thermistor Class for thermistors connected to ADS1115
    %{
    The constructor sets up the I2C device object. The standard address is
    0x48, but can be set to 0x49,0X4A,or 0x4B. For Radiometer, make sure
    address is 0x48 or 0x49 and for thermistors it is  0x4A or0x4B.

    Default pin and voltage scale are 0 and 4.096. To change simply add in
    constructor.

    The Read method requires no input.
    %}
    properties
        PIN = 0;
        V_SCALE = 4.096;
        Address = hex2dec('0x4A');
        TF %Transfer Function
    end

    methods
        function obj = Thermistor(arduino,model,pin,addrs,v_scale)
            %UNTITLED6 Construct an instance of this class
            %   Detailed explanation goes here
            switch nargin
                case 3
                    obj.PIN= pin;
                case 4
                    obj.PIN = pin;
                    obj.Address = addrs;
                case 5
                    obj.PIN = pin;
                    obj.V_SCALE = vscale;
                    obj.Address = addrs;
            end

            switch model
                case 1%first model of thermistor
                    obj.TF = @(v) v;
                case 2%Second model of thermistor
                    obj.TF = @(v) v;
            end

            obj.AR = arduino;
            obj.Therm = ads1115(obj.AR, obj.Address);

            obj.Therm.VoltageScale = obj.V_SCALE;
        end

        function T = Read(obj)
            %METHOD1 Summary of this method goes here
            %   Detailed explanation goes here
            v = readVoltage(obj.Therm,obj.PIN);

            T = obj.TF(v);
        end
    end
    methods
        %This simply allows it to take hex or numeric input
        function set.Address(obj, value)
            if isnumeric(value)
                validateattributes(value, {'numeric'}, ...
                    {'scalar', 'nonnegative'}, '', 'Address');
            else
                validateattributes(value, {'char'}, ...
                    {'nonempty'}, '', 'Address');
                value = obj.hex2dec(value);
            end
            if (value < obj.hex2dec('0x48')) || (value > obj.hex2dec('0x51'))
                error('raspi:ads1115:InvalidI2CAddress', ...
                    'Invalid I2C address. I2C address must be one of the following: 0x48,↙
0x49, 0x50, 0x51');
            end
            obj.Address = value;
        end
    end
end
```