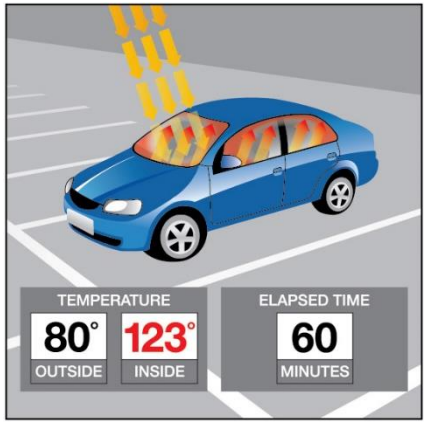
Pet Car Alarm: Final Challenge- Building the Pet Car Alarm

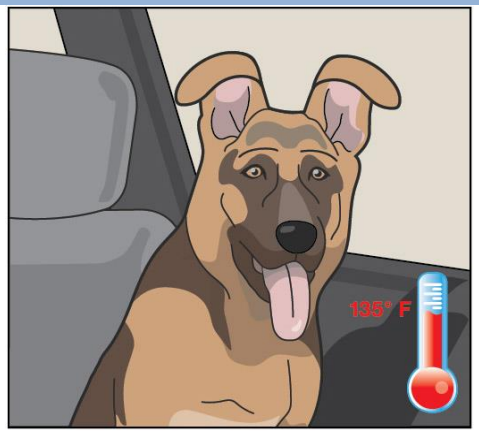
In the final challenge, you will put together the skills you have learned in prior challenges, and create a system that detects if the pet is in the car and measure the temperature. The car should sound a warning with flashing lights and roll down the windows if your pet could be in danger.

Goals:

1. Use the SET LED command to turn led lights on and off.
2. Use the SET SERVO.CONTINUOUS command to roll down the car window
3. Use the READ TEMPERATURE command to read ambient temperature using a sensor.
4. Use a Grove Hall effect magnetic proximity sensor to determine if a pet is in the car.
5. Use the SET SOUND command to sound a warming.
6. Use an If statement to make a selection.
7. Use a For loop or a While loop to repeat code.

Background:





Estimated Vehicle Interior Air Temperature v. Elapsed Time

Elapsed Time	Outside Air Temperature (F)					
	70	75	80	85	90	95
0 minutes	70	75	80	85	90	95
10 minutes	89	94	99	104	109	114
20 minutes	99	104	109	114	119	124
30 minutes	104	109	114	119	124	129
40 minutes	108	113	118	123	128	133
50 minutes	111	116	121	126	131	136
60 minutes	113	118	123	128	133	138
> 1 hour	115	120	125	130	135	140

Inside Air Temperature (F)

Light to Heat

Each summer, there are horrifying stories of children and pets left in hot cars. Ultimately, most end up having a heat stroke and in many cases pass away. The inside of a car heats up so much faster than the outside due to the greenhouse effect. Rays of sunlight stream into the vehicle through the windows and strike, the light strikes the surfaces of the interior of the car. That visible light is absorbed and reradiated as infrared light. Infrared radiation has a larger wavelength than visible light. The infrared radiation is unable to escape back through the windows. The trapped radiation causes the temperature inside the car to rise faster than the outside temperature.

Stayin' Cool!

Mammals such as humans, dogs, and cats all have ways to regulate temperature, thermoregulation, to maintain homeostasis. Humans sweat to increase the removal of heat from the body through evaporation. Dogs usually pant to remove heat although they also have a small number of sweat glands in the pads of their

paws. Cats will sprawl out on surfaces that are relatively cool to help remove body heat. They will lick their paws and rub the saliva on warmer parts of their bodies to increase evaporative cooling which is a similar mechanism as sweating in humans. When thermal regulation mechanisms are unable to maintain homeostasis, that mammal will go into heat distress.

The problem with too much heat

Heat distress leads to brain impairment, dehydration, heart failure, cell swelling, and protein denaturation… and potentially death.
Conduct research into why mammals such as humans, dogs, and cats are unable to endure hot environments like closed cars in the summer for long periods. Then design a solution using technology to protect car occupants from heat distress. Refine your design until you have a working prototype. With your teacher's permission, compare your prototype with those of your classmates to determine which team has the "best" system.

Command	Example	Behavior
CONNECT <type> <number> TO <port>	`Send "CONNECT TEMPERATURE 1 TO IN1"`	Associates the first TEMPERATURE object with a temperature module plugged into port IN1 on the Hub.
SET <type><number> TO <value>.	`Send "SET LED 1 TO ON"`	Turns on LED 1. Other parameters may include BLINK <rate> TIIME<duration> "SET LED 1 ON BLINK 3 TIME 20"
SET <type><number> TO <value> [<blink rate>] [<duration in seconds>]	`Send "SET LED 1 ON BLINK 3 TIME 20"`	Sets LED 1 to a blink rate of 3 times per second for 20 seconds. See Hub Settings menu for ON, OFF, BLINK, etc.
SET <type> <number> TO <value>	`Send "SET SERVO.CONTINUOUS 1 CW 20 TIME 1"`	Turns on the first continuous servo motor object at power 20 (range 0-100) in the clockwise direction (CW vs. CCW) for a time of 1 second.
Wait <number>	`Wait 3`	Pauses the program for a specified number of seconds. In this case 3 seconds.
READ <type> <number>	`Send "READ TEMPERATURE 1"`	Reads a measurement from the first temperature object.
Get <variable>	`Get t`	Stores the temperature measurement into a variable named *t*. The value stored will contain the measurement from the immediately preceding READ command. Note: a Get command must immediately follow a READ command.
DispAt <line #> , <"text"> , <variable name>	`DispAt 3, "Temperature = ", t`	When variable *t* has a value of 26, the following line is displayed on the calculator: Temperature = 26 (temp readings are in °C by default)

For <counter variable>,<start value>,<end value>,[<step value>]
<statements>
EndFor
	`For n,1,10`
	`  DispAt 3,n`
	`EndFor`
		Runs For loop 10 times, starting at 1 and ending at 10. Executes the statement in the block each time, displays the value for the counter variable on row 3.

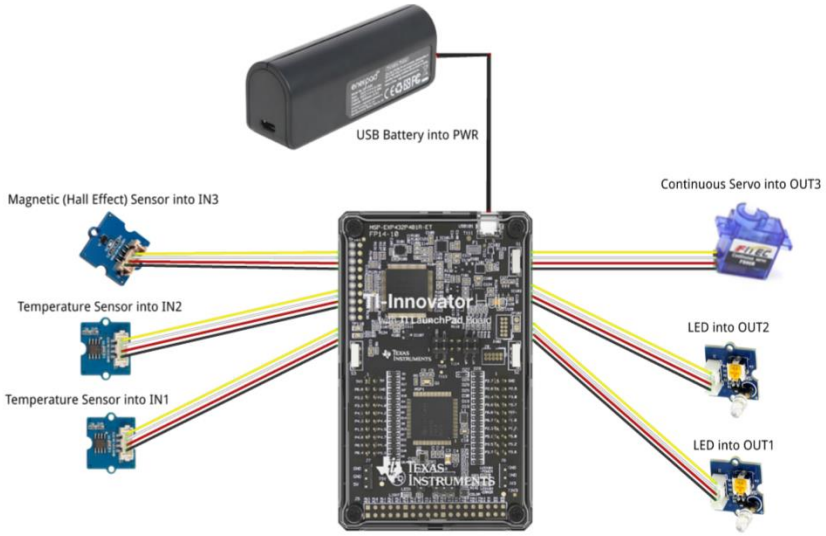
While <Boolean expression>	`key:=" "`	The statements in the While loop are executed until the escape key is pressed. The

<statements> EndWhile	**While *key*≠"esc"** **READ "TEMPERATURE 1"** **Get *t*** **DispAt 3,"Temperature= ",*t*** ***key*:=getKey()** **Wait 1** **EndWhile**	While loop continues as long as the Boolean expression evaluates to "true". The variable *key* is set to an initial value of an empty string using the assign function, ":=". getKey() is a function that returns a string with the name of the last key pressed while a programming is running. In this program the value of getKey() is stored to the variable *key*.

<Boolean expression> and <Boolean expression>	**If *t*>25 and *m*<100 Then**
	Send "SET COLOR 0 255 0"
	EndIf
		When both expressions are true the "and" function is "true" and the statements are executed. Otherwise, the function returns false and the statements are skipped.

If <Boolean expression> Then
<statements 1>
Else
<statements 2>
End
	If m<100 Then
	DispAt 5,"Magnet is present"
	Else
	DispAt 5,"Magnet is not present"
	EndIf
		The example decision tree has a Boolean expression with corresponding statements to execute if true. It also has an Else condition that executes corresponding statements when the Boolean expression is false. This Else condition ensures that a set of statements will always be executed. When this decision tree executes, focus proceeds from top-down. If the value of *m* is less than 100 the commands after Then are executed. In this case the output "Magnet is present" is displayed on row 5. If the value of *m* is greater than or equal to 100 the commands after Else are executed. In this case the output "Magnet is not present" is displayed on row 5.

See TI-Innovator Technology eGuide for more background https://education.ti.com/html/webhelp/EG_Innovator/EN/index.html



- Calculator
- Unit to Unit Cable
- TI-Innovator Hub
- Grove Cable x5
- Grove Temperature sensor x2
- Fashion Doll Car, shoebox, or another object to model a car
- Grove Hall effect magnetic proximity sensor
- Grove White LED Light x2
- Grove Continuous Servo motor
- External USB Battery w/ Cable

Challenge:

Use the skills developed in the earlier challenges to develop an alarm system for a model of a car that determines if a pet is present (magnet) AND the temperature reading inside the car reaches a critical threshold before triggering the alarm.

Some things to consider when you make your design:

1.) What temperature do you consider too hot for a pet in a car? How will you use this information in your design?

2.) How will you use the LED lights in your design?

3.) How will you use the SET SOUND command to sound a warning?