

Basic Expressions

```

{{ "Hello World" }}      Outputs Hello World
{{ foo }}                Outputs value of 'foo'
                           variable
{{ 2 + 3 }}              Outputs number 5
{{ 5 > 10 }}             Outputs "False"
    
```

If Statement

```

{% if hour > 5 and hour < 12 %}
    Good Morning!
{% elif number >= 12 and hour < 17 %}
    Good Afternoon!
{% elif number >= 17 %}
    Good Evening!
{% endif %}
    
```

For Loop

```

{% for number in 10 %}
    {{ number }}
{% endfor %}
    
```

For loop with conditions

```

{% for item in states if item.attributes.emulated_hue_name|length > 0-%}
    {% if loop.first %}{% elif loop.last %}, and
    {% else %}, {% endif -%}
    {{item.attributes.emulated_hue_name|title}}
    {%- endfor -%}.
    
```

When iterating through a loop, sometimes we would like to add additional logic based on the loop counter. This is an example.

String Manipulations

```

{{ "Hello" ~ "World" }}      Outputs "Hello World"
                               (String Concatenation)
{{ "Hello World" | lower }}  Outputs "hello world"
{{ "hello world" | title }}   Outputs "Hello World"
{{ "hello world" | upper }}   Outputs "HELLO WORLD"
    
```

String Manipulations (cont)

```

{{ "hello world".split(' ') }} Splits by delimiter ' ' and
                               outputs "hello"
{{ "hello" | replace("he", "she") }} Replaces "he" with "she",
                                       and outputs "shello"
{{ "hello" | length }}           Outputs number of characters,
                                       which is 5
{{ "hello" == "hello" }}        Compares strings, if equal,
                                       returns True
{{ "Hello world"[:4] }}         Get Substring - Returns "-
                                       Hell", the first 4 characters
{{ "HELLO" [2:4] }}             Get Substring - Returns "-
                                       LL"
{{ "HELLO"[:-3] }}             Get Substring - Removes
                                       last three characters,
                                       Returns "HE"
    
```

Date and Time

```

{{ now() }}                    Prints current date and time value
{{ now().strftime("%m-%d-%Y") }} Outputs "07-26-2017"
                               by converting
                               datetime value to the
                               specified format
{{ as_timestamp(now()) | timestamp_custom("%m-%d-%Y", true) }} Outputs "07-26-2017"
                               by converting
                               datetime value to the
                               specified format
{{ now().strftime("%m-%m") }}  Outputs Month value.
                               For ex: 07
    
```

For additional date and time related formats, please visit <http://strftime.org/>

Number Conversions

```

{{ 10.5 }}                    Prints the number as is
{{ 10.500482737 | int }}      Converts float to decimal
{{ 10.6 | round }}            Rounds to nearest decimal number
{{ 10.4 | round }}            Rounds to nearest decimal number
{{ 10 | float }}              Converts Decimal to float
{{ '%0x' % 255 }}             Changes Decimal to Hex Value
{{ "%0x" | format (some_numeric_value | int) }} Converts numeric
                                       value to Hex Value
{{ '%0.2f' % 24.12345 }}      Only displays two digits after decimal
                                       point
{{ "${:.2f}".format(15.45) }} Formats currency
    
```

Basic Macros

```

{% macro sayHello() -%}
    Hello!
{%- endmacro %}
Call the macro using {{ sayHello() }}
    
```

Random

```

{{["hello", "hi there!", "howdy!", "hey there!"]|random}}
    
```

Troubleshooting

1. Always convert to proper data type prior to using any filters.
 For example,
 {{ set value = "123" }}
 {{ value > 100 }} given the value is of "string" type, make sure you convert to decimal first.
 {{ (value | int) > 123 }}

