



## 7: Additional Filters





# Person Filters



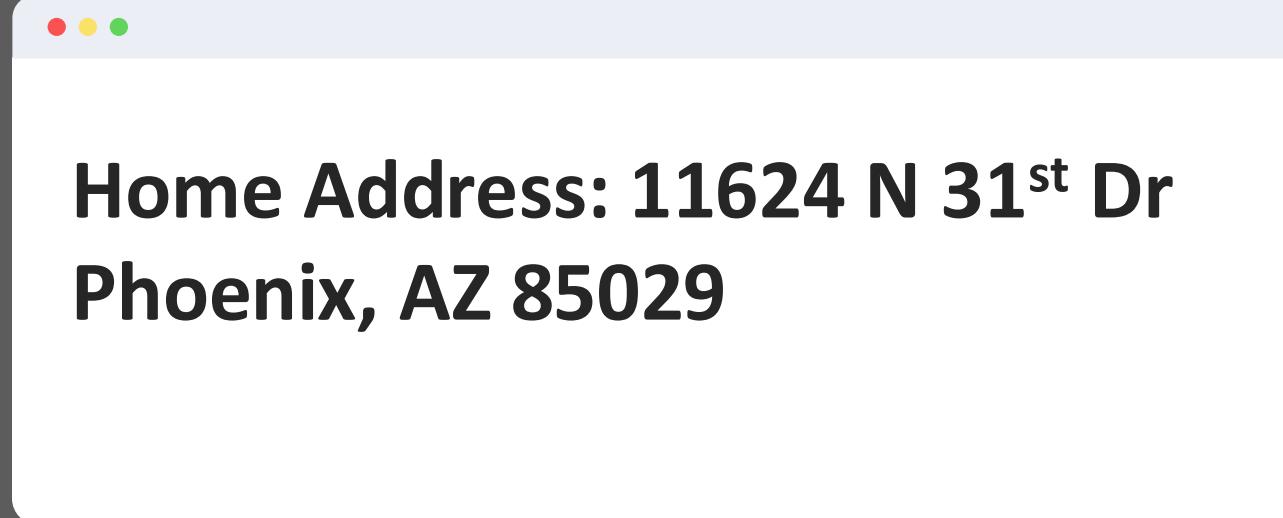


# Address

```
1 Home Address: {{ CurrentPerson | Address:'Home' }}
```

## Additional Options

1. **Address Type:** Home, Work, etc.
2. **Format Template:** Optional template parameter to format the address any way you wish.



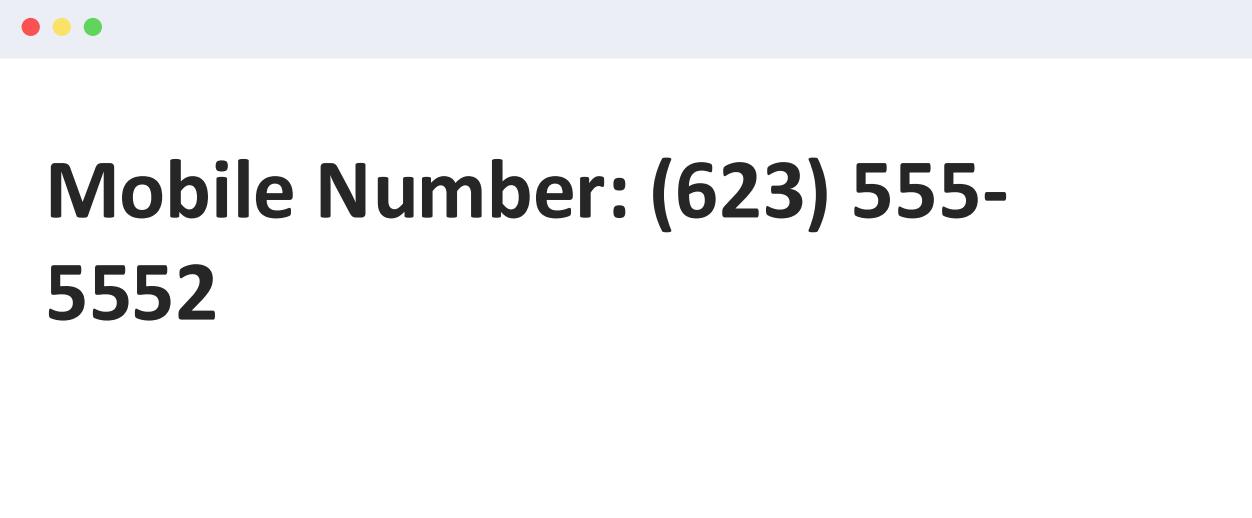


# PhoneNumber

```
1 Mobile Number: {{ CurrentPerson | PhoneNumber:'Mobile' }}
```

## Additional Options

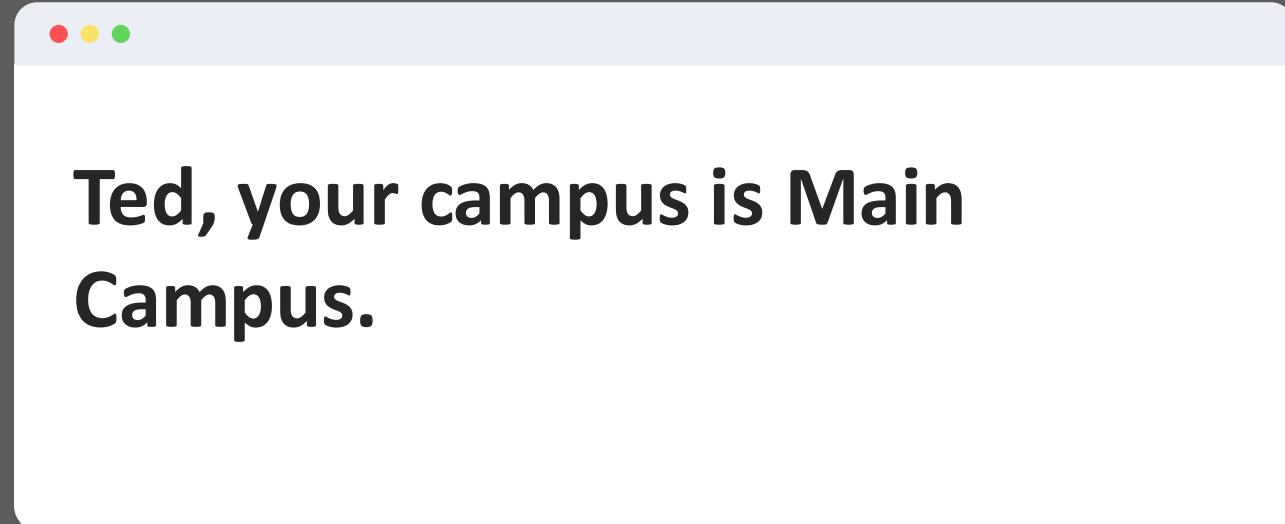
- 1. Phone Type:** Home, Mobile, Work, etc.
- 2. Show Country Code:** Set to true, to optionally display the country code.





# Campus

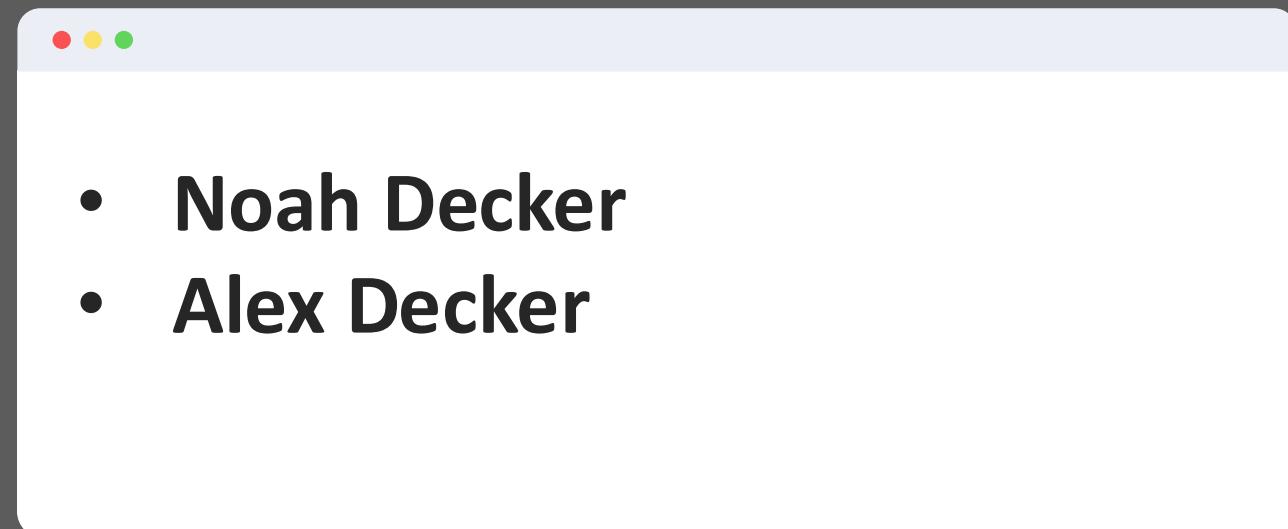
```
1  {% assign personCampus = CurrentPerson | Campus %}  
2  {{ CurrentPerson.NickName }},  
3  your campus is {{ personCampus.Name }}
```





# Children

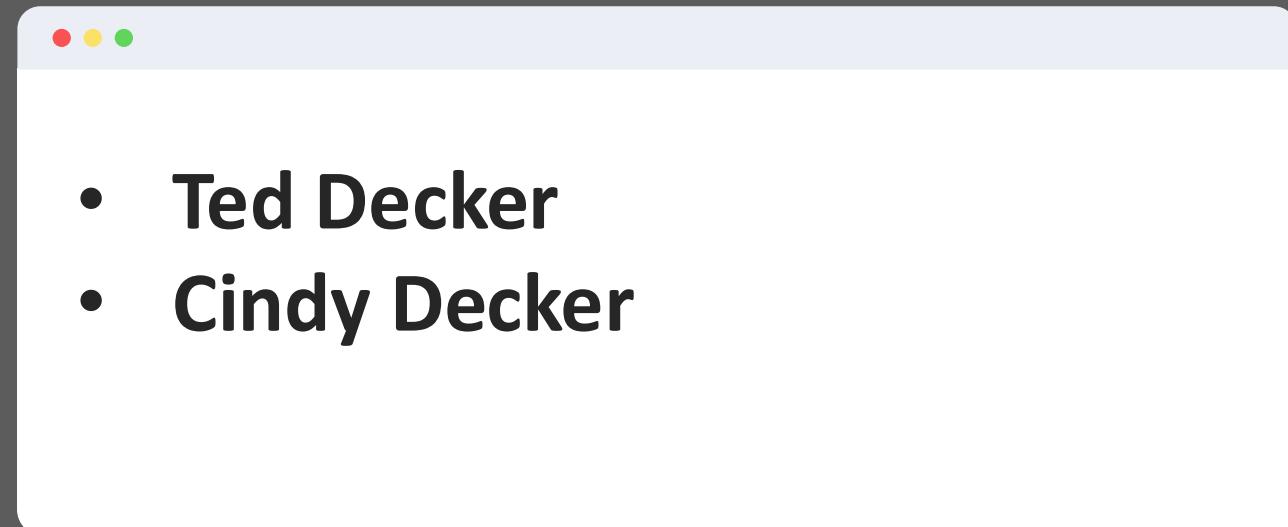
```
1  {% assign children = CurrentPerson | Children %}  
2  <ul>  
3  {% for child in children %}  
4      <li>{{ child.FullName }}</li>  
5  {% endfor %}  
6  </ul>
```





# Parents

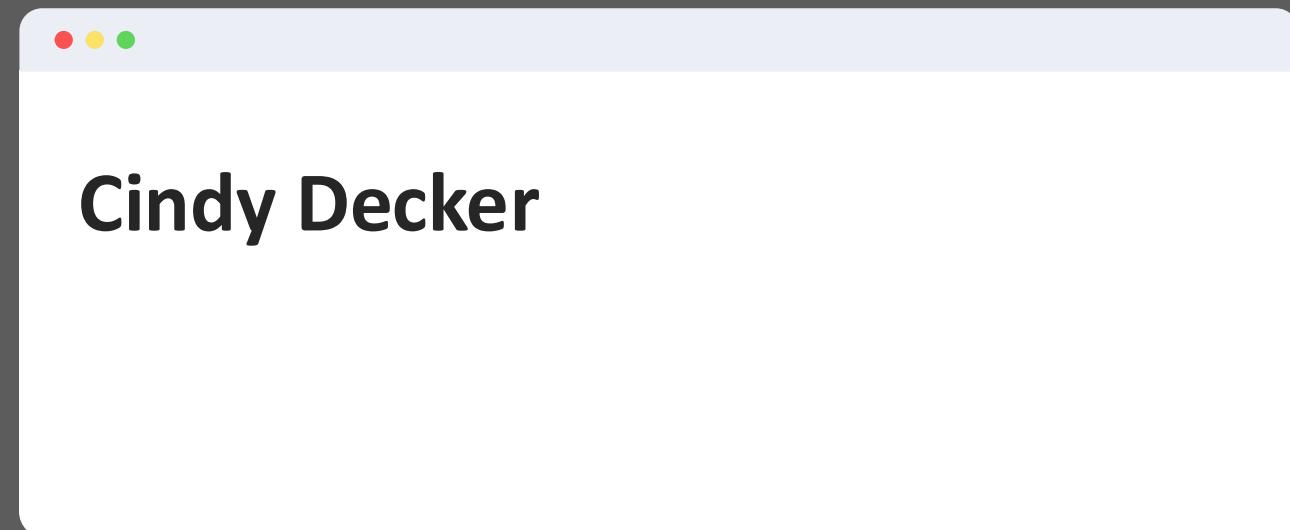
```
1  {% assign parents = CurrentPerson | Parents %}  
2  My parents are:  
3  {% for person in parents %}  
4      <li>{{ person.FullName }}</li>  
5  {% endfor %}
```





# Spouse

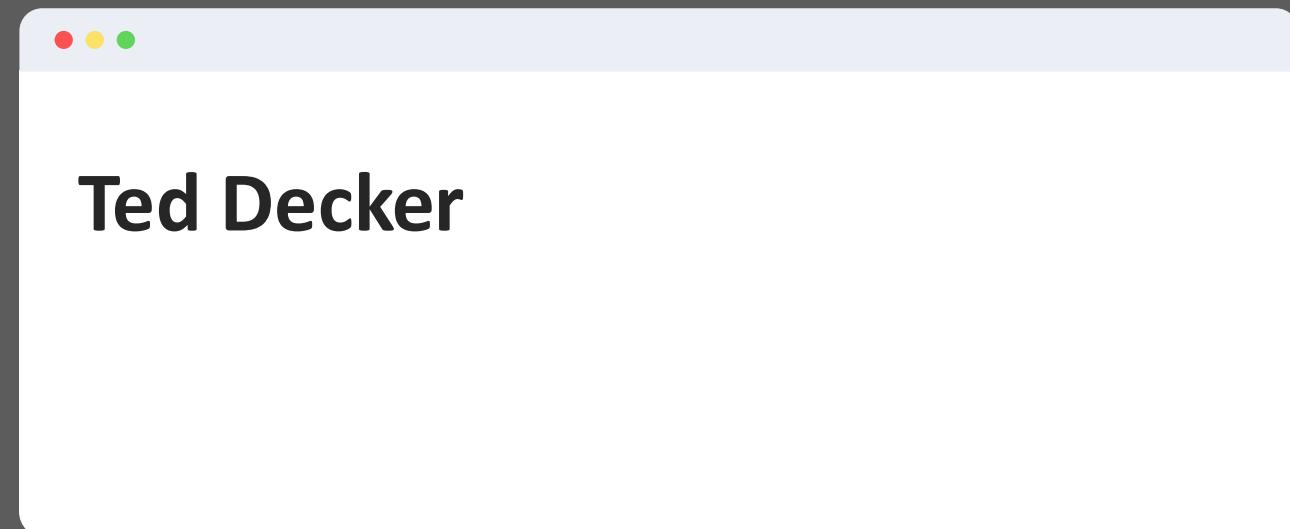
```
1  {% assign spouse = CurrentPerson | Spouse %}  
2  {{ spouse.FullName }}
```





# HeadOfHousehold

```
1  {% assign headOfHouse = CurrentPerson | HeadOfHousehold %}  
2  {{ headOfHouse.FullName }}
```



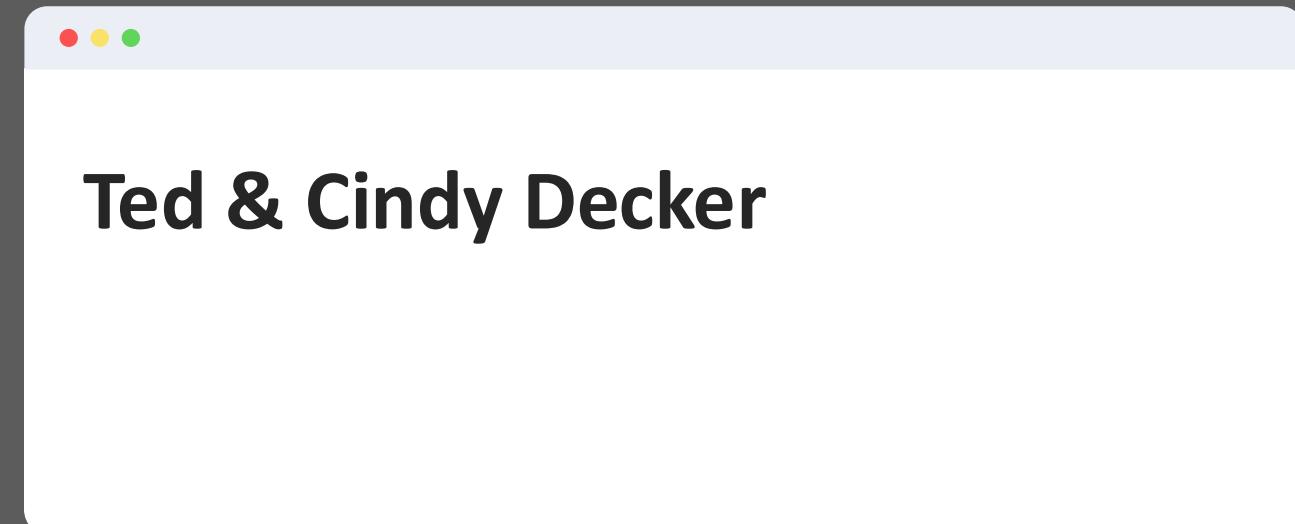


# FamilySalutation

```
1 {{ CurrentPerson | FamilySalutation }}
```

## Additional Options

1. **Include Children:** Should children be included. [Default: false]
2. **Include Inactive:** Should inactive people be shown. [Default: true]
3. **Use Formal Name:** [Default: false]
4. **Final Separator:** [Default: &]
5. **Separator:** [Default: , ]



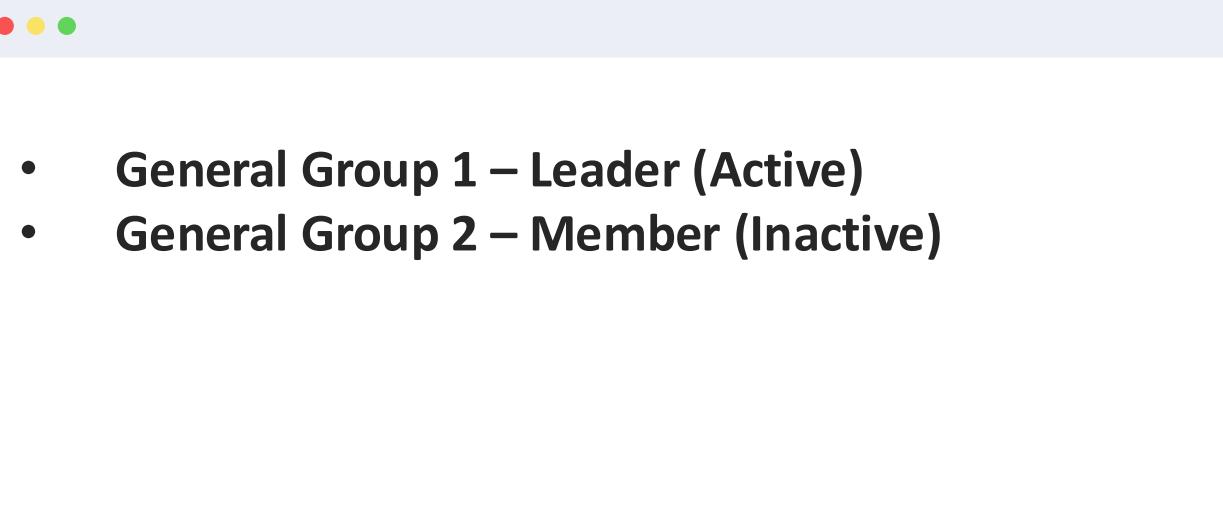


# Group

```
1  {% assign groupMembers = CurrentPerson | Group:'29','All' %}  
2  <ul>  
3  {% for groupMember in groupMembers %}  
4      <li>{{ groupMember.Group.Name }} -  
5          {{ groupMember.GroupRole.Name }}  
6          ({{ groupMember.GroupMemberStatus }})</li>  
7  {% endfor %}  
8  </ul>
```

## Additional Options

1. **Group Id**
2. **Group Member Status:** All, Active, Inactive, or Pending





# Groups

```
1  {% assign groupMembers = CurrentPerson | Groups:'29','All' %}  
2  <ul>  
3  {% for groupMember in groupMembers %}  
4      <li>{{ groupMember.Group.Name }}  
5          ({{ groupMember.GroupMemberStatus }})</li>  
6  {% endfor %}  
7  </ul>
```

## Additional Options

1. **Group Type**
2. **Group Member Status:** All, Active, Inactive, or Pending
3. **Group Status:** All, Active, or Inactive



- **General Group 1 (Active)**
- **General Group 2 (Inactive)**

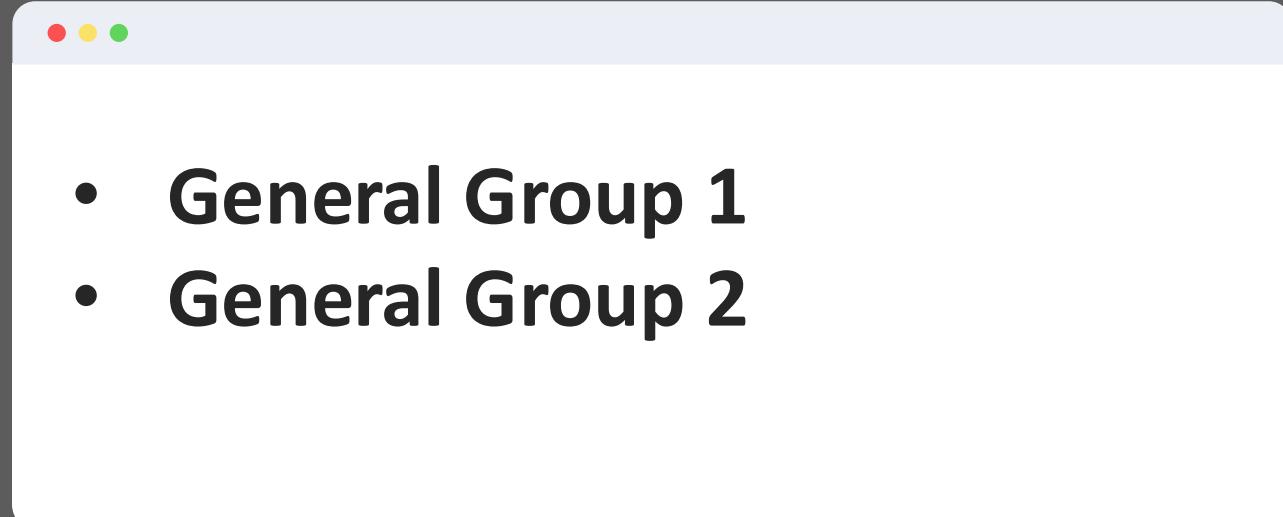


# GroupsAttended

```
1  {% assign groups = CurrentPerson | GroupsAttended:'29' %}  
2  <ul>  
3  {% for group in groups %}  
4      <li>{{ group.Name }}</li>  
5  {% endfor %}  
6  </ul>
```

## Additional Options

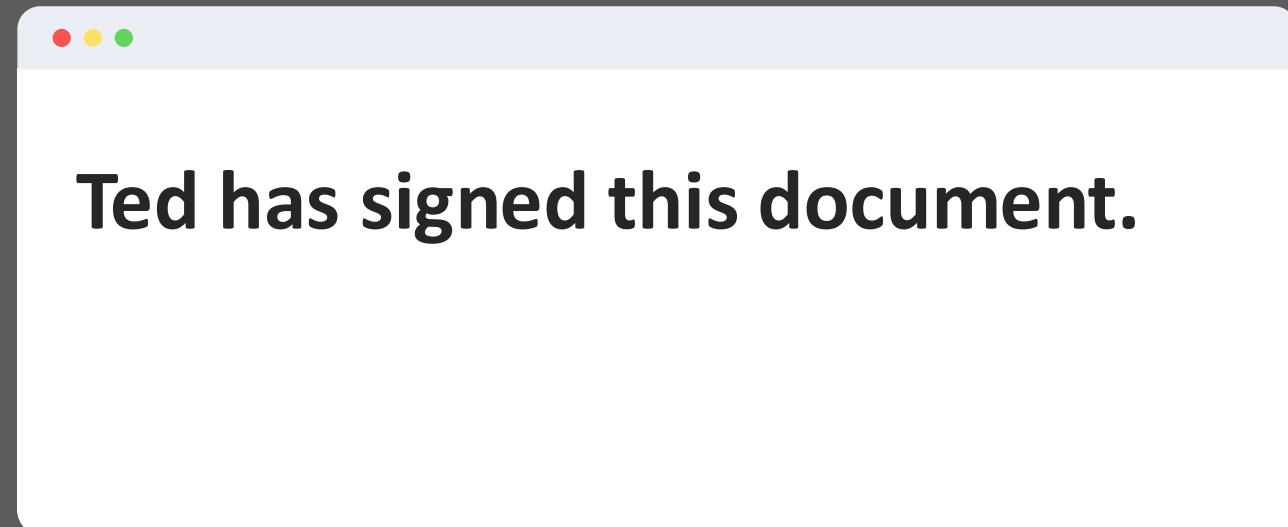
1. **Group Type Id:** The Id of the Group Type you are interested in.





# HasSignedDocument

```
1  {% assign hasSigned = CurrentPerson | HasSignedDocument:1 %}
2  {% if hasSigned %}
3      {{ CurrentPerson.NickName }} has signed this document.
4  {% endif %}
```



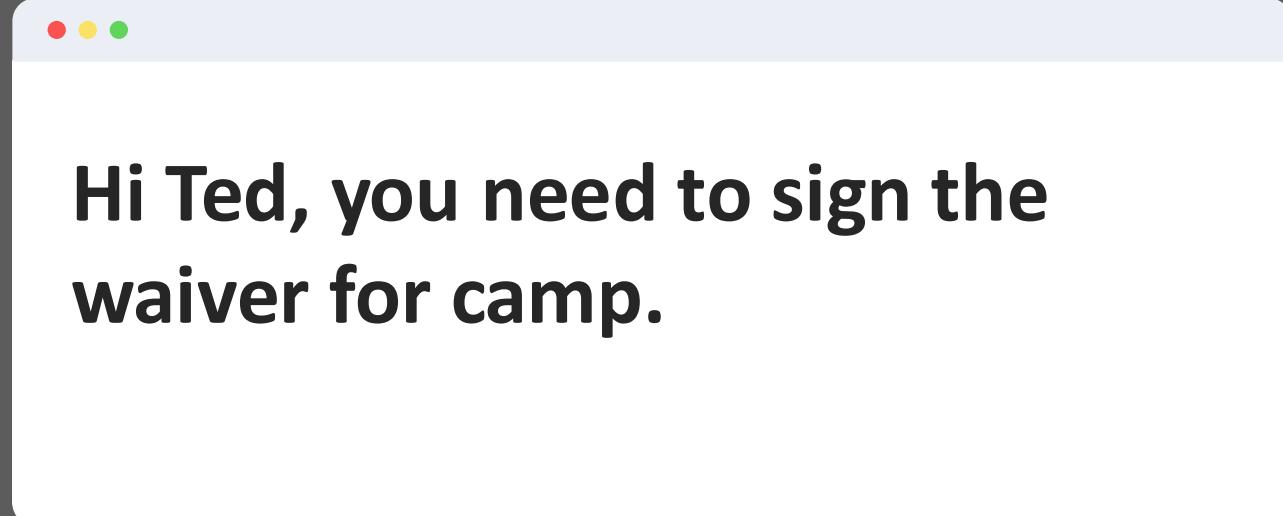


# HasSignedDocument (cont'd)

```
1 Hi {{ CurrentPerson.NickName }}, you  
2 {{ CurrentPerson | HasSignedDocument:1,'have signed','need to sign' }}  
3 the waiver for camp.
```

## Additional Options

1. Signature Document Type Id.
2. Text to show if “true” (signed).
3. Text to show if “false” (not signed).



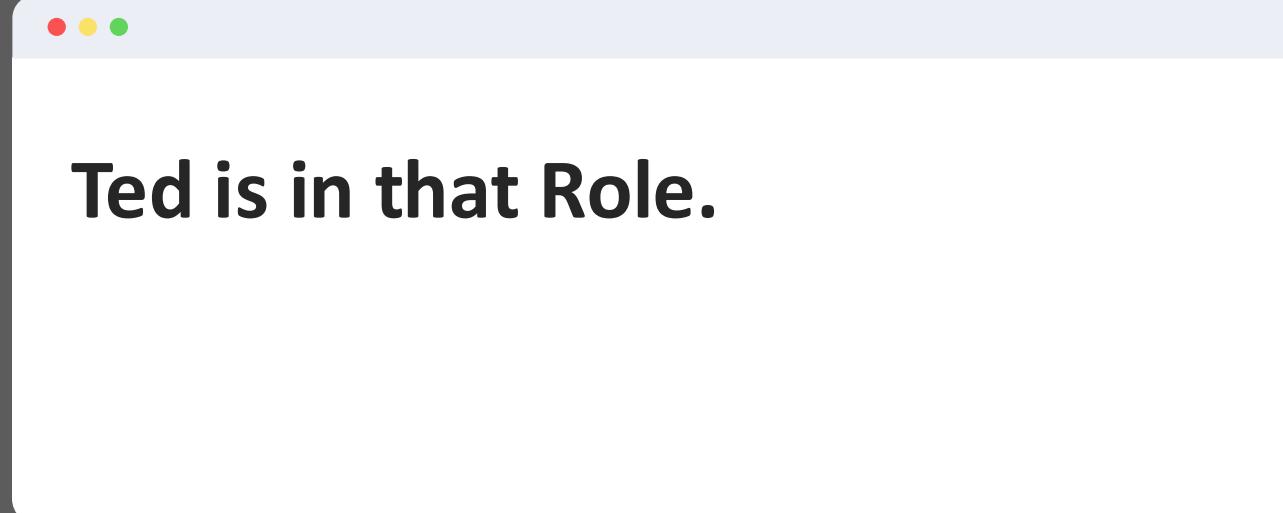


# IsInSecurityRole

```
1  {% assign isInRole = CurrentPerson | IsInSecurityRole:18 %}  
2  {% if isInRole %}  
3      {{ CurrentPerson.FullName }} is in that Role.  
4  {% endif %}
```

## Additional Options

1. **Group Id:** The Id of the specific security role you are interested in.



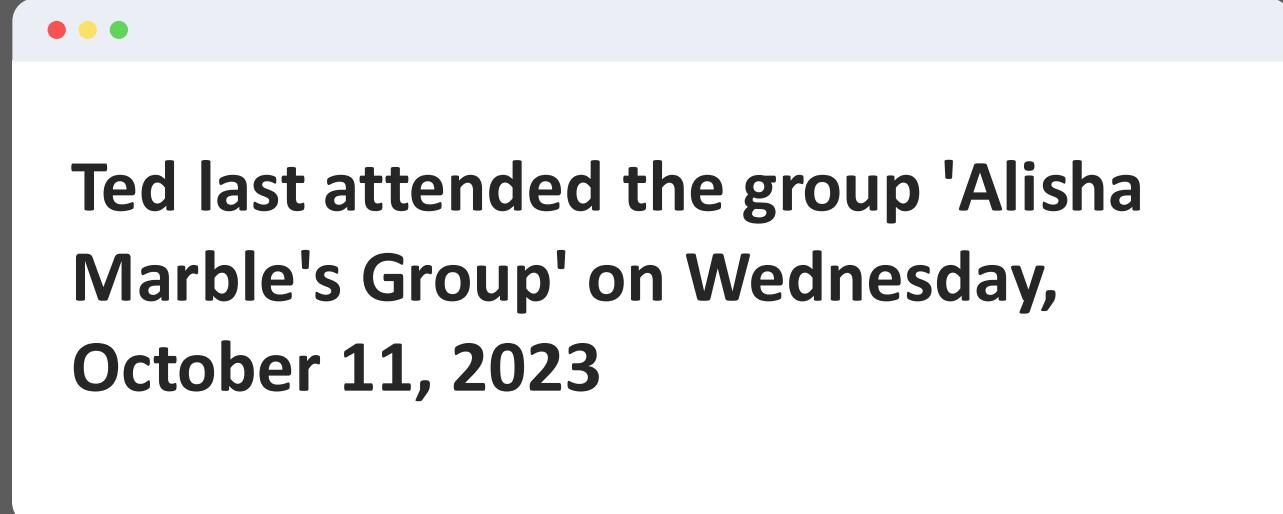


# LastAttendedGroupOfType

```
1  {% assign lastAttended = CurrentPerson | LastAttendedGroupOfType:'25' %}  
2  
3  {{ CurrentPerson.NickName }} last attended the group  
4  '{{ lastAttended.Occurrence.Group.Name }}' on  
5  {{ lastAttended.StartDateTime | Date:'dddd, MMMM d, yyyy' }}.
```

## Additional Options

1. **Group Type Id:** The Id of the Group Type you are interested in.



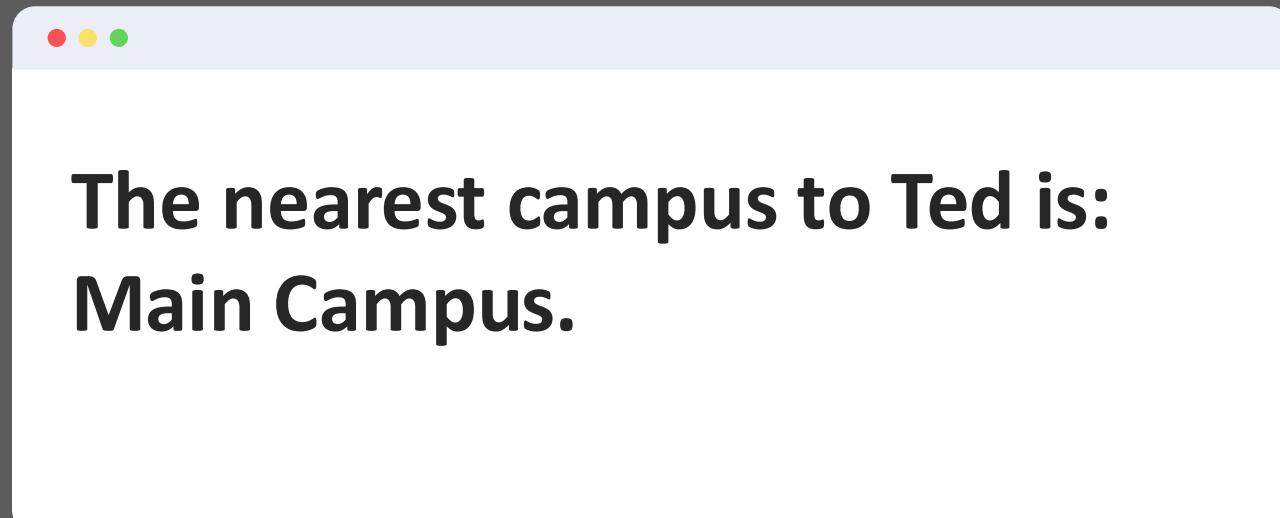


# NearestCampus

```
1  {% assign campus = CurrentPerson | NearestCampus %}  
2  The nearest campus to {{ CurrentPerson.NickName }}  
3  is: {{ campus.Name }}.
```

## Additional Options

- Result Count:** The maximum number of campuses to return.  
[Default: 1]



The nearest campus to Ted is:  
**Main Campus.**

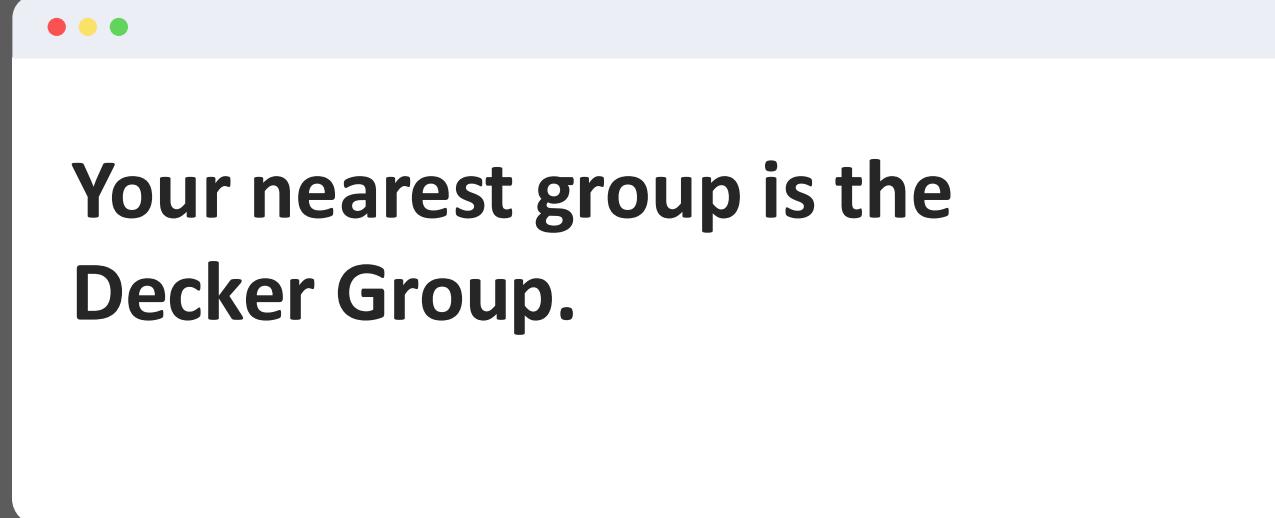


# NearestGroup

```
1  {% assign nearestGroup = CurrentPerson | NearestGroup:'25' %}  
2  
3 Your nearest group is the {{ nearestGroup.Name }}.
```

## Additional Options

1. **Group Type Id:** The Id of the Group Type you are interested in.



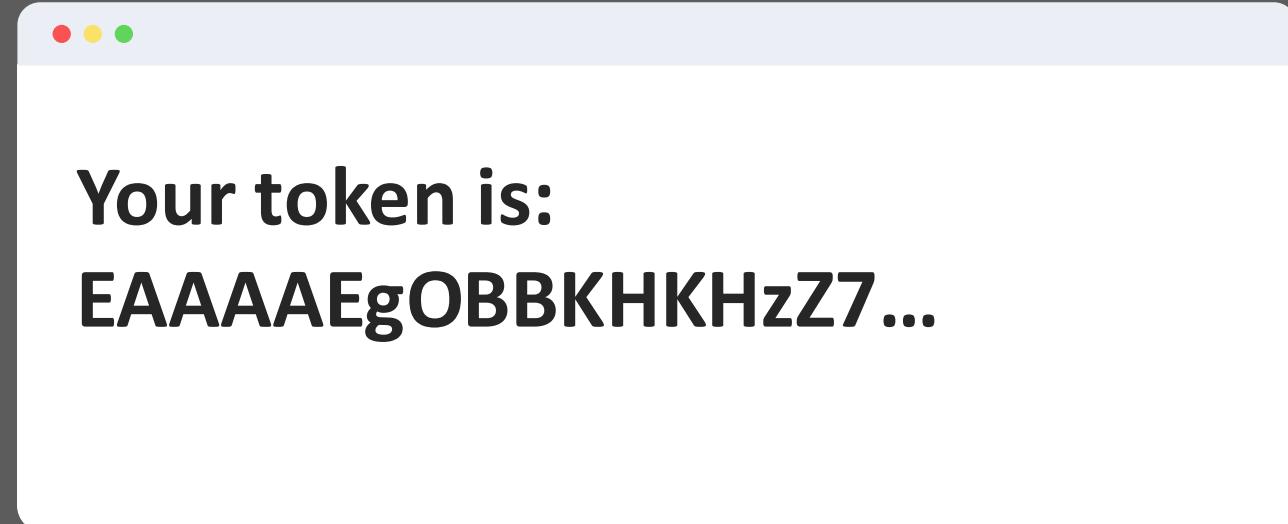


# PersonTokenCreate

```
1 Your token is:  
2 {{ Person | PersonTokenCreate }}
```

## Additional Options

- 1. Minutes:** The number of minutes the token is good for.
- 2. Max Usage:** The number of times the token can be used.
- 3. PageId:** Optional Id to restrict the token to.



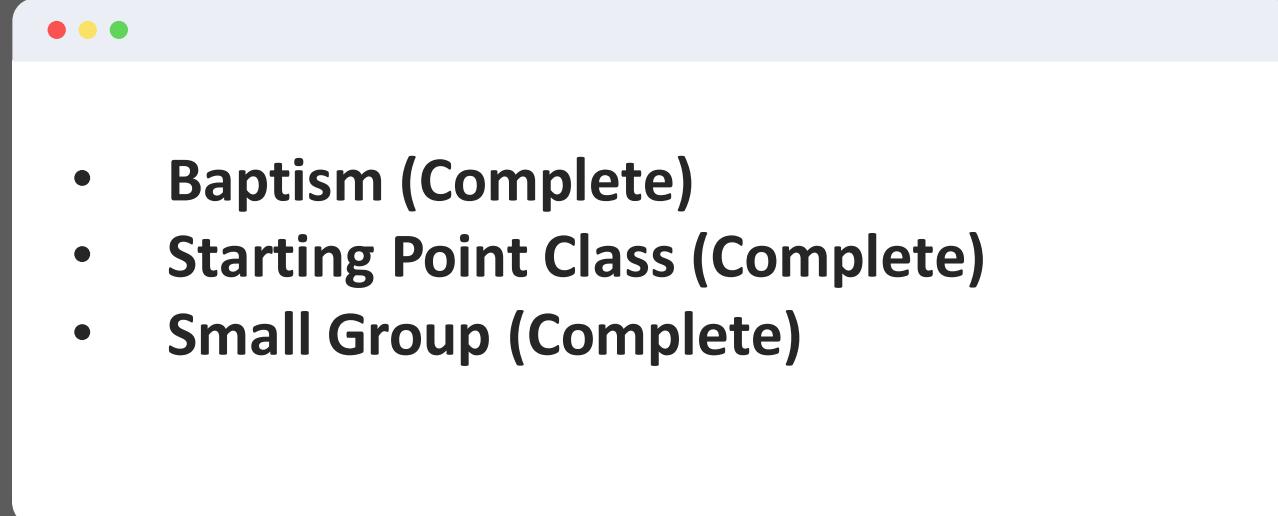


# Steps

```
1  {% assign steps = CurrentPerson | Steps:'1','Complete' %}  
2  <ul>  
3  {% for step in steps %}  
4      <li>{{ step.StepType.Name }} ({{ step.StepStatus.Name }})</li>  
5  {% endfor %}  
6  </ul>
```

## Additional Options

1. Step Program: Id, Guid, or 'All'
2. Step Status: Id, Guid, or 'All'
3. Step Type: Id, Guid, or 'All'





# Other Person Filters

- GeofencingGroupMembers
- GeofencingGroups
- AddSegment
- GetUserPreference
- SetUserPreference
- DeleteUserPreference
- ZebraPhoto

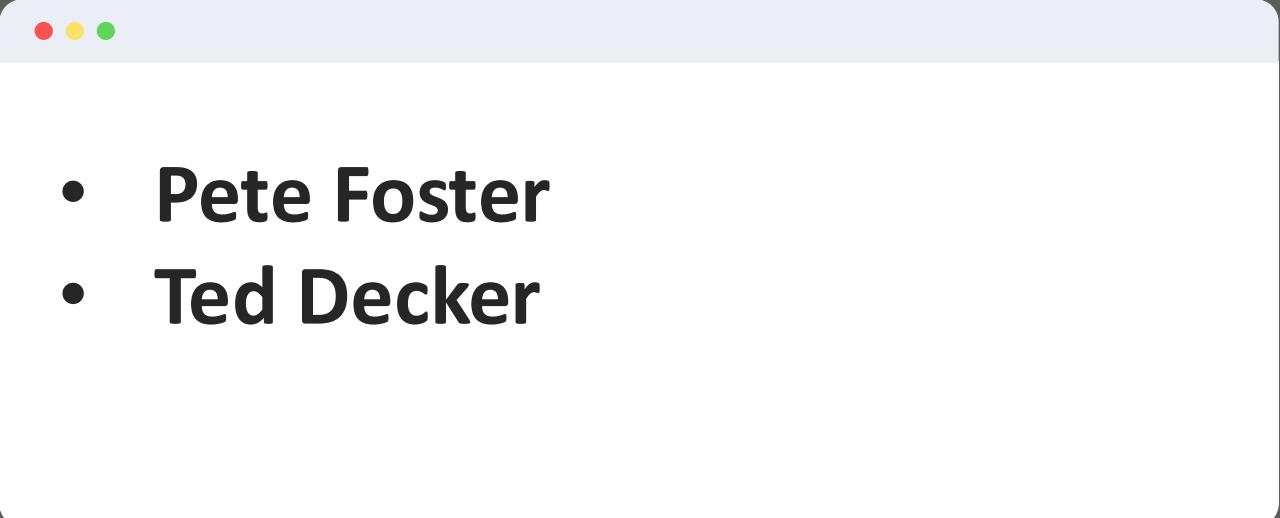


# GeofencingGroupMembers

```
1  {% assign areaLeaders = CurrentPerson | GeofencingGroupMembers:'24','44' %}  
2  <ul>  
3  {% for leader in areaLeaders %}  
4      <li>{{ leader.FullName }}</li>  
5  {% endfor %}  
6  </ul>
```

## Additional Options

1. **Group Type Id:** The group type id to filter on.
2. **Group Role Id:** The group role to filter on.

- 
- The screenshot shows a window with a title bar containing three colored dots (red, yellow, green). The main content area displays two bullet points: "Pete Foster" and "Ted Decker".
- Pete Foster
  - Ted Decker



# GeofencingGroups

```
1  {% assign areas = CurrentPerson | GeofencingGroups:'24' %}  
2  <ul>  
3  {% for area in areas %}  
4      <li>{{ area.Name }}</li>  
5  {% endfor %}  
6  </ul>
```

## Additional Options

1. **Group Type Id:** The group type id to filter on.





# Array Filters





# Types of Arrays

## Primitive Array

```
[  
  "Bear",  
  "Dog",  
  "Rhino"  
]
```

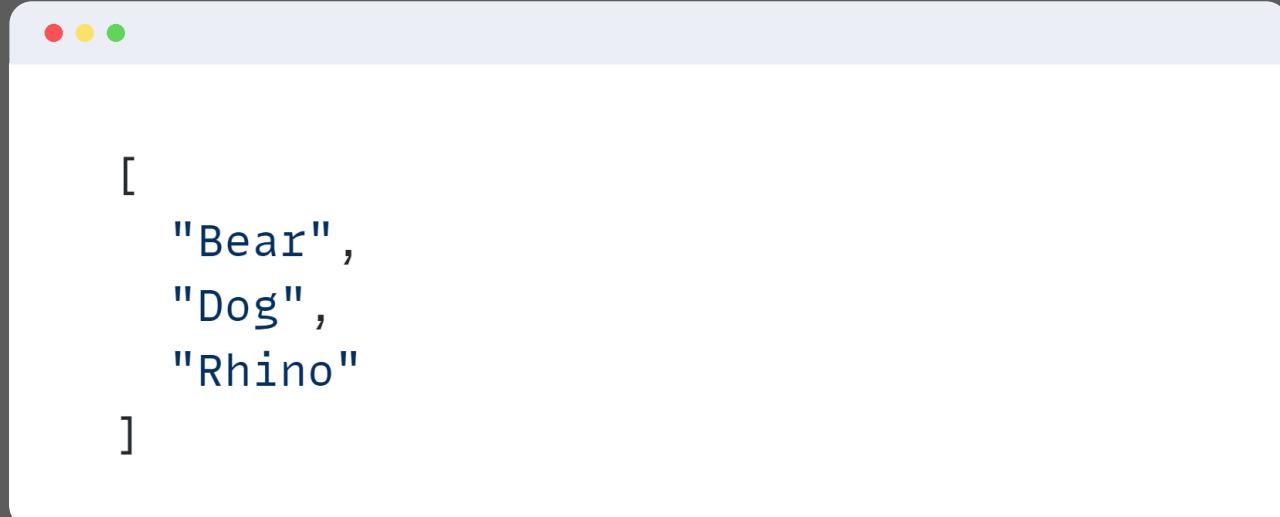
## Object Array

```
[  
  {  
    "Name": "Bear"  
  },  
  {  
    "Name": "Dog"  
  },  
  {  
    "Name": "Rhino"  
  }  
]
```



# Primitive Array

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  
3  <pre>{{ list | ToJSON }}</pre>
```



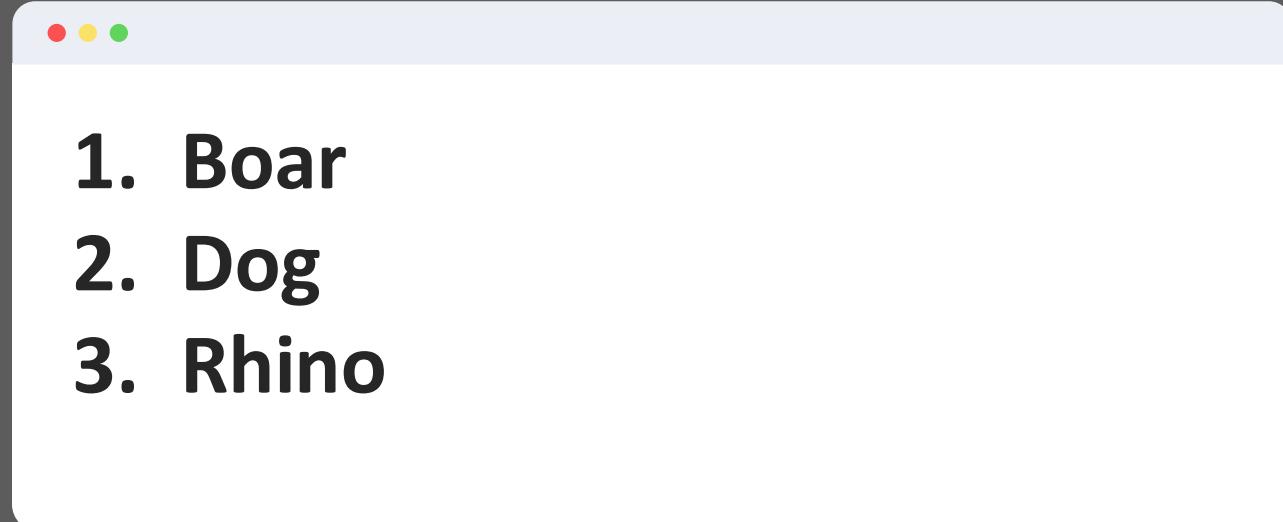
A screenshot of a macOS application window, likely a terminal or code editor, showing the output of the provided Liquid template. The window has a white background and a standard OS X title bar with red, yellow, and green buttons. The main content area displays a JSON array:

```
[  
  "Bear",  
  "Dog",  
  "Rhino"  
]
```



# Split

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  <ol>  
3      {% for item in list %}  
4          <li>{{ item }}</li>  
5      {% endfor %}  
6  </ol>
```



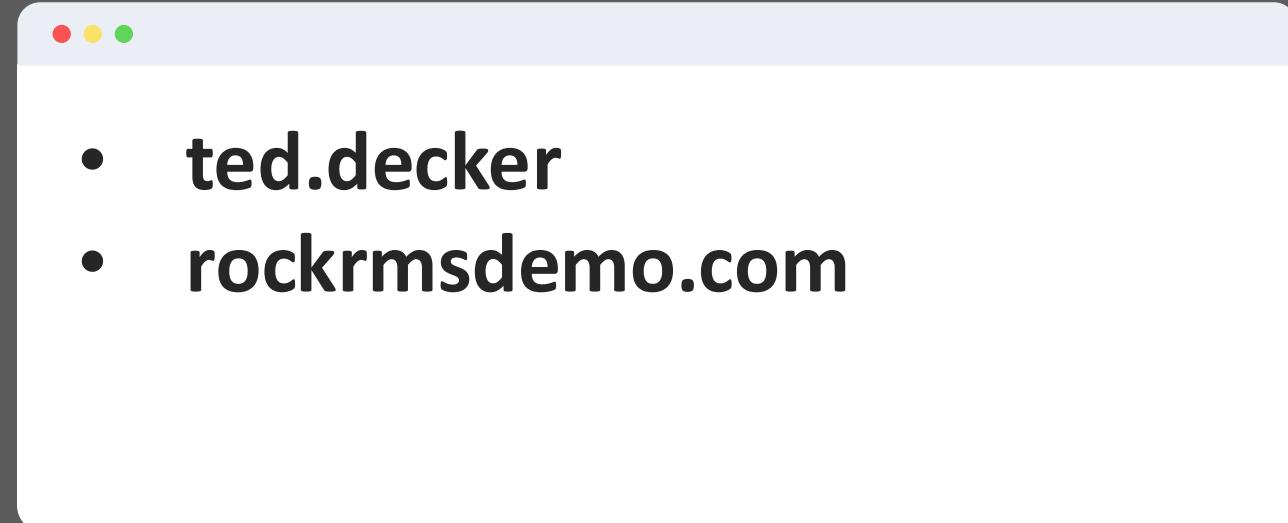


# Split

```
1  {% assign emailParts = CurrentPerson.Email | Split:'@' %}  
2  <ul>  
3    {% for part in emailParts %}  
4      <li>{{ part }}</li>  
5    {% endfor %}  
6  </ul>
```

## Additional Options

1. **Pattern:** The text to split at.
2. **Remove Empty:** Whether to delete “empty” elements.
3. **Maximum:** The maximum number of elements to return.



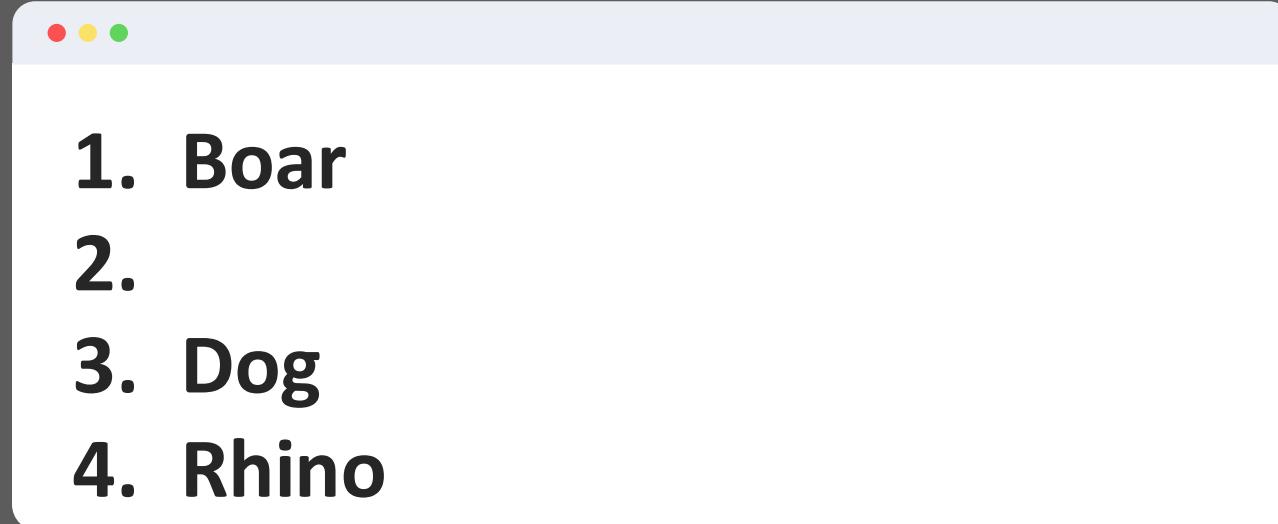


# Split

```
1  {% assign list = 'Boar,,Dog,Rhino' | Split:',', false %}  
2  <ol>  
3    {% for item in list %}  
4      <li>{{ item }}</li>  
5    {% endfor %}  
6  </ol>
```

## Additional Options

1. **Pattern:** The text to split at.
2. **Remove Empty:** Whether to delete “empty” elements.
3. **Maximum:** The maximum number of elements to return.



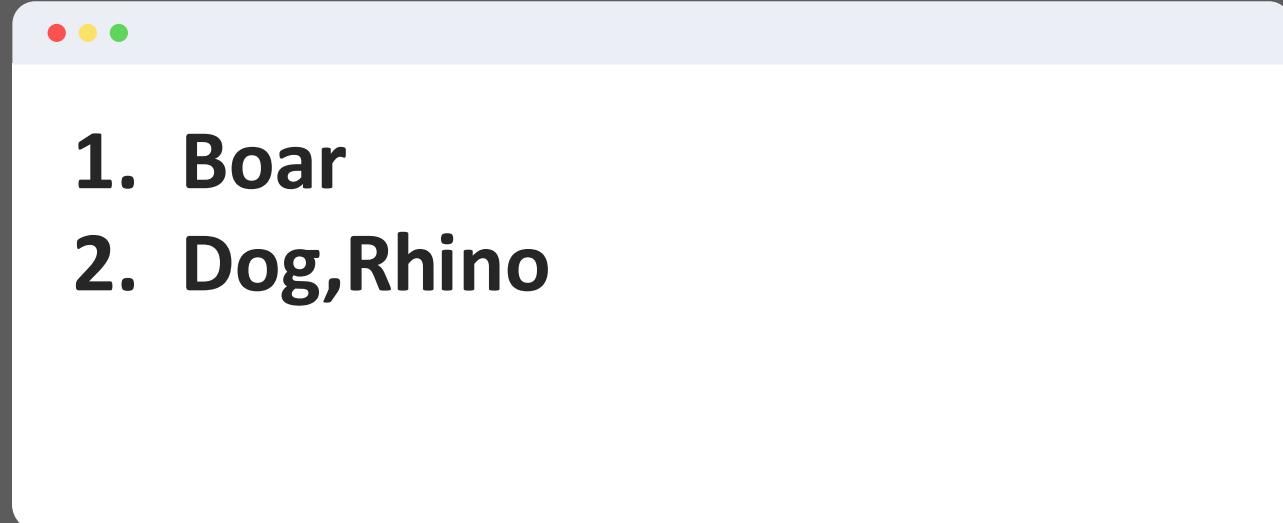


# Split

```
1  {% assign list = 'Boar,Dog,Rhino' | Split: ',', true, 2 %}  
2  <ol>  
3    {% for item in list %}  
4      <li>{{ item }}</li>  
5    {% endfor %}  
6  </ol>
```

## Additional Options

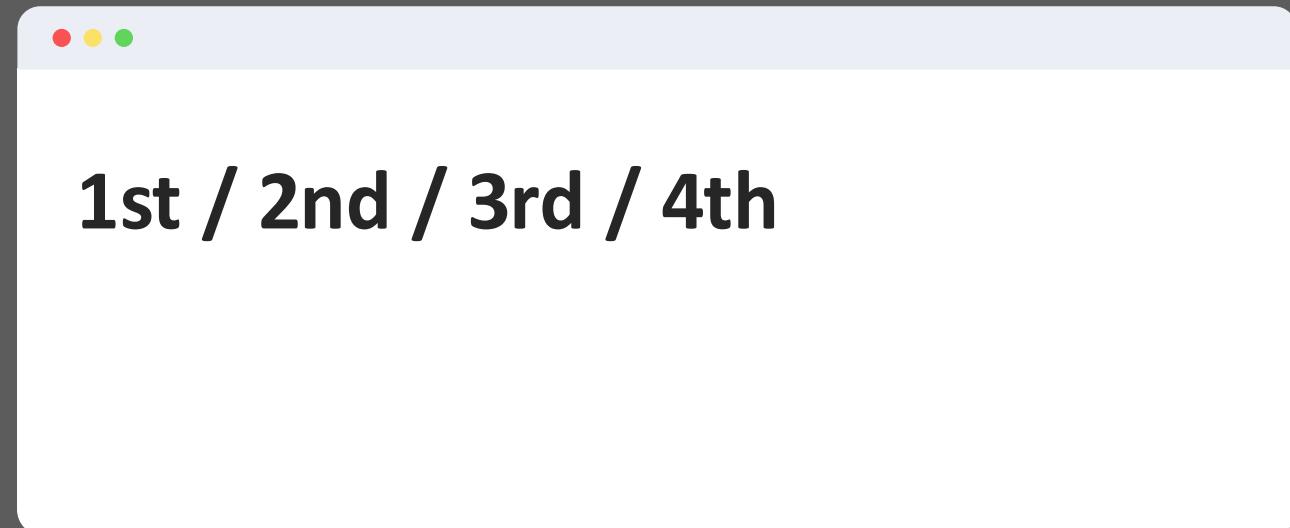
1. **Pattern:** The text to split at.
2. **Remove Empty:** Whether to delete “empty” elements.
3. **Maximum:** The maximum number of elements to return.





# Join

```
1  {% assign list = '1st,2nd,3rd,4th' | Split:',' %}  
2  {{ list | Join:' / ' }}
```





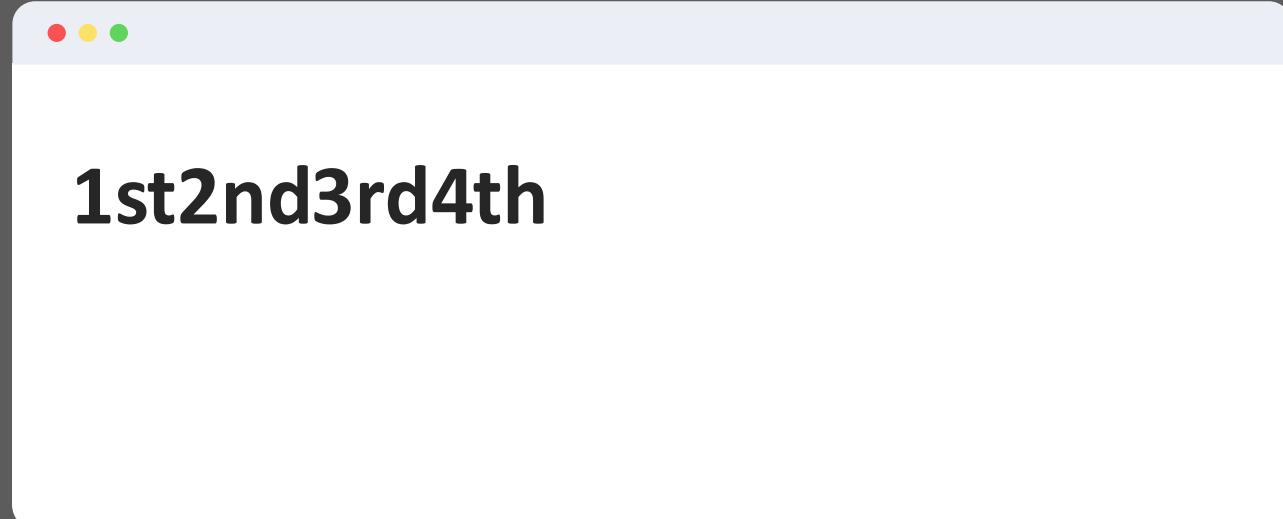
# Join

```
1  {% assign list = '1st,2nd,3rd,4th' | Split:', ' %}  
2  {{ list }}
```



## Helpful Note:

You can directly output a primitive array, but Join allows you to specify a separator.



The screenshot shows a standard Mac OS X application window with a light gray header bar featuring red, yellow, and green buttons. The main content area contains the text "1st2nd3rd4th" in a large, bold, black sans-serif font.



# First

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  {{ list | First }}
```





# Last

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  {{ list | Last }}
```





# Index

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  {{ list | Index:1 }}
```



## Remember:

Arrays are 0-indexed. The first item is #0, the second is #1.





# Indexer

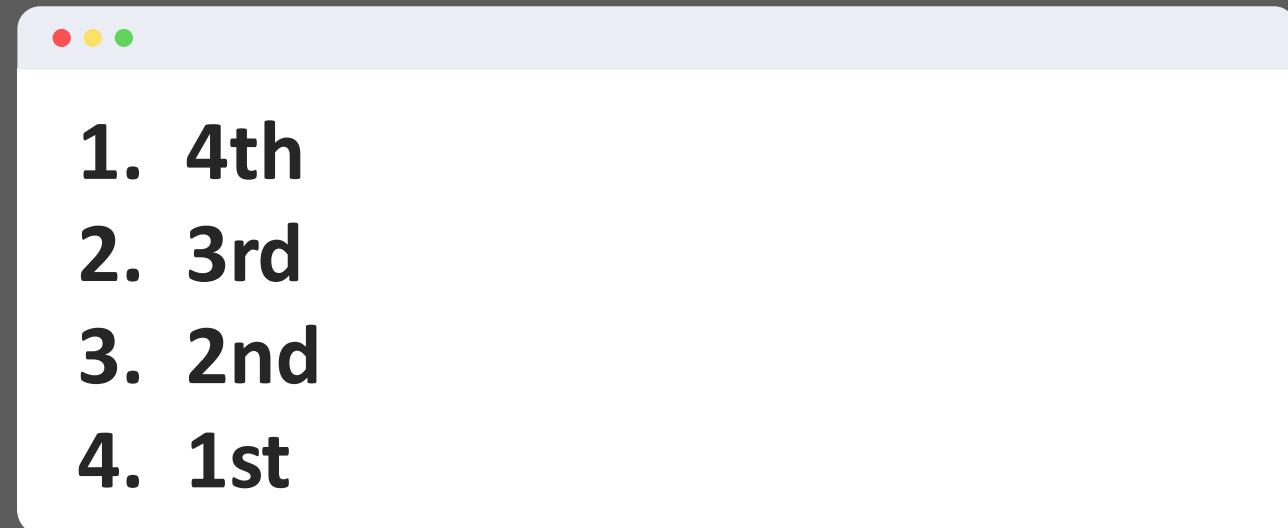
```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  {{ list[0] }}
```





# Reverse

```
1  {% assign list = '1st,2nd,3rd,4th' | Split:',' %}
2  {% assign list = list | Reverse %}
3  <ol>
4  {% for item in list %}
5    <li>{{ item }}</li>
6  {% endfor %}
7  </ol>
```





# Reverse

```
1  {% assign list = '1st,2nd,3rd,4th' | Split:',' | Reverse %}
2  <ol>
3  {% for item in list %}
4    <li>{{ item }}</li>
5  {% endfor %}
6  </ol>
```

You're getting it!



Yes, this does the same thing as the previous slide. Use whichever is most readable.

- 
1. 4th  
2. 3rd  
3. 2nd  
4. 1st



# Size

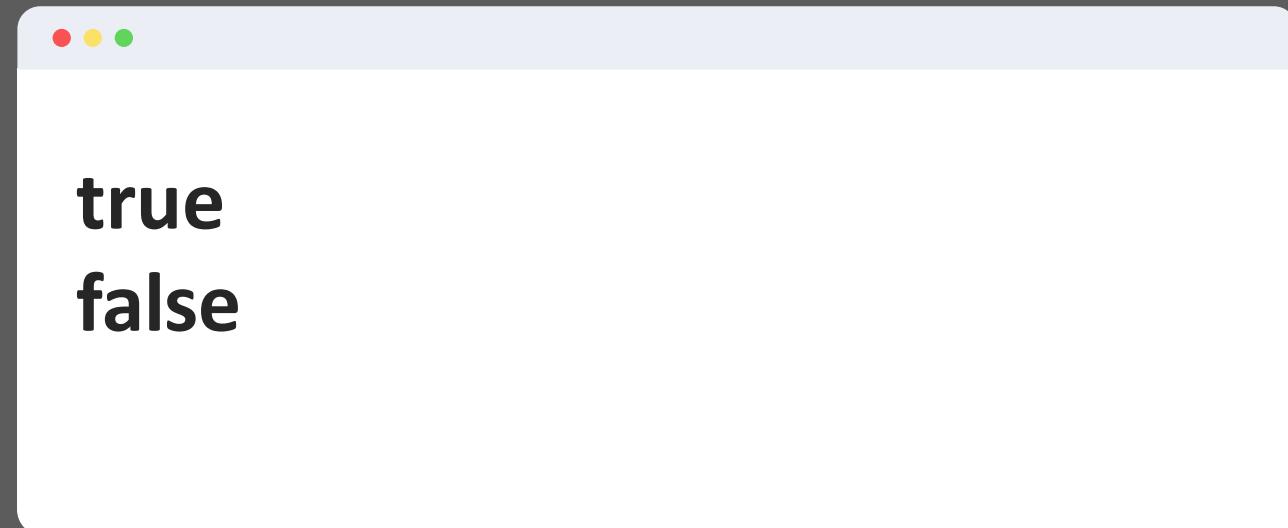
```
1  {% assign list = '1st,2nd,3rd,4th' | Split:',' %}  
2  {{ list | Size }}
```





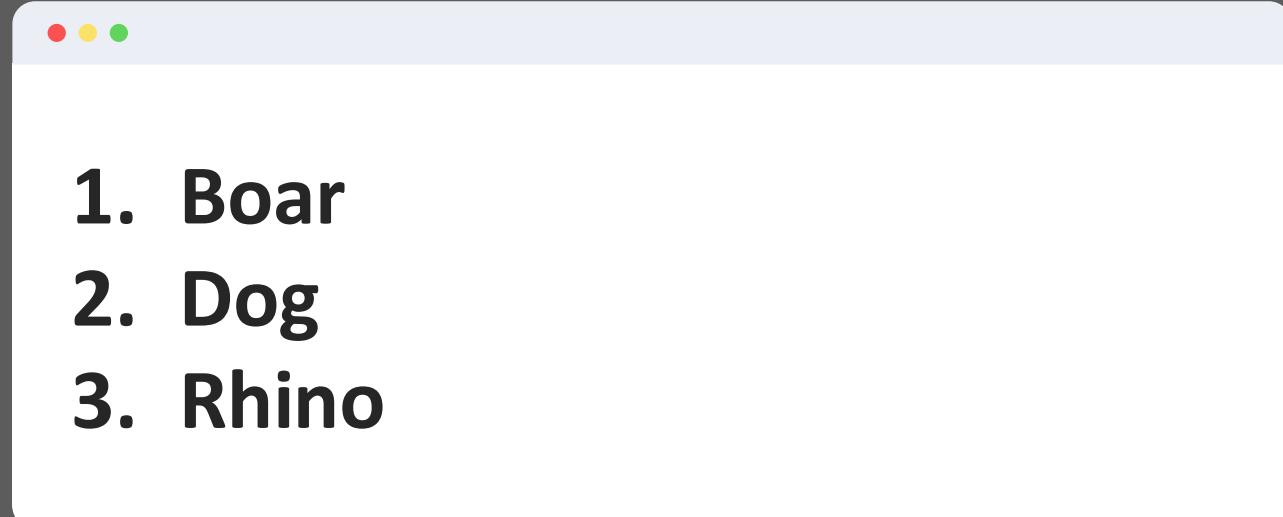
# Contains

```
1  {% assign list = 'Boar,Dog,Rhino' | Split:', ' %}  
2  {{ list | Contains:'Dog' }}<br>  
3  {{ list | Contains:'Rabbit' }}
```





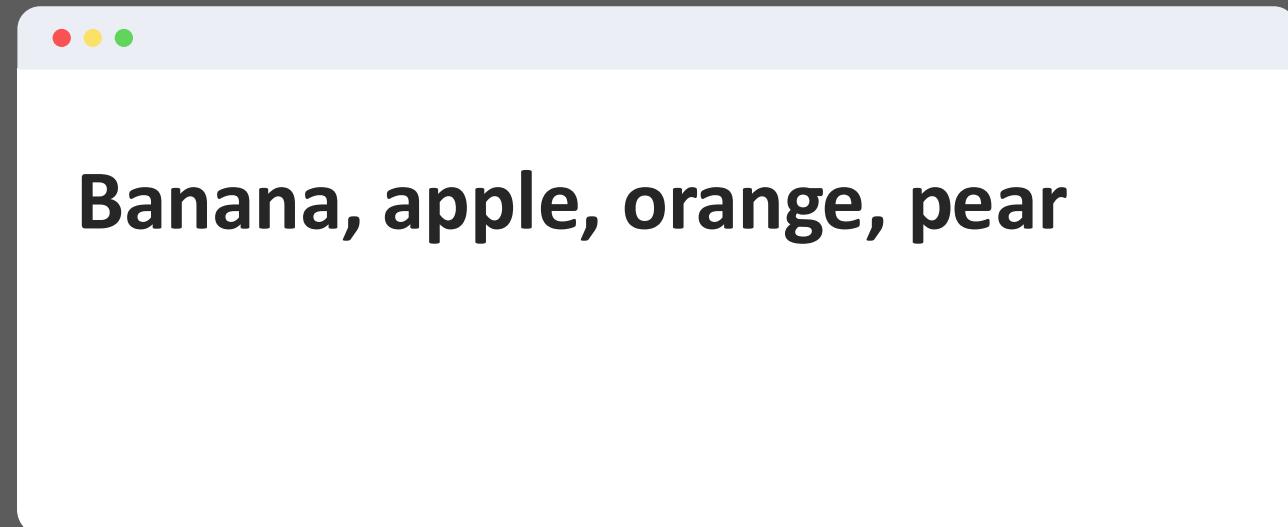
```
1  {% assign list = 'Boar,Dog,Boar,Rhino,Rhino' | Split:',' | Uniq %}
2  <ol>
3  {% for item in list %}
4    <li>{{ item }}</li>
5  {% endfor %}
6  </ol>
```





# Sort

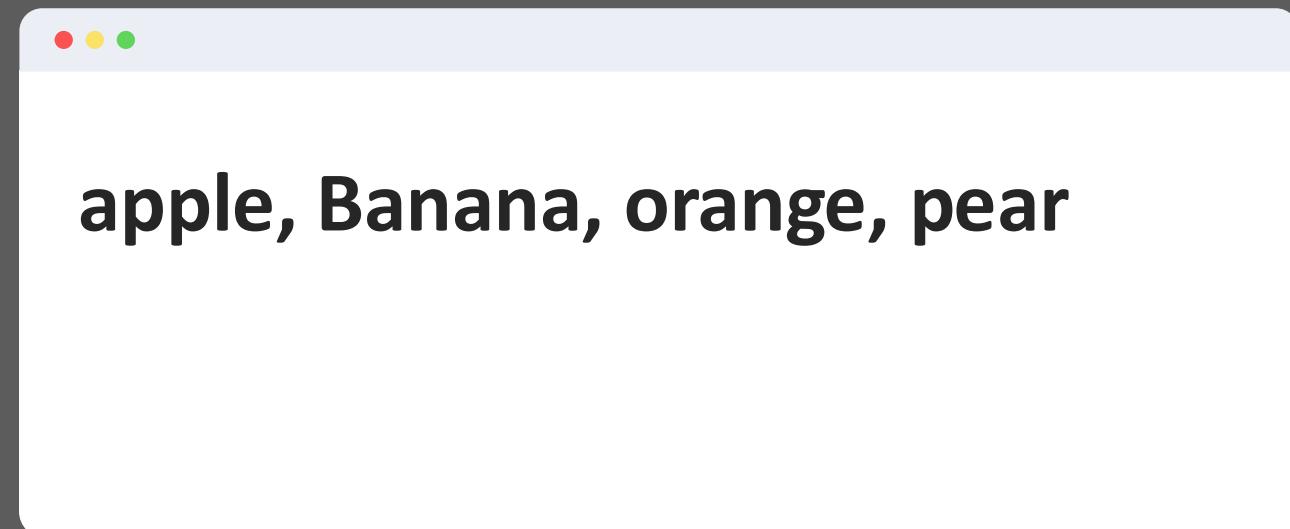
```
1  {% assign fruits = 'Banana,orange,apple,pear' | Split:', ' %}
2  {{ fruits | Sort | Join:', ' }}
```





# SortNatural

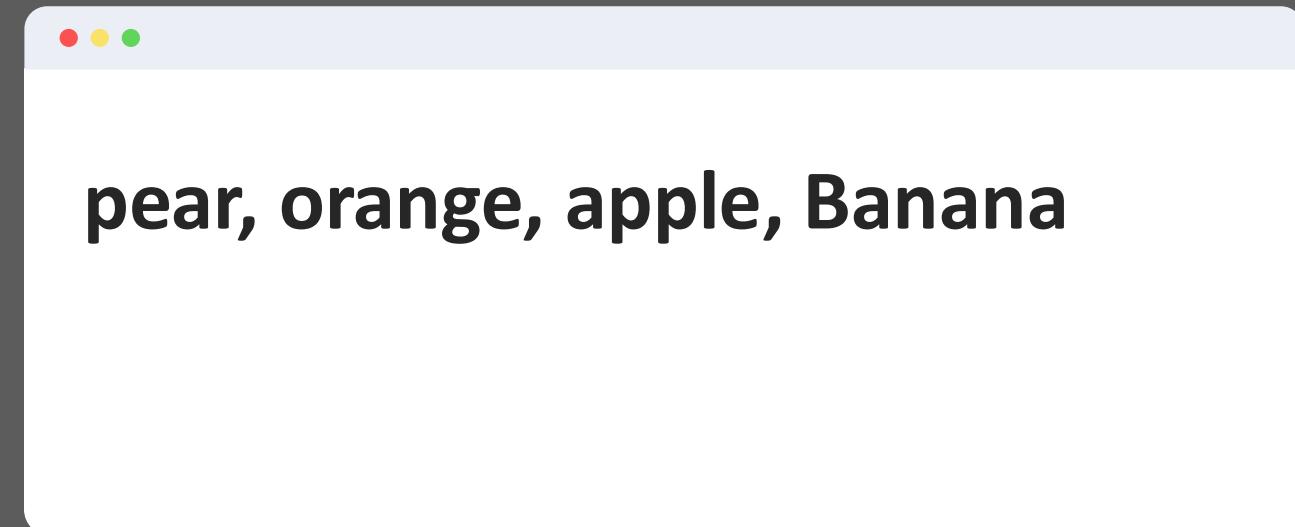
```
1  {% assign fruits = 'Banana,orange,apple,pear' | Split:', ' %}
2  {{ fruits | SortNatural | Join:', ' }}
```





# Shuffle

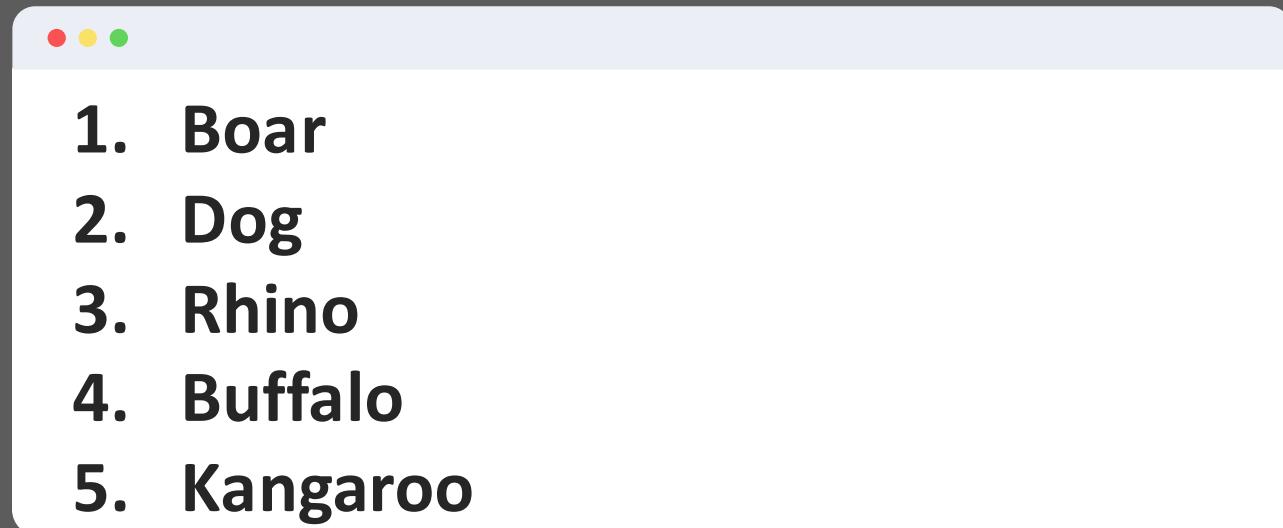
```
1  {% assign fruits = 'Banana,orange,apple,pear' | Split:',' %}  
2  {{ fruits | Shuffle | Join:', ' }}
```





# AddToArray

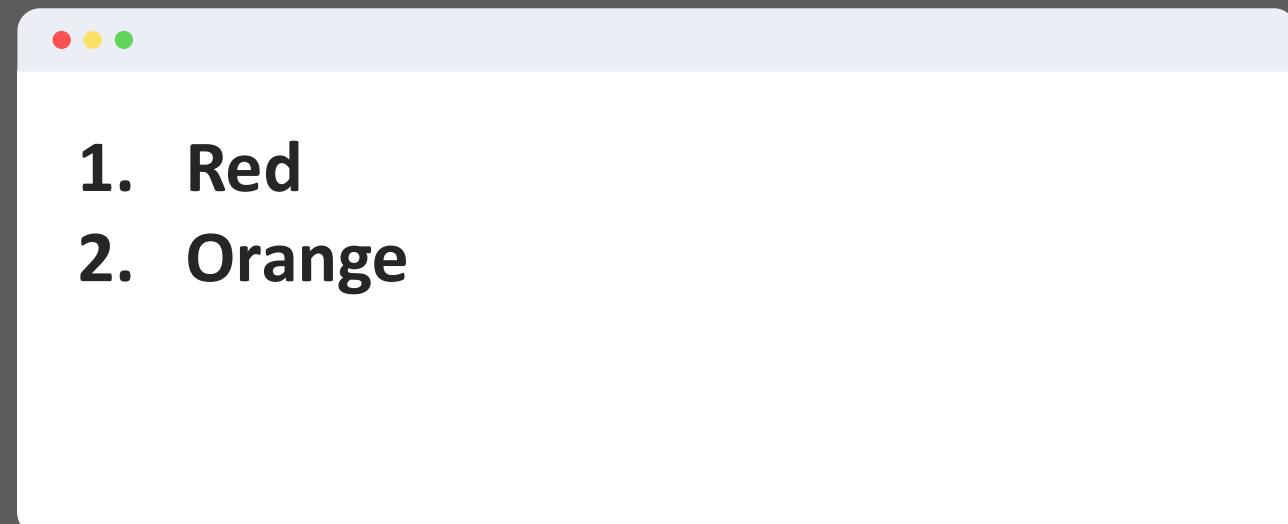
```
1  {% assign list = 'Boar,Dog,Rhino' | Split:',' %}  
2  {% assign list = list | AddToArray:'Buffalo' %}  
3  {% assign list = list | AddToArray:'Kangaroo' %}  
4  <ol>  
5      {% for item in list %}  
6          <li>{{ item }}</li>  
7      {% endfor %}  
8  </ol>
```





# AddToArray

```
1  {% assign list = '' | AddToArray:'Red' | AddToArray:'Orange' %}  
2  <ol>  
3      {% for item in list %}  
4          <li>{{ item }}</li>  
5      {% endfor %}  
6  </ol>
```





# RemoveFromArray

```
1  {% assign fruits = 'Banana,orange,apple,pear' | Split:',' %}  
2  {% assign fruits = fruits | RemoveFromArray:'pear' %}  
3  <ul>  
4  {% for fruit in fruits %}  
5      <li>{{ fruit }}</li>  
6  {% endfor %}  
7  </ul>
```





# Types of Arrays

## Primitive Array

```
[  
  "Bear",  
  "Dog",  
  "Rhino"  
]
```

## Object Array

```
[  
  {  
    "Name": "Bear"  
  },  
  {  
    "Name": "Dog"  
  },  
  {  
    "Name": "Rhino"  
  }  
]
```



# Object Array

```
1  {% assign kids = CurrentPerson | Children %}  
2  <pre>{{ kids | ToJSON }}</pre>
```



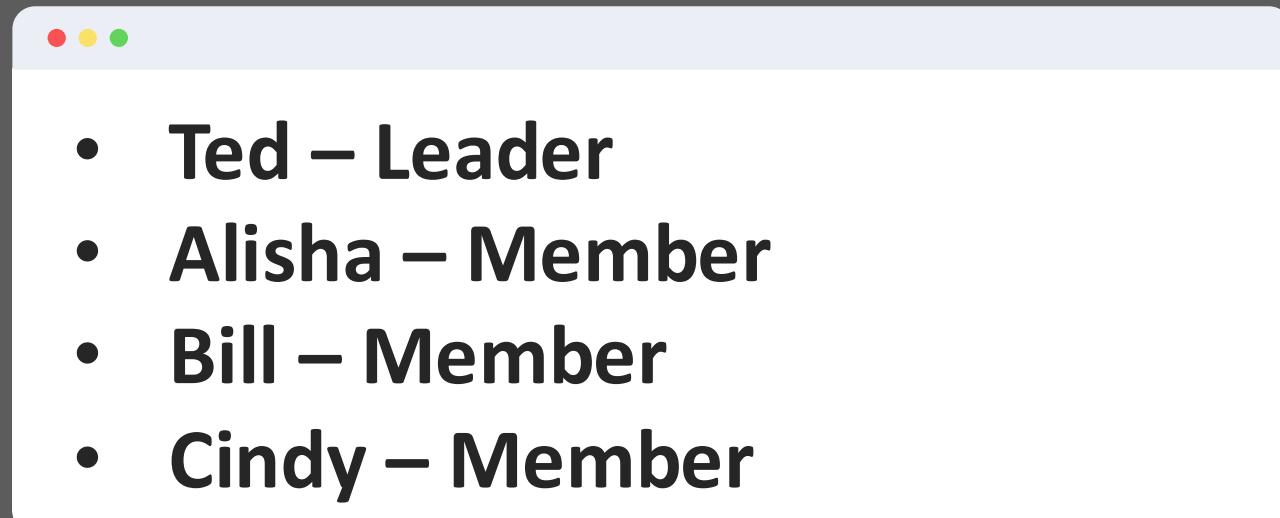
A screenshot of a Mac OS X terminal window. The window has a white background and a light gray border. In the top-left corner, there are three colored window control buttons: red, yellow, and green. The main content area displays a JSON object array. The array starts with a left square bracket [ and contains two objects. Each object is enclosed in a pair of curly braces {}, has an "Id" key with the value 7 and 8 respectively, and an "FirstName" key with the values "Noah" and "Alexis". Both objects also have an "LastName" key with the value "Decker". Each object ends with a red [...] placeholder for additional items. The entire array concludes with a right square bracket ].

```
[  
  {  
    "Id": 7,  
    "FirstName": "Noah",  
    "LastName": "Decker",  
    [...]  
  },  
  {  
    "Id": 8,  
    "FirstName": "Alexis",  
    "LastName": "Decker",  
    [...]  
  }  
]
```



# OrderBy

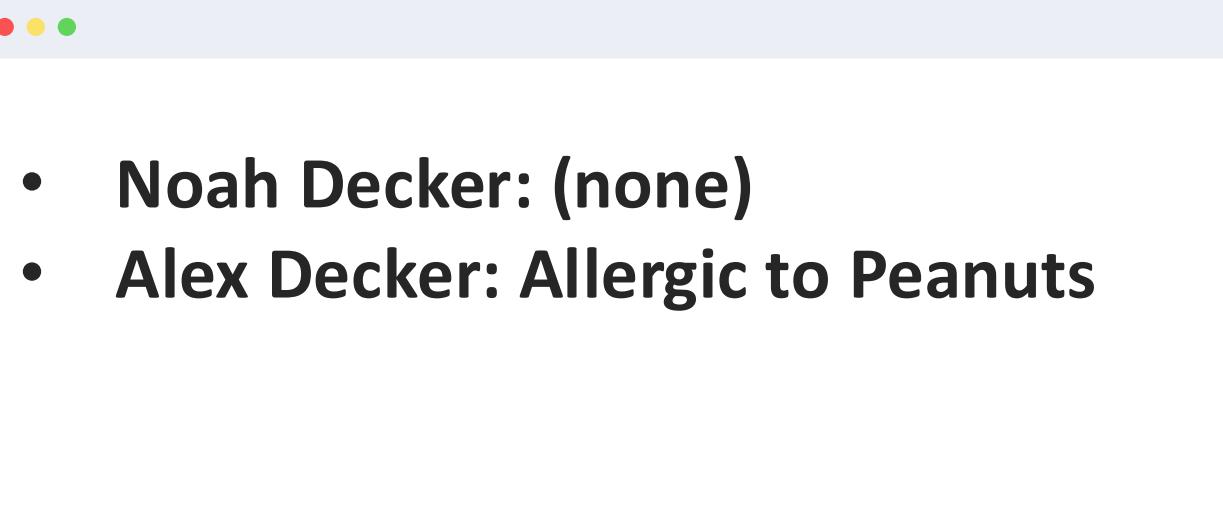
```
1  {% assign members = Members | OrderBy:'GroupRole.IsLeader desc,Person.FirstName' %}  
2  <ul>  
3  {% for member in members %}  
4      <li>{{ member.Person.FirstName }} - {{ member.GroupRole.Name }}</li>  
5  {% endfor %}  
6  </ul>
```





# SortByAttribute

```
1  {% assign children = CurrentPerson | Children | SortByAttribute:'Allergy' %}  
2  <ul>  
3  {% for child in children %}  
4      <li>  
5          {{ child.FullName }}:  
6          {{ child | Attribute:'Allergy' | Default:'(none)' }}  
7      </li>  
8  {% endfor %}  
9  </ul>
```





# Where

```
1  {% assign girls = CurrentPerson | Children | Where:'Gender', 'Female' %}  
2  <ul>  
3      {% for child in girls %}  
4          <li>{{ child.FullName }}</li>  
5      {% endfor %}  
6  </ul>
```

## Additional Options

- Property:** The property key to match on.
- Value:** The value to match on.

- Alex Decker



# Distinct

```
1  {% assign kids = CurrentPerson | Children %}  
2  {% assign statusReps = kids | Distinct:'ConnectionStatusValueId' %}  
3  <ol>  
4    {% for record in statusReps %}  
5      <li>  
6        {{ record.ConnectionStatusValue.Value }}  
7        ({{ record.NickName }}), for example).  
8      </li>  
9    {% endfor %}  
10   </ol>
```

```
[  
 {  
   "Id": 7,  
   "FirstName": "Noah",  
   "LastName": "Decker",  
   "ConnectionStatusValueId": 146,  
   [...]  
 },  
 {  
   "Id": 8,  
   "FirstName": "Alexis",  
   "LastName": "Decker",  
   "ConnectionStatusValueId": 146,  
   [...]  
 }  
 ]
```



# Distinct

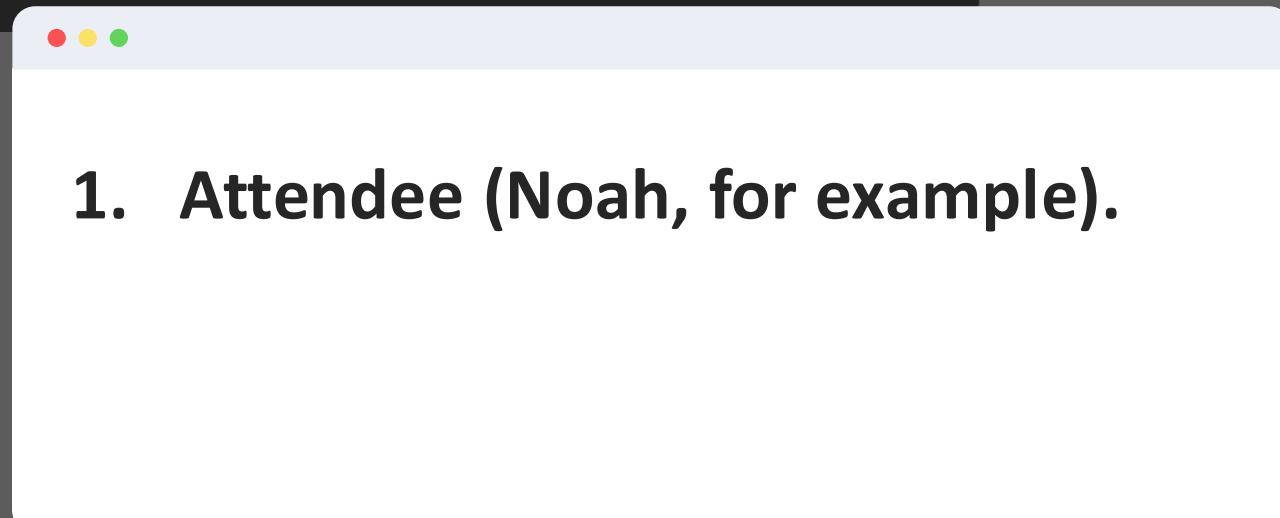
```
1  {% assign kids = CurrentPerson | Children %}  
2  {% assign statusReps = kids | Distinct:'ConnectionStatusValueId' %}  
3  <ol>  
4    {% for record in statusReps %}  
5      <li>  
6        {{ record.ConnectionStatusValue.Value }}  
7        ({{ record.NickName }}, for example).  
8      </li>  
9    {% endfor %}  
10   </ol>
```

```
[  
 {  
   "Id": 7,  
   "FirstName": "Noah",  
   "LastName": "Decker",  
   "ConnectionStatusValueId": 146,  
   "ConnectionStatus": {  
     "Id": 146,  
     "Value": "Attendee"  
   },  
   [...]  
 }  
 ]
```



# Distinct

```
1  {% assign kids = CurrentPerson | Children %}  
2  {% assign statusReps = kids | Distinct:'ConnectionStatusValueId' %}  
3  <ol>  
4    {% for record in statusReps %}  
5      <li>  
6        {{ record.ConnectionStatusValue.Value }}  
7        ({{ record.NickName }}, for example).  
8      </li>  
9    {% endfor %}  
10   </ol>
```





# Map

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Members %}  
3  <pre>{{ members | ToJSON }}</pre>
```

```
[  
 {  
   "Id": 90,  
   "GroupRoleId": 24,  
   "GroupId": 125,  
   "Person": {  
     "Id": 5,  
     "FullName": "Ted Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 91,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 6,  
     "FullName": "Cindy Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 92,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 18,  
     "FullName": "Bill Marble",  
     [...]  
   }  
 },  
 {  
   "Id": 93,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 19,  
     "FullName": "Alisha Marble",  
     [...]  
   }  
 },  
 ]
```



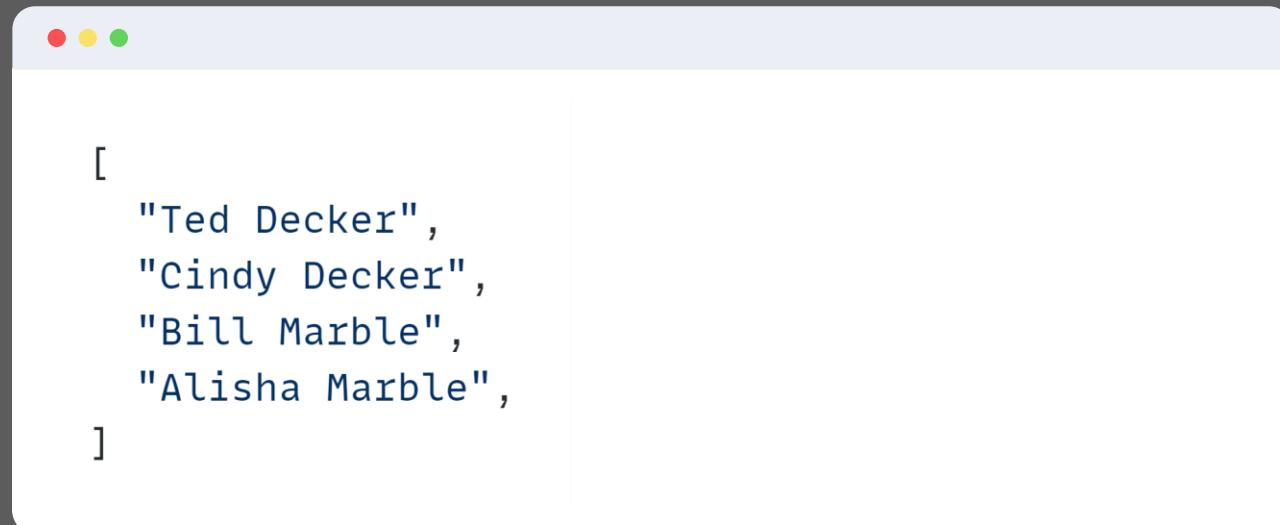
# Map

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Mbers %}  
3  <pre>{{ members | Map:'Person.FullName' | ToJSON }}</pre>
```



## Take Note:

If you map a property that stores a primitive, the Map filter “collapses” the object array to a primitive array.



A screenshot of a Mac OS X application window. The window has a title bar with three colored buttons (red, yellow, green) at the top right. The main content area contains a JSON array represented by square brackets [ and ]. Inside the brackets is a list of four strings: "Ted Decker", "Cindy Decker", "Bill Marble", and "Alisha Marble".

```
[  
  "Ted Decker",  
  "Cindy Decker",  
  "Bill Marble",  
  "Alisha Marble",  
 ]
```



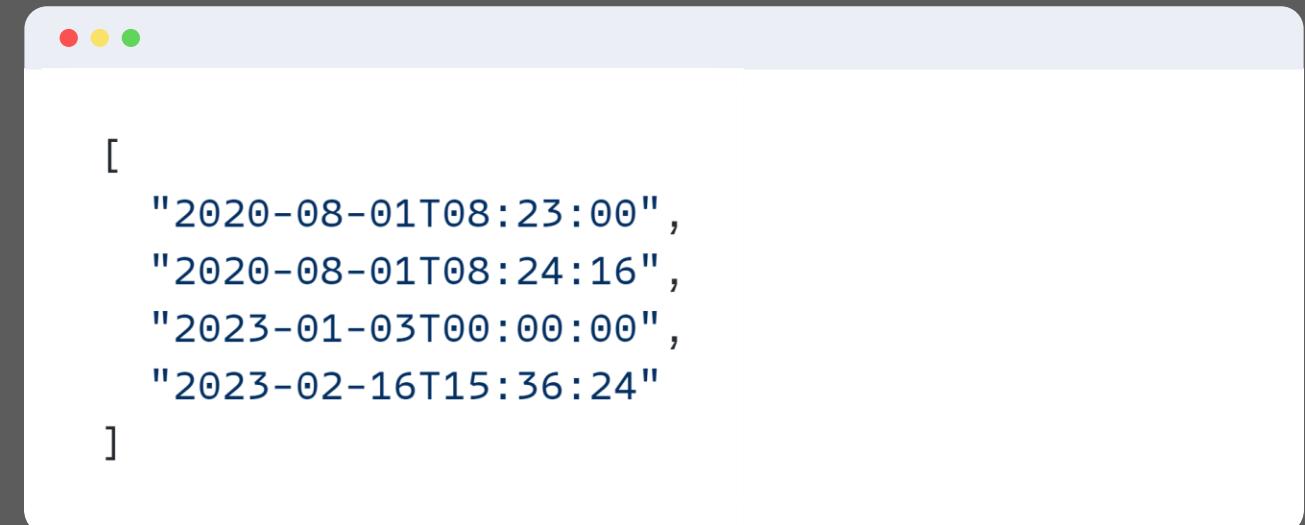
# Select

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Mbers %}  
3  <pre>{{ members | Select:'DateTimeAdded' | ToJSON }}</pre>
```



## Map vs Select:

Select works just like Map does, but you can't use Select if you want a sub-property.



```
[  
  "2020-08-01T08:23:00",  
  "2020-08-01T08:24:16",  
  "2023-01-03T00:00:00",  
  "2023-02-16T15:36:24"  
]
```



# Slice

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Members %}  
3  <pre>{{ members | ToJSON }}</pre>
```

```
[  
 {  
   "Id": 90,  
   "GroupRoleId": 24,  
   "GroupId": 125,  
   "Person": {  
     "Id": 5,  
     "FullName": "Ted Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 91,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 6,  
     "FullName": "Cindy Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 92,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 18,  
     "FullName": "Bill Marble",  
     [...]  
   }  
 },  
 {  
   "Id": 93,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 19,  
     "FullName": "Alisha Marble",  
     [...]  
   }  
 }]
```



# Slice

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Members %}  
3  <pre>{{ members | Slice:1,2 | ToJSON }}</pre>
```

## Settings

1. Index of first item.
2. Number of items to include  
(default=1).

```
[  
 {  
   "Id": 90,  
   "GroupRoleId": 24,  
   "GroupId": 125,  
   "Person": {  
     "Id": 5,  
     "FullName": "Ted Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 91,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 6,  
     "FullName": "Cindy Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 92,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 18,  
     "FullName": "Bill Marble",  
     [...]  
   }  
 },  
 {  
   "Id": 93,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 19,  
     "FullName": "Alisha Marble",  
     [...]  
   }  
 }]
```



# Slice

```
1  {% assign deckerSmallGroup = 125 | GroupById %}  
2  {% assign members = deckerSmallGroup.Members %}  
3  <pre>{{ members | Slice:1,2 | ToJSON }}</pre>
```

## Settings

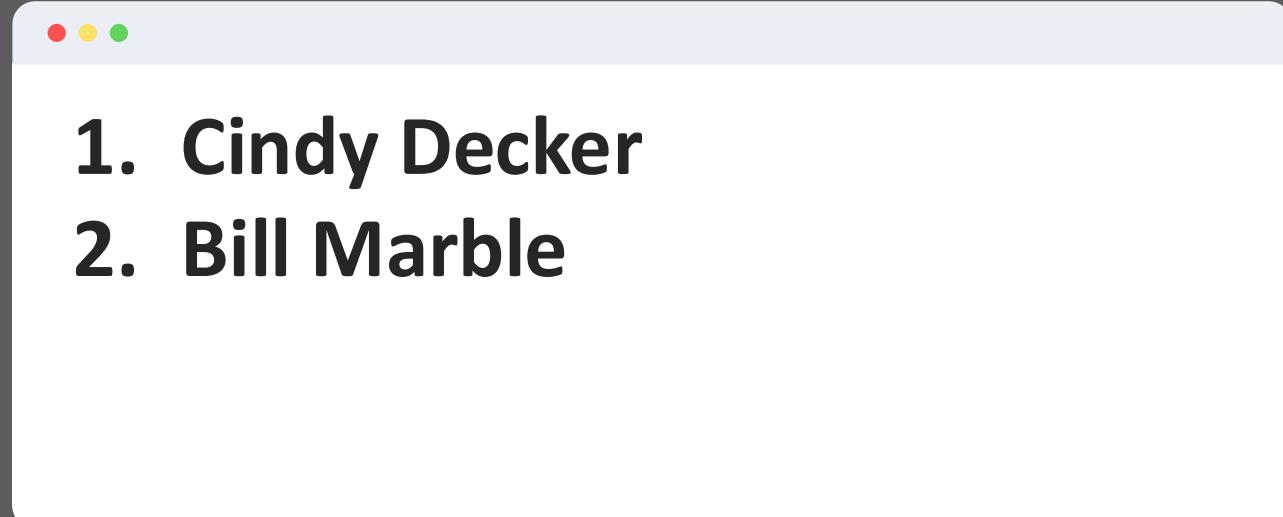
1. Index of first item.
2. Number of items to include  
(default=1).

```
[  
 {  
   "Id": 91,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 6,  
     "FullName": "Cindy Decker",  
     [...]  
   }  
 },  
 {  
   "Id": 92,  
   "GroupRoleId": 23,  
   "GroupId": 125,  
   "Person": {  
     "Id": 18,  
     "FullName": "Bill Marble",  
     [...]  
   }  
 }]
```



# Slice

```
1  {% assign subList = Members | Slice:1,2 | Map:'Person.FullName' %}  
2  <ol>  
3      {% for member in sublist %}  
4          <li>{{ member }}</li>  
5      {% endfor %}  
6  </ol>
```





# GroupBy

```
1  {% assign groupedMembers = Members | GroupBy:'GroupRole.Name' %}  
2  <ul>  
3      {% for role in groupedMembers %}  
4          {% assign parts = role | PropertyToKeyValue %}  
5          <li>{{ parts.Key }}</li>  
6          <ul>  
7              {% for member in parts.Value %}  
8                  <li>{{ member.Person.FullName }}</li>  
9              {% endfor %}  
10             </ul>  
11     {% endfor %}  
12 </ul>
```

The screenshot shows a window with a title bar containing three colored dots (red, yellow, green). The main content area displays a hierarchical list:

- **Leader**
  - **Ted**
- **Member**
  - **Cindy**
  - **Bill**
  - **Alisha**



# GroupBy

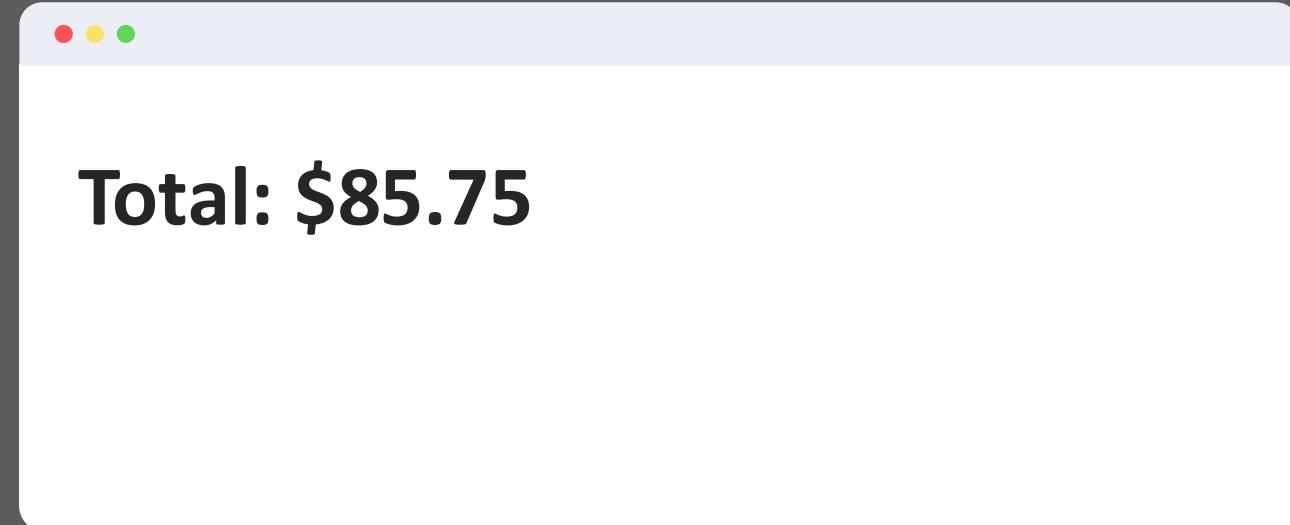
```
1  {% assign groupedMembers = Members | GroupBy:'GroupRole.Name' %}  
2  <ul>  
3    {% for role in groupedMembers %}  
4      {% assign parts = role | PropertyToKeyValue %}  
5      <li>{{ parts.Key }}</li>  
6      <ul>  
7        {% for member in parts.Value %}  
8          <li>{{ member.Person.FullName }}</li>  
9        {% endfor %}  
10       </ul>  
11     {% endfor %}  
12   </ul>
```

```
{
  "Leader": [
    {
      "Id": 90,
      "Person": {
        "Id": 5,
        "FullName": "Ted Decker",
        [...]
      }
    },
    "Member": [
      {
        "Id": 91,
        "Person": {
          "Id": 6,
          "FullName": "Cindy Decker",
          [...]
        }
      },
      {
        "Id": 92,
        "Person": {
          "Id": 18,
          "FullName": "Bill Marble",
          [...]
        }
      },
      {
        "Id": 93,
        "Person": {
          "Id": 17,
          "FullName": "Alisha Marble",
          [...]
        }
      }
    ]
}
```



# Sum

```
1 Total: ${\{ Items | Select:'Price' | Sum \}}
```





# Array Filters and Text

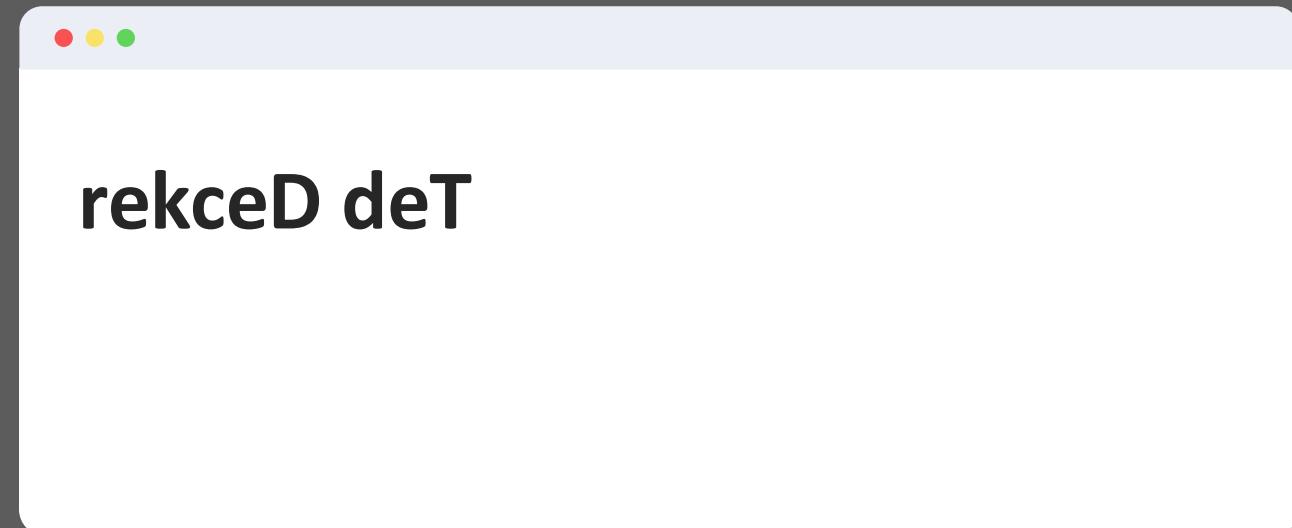
Many Array filters that work on primitive arrays, will turn text *into* a primitive array, and run.





# Array Filters and Text

```
1 {{ 'Ted Decker' | Reverse }}
```





# Array Filters and Text

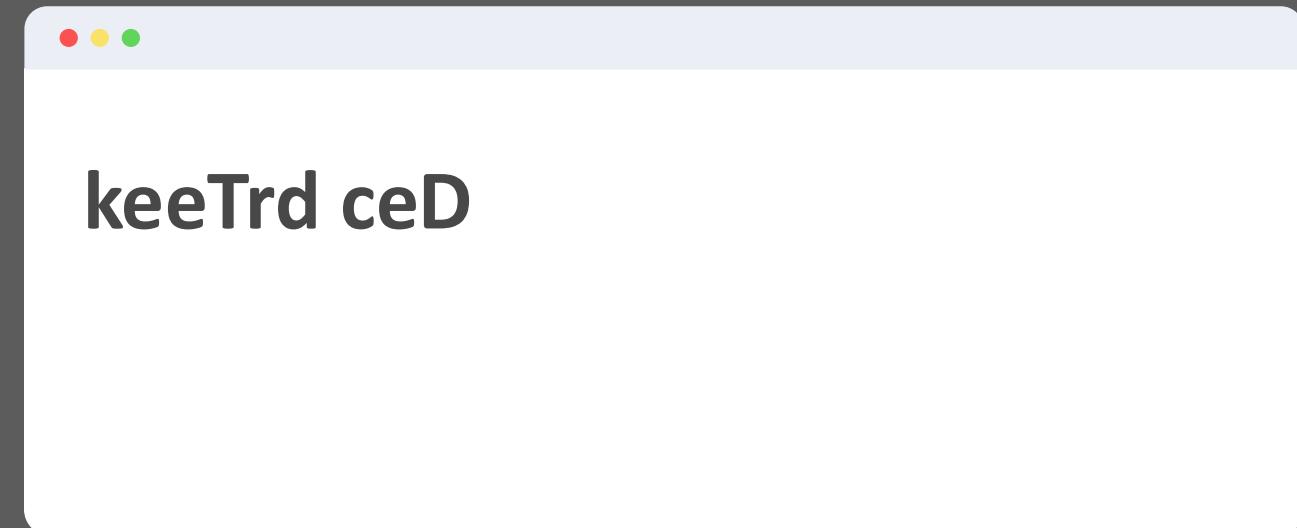
```
1 {{ 'Ted Decker' | Reverse | ToJSON }}
```

```
[  
  "r",  
  "e",  
  "k",  
  "c",  
  "e",  
  "D",  
  " ",  
  "d",  
  "e",  
  "T"]
```



# Array Filters and Text

```
1  {{ 'Ted Decker' | Reverse | Shuffle }}
```





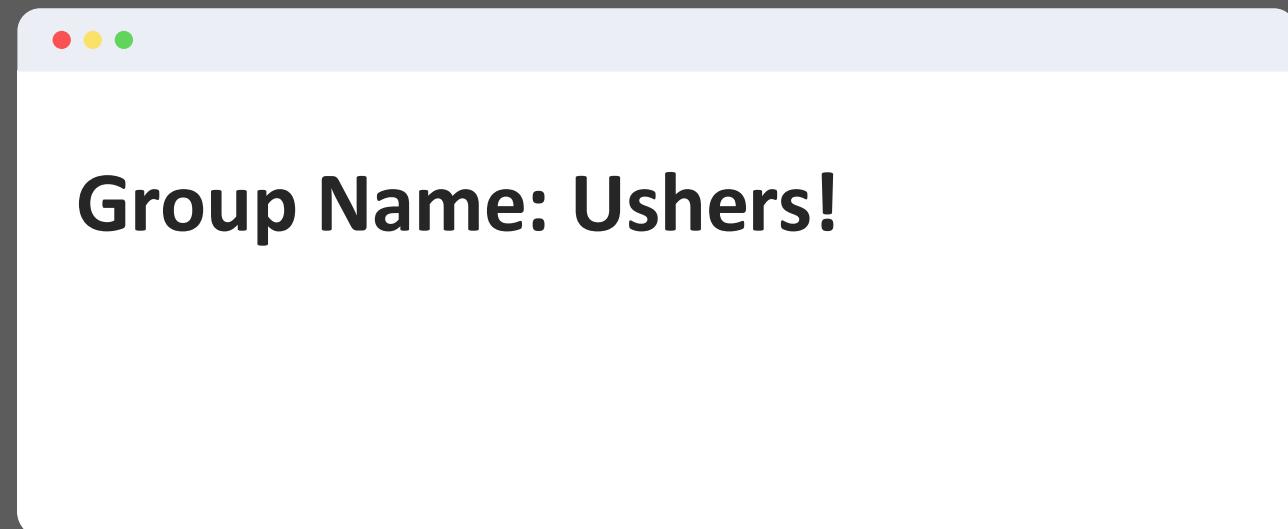
# Other Filters





# GroupByGuid

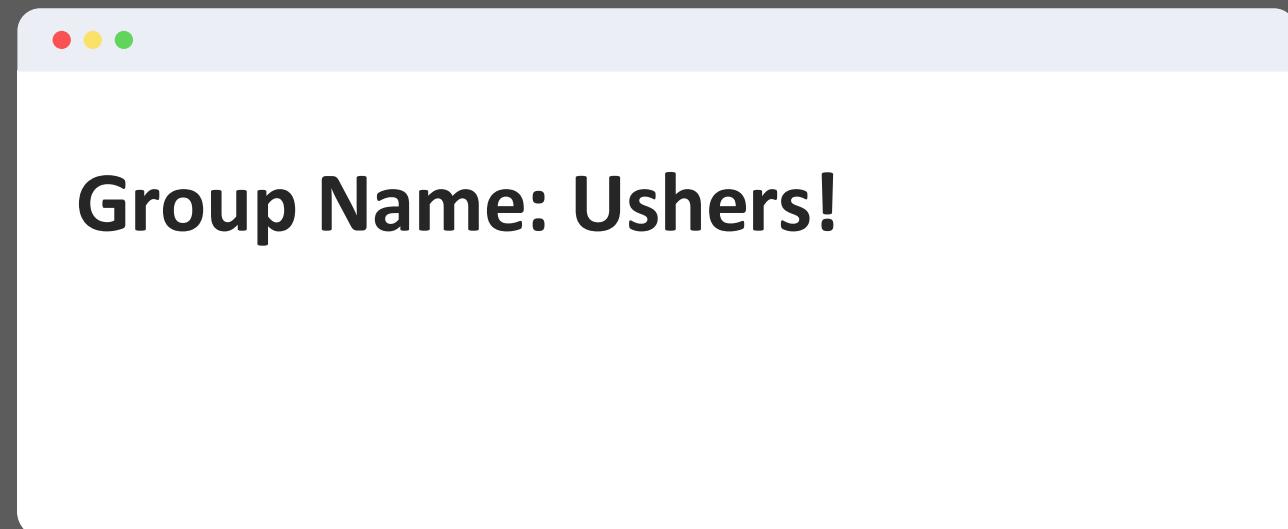
```
1  {% assign group = GroupMember.Group.Guid | GroupByGuid %}  
2  
3 Group Name: {{ group.Name }}!
```





# GroupById

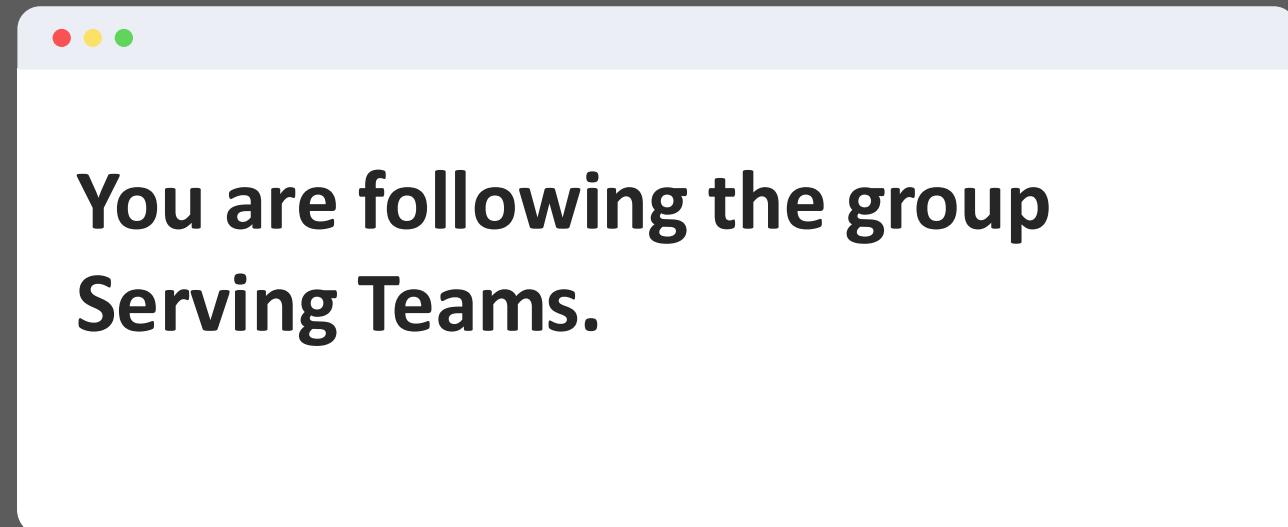
```
1  {% assign group = GroupMember.GroupId | GroupById %}  
2  
3 Group Name: {{ group.Name }}!
```





# IsFollowed

```
1  {% assign group = 56 | GroupById %}  
2  {% assign followed = group | IsFollowed %}  
3  {% if followed %}  
4      You are following the group {{ group.Name }}.  
5  {% endif %}
```



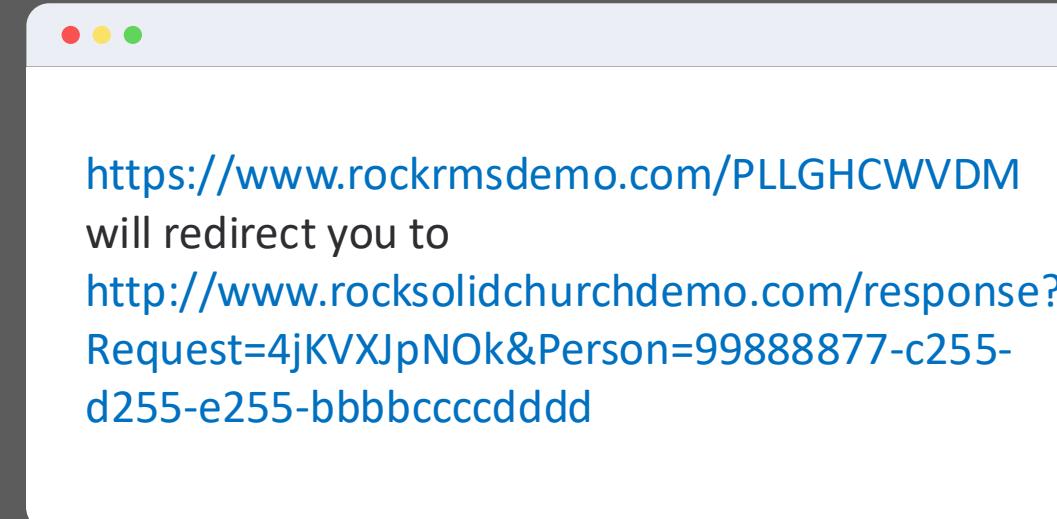


# CreateShortLink

```
1  {% assign url = 'Global' | Attribute:'PublicApplicationRoot' | Append:'response' %}  
2  {% assign url = url | Append:'?Request=4jKVXJpNOK' %}  
3  {% assign url = url | Append:'&Person=' | Append:CurrentPerson.PrimaryAlias.Guid %}  
4  
5  {{ url | CreateShortLink:'', 3, false, 10 }} will redirect you to {{ url }}
```

## Additional Options

- Token:** A specific string to use as the token.
- SiteId:** The site whose domain to use.
- Overwrite:** If the specified token already exists, should the old redirect be overwritten?
- Length:** Length of the random token.





# HasRightsTo

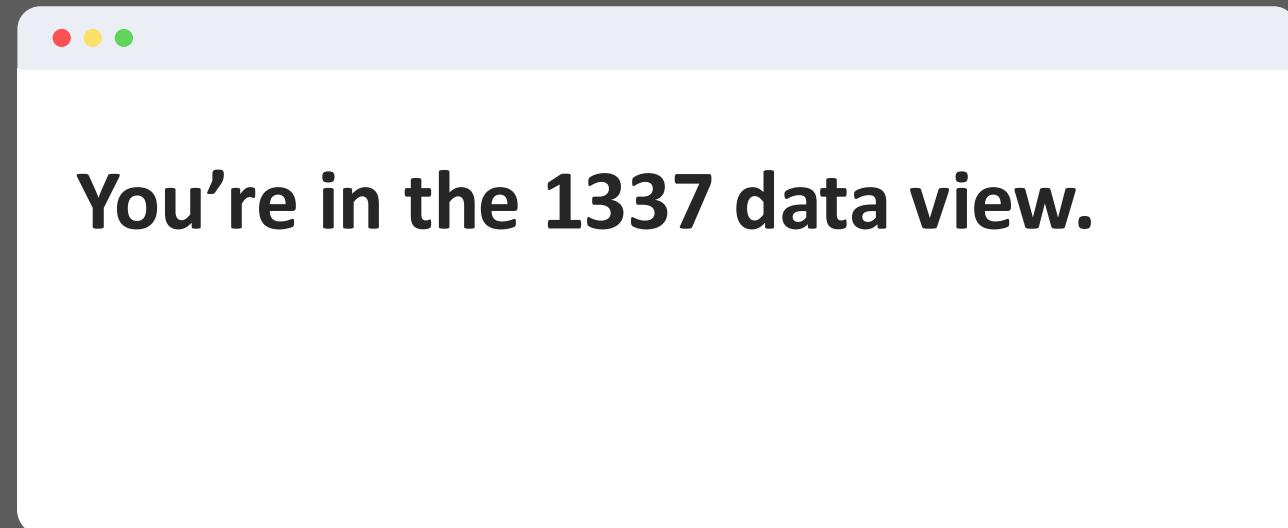
```
1 {{ person | HasRightsTo:'View' }}  
2  
3 {{ 12 | HasRightsTo:'Edit','Rock.Model.Group' }}
```





# IsInDataView

```
1  {%- assign in DataView = CurrentPerson.Id | IsInDataView:'1337' %}
2  {% if in DataView %}
3      You're in the 1337 data view.
4  {% endif %}
```



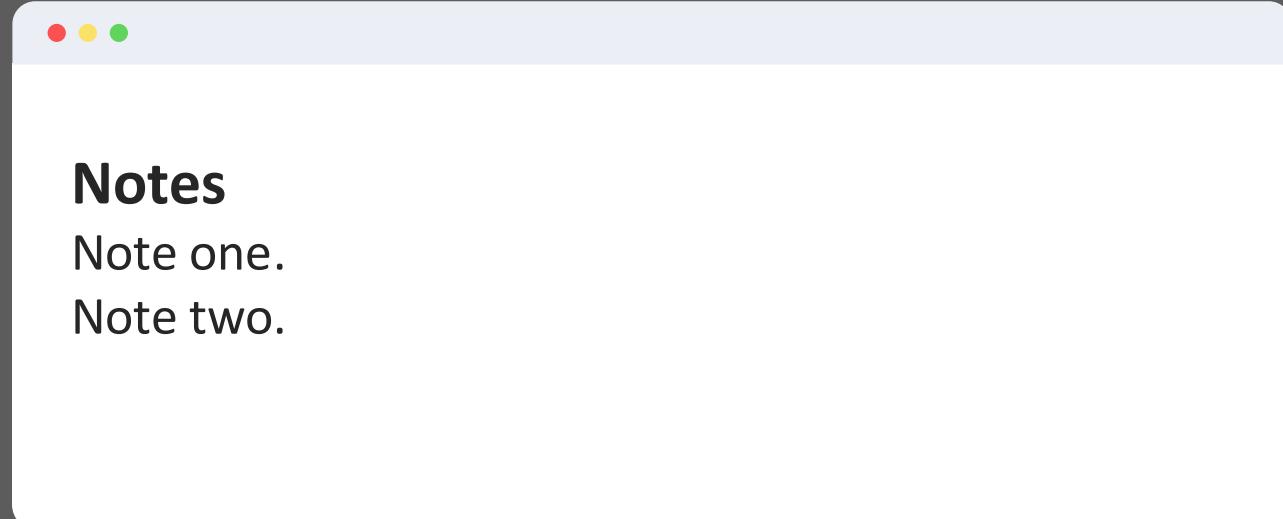


# Notes

```
1  {% assign notes = CurrentPerson | Notes:1 %}  
2  Notes  
3  {% for note in notes %}  
4      <p>{{ note.Text }}</p>  
5  {% endfor %}
```

## Additional Options

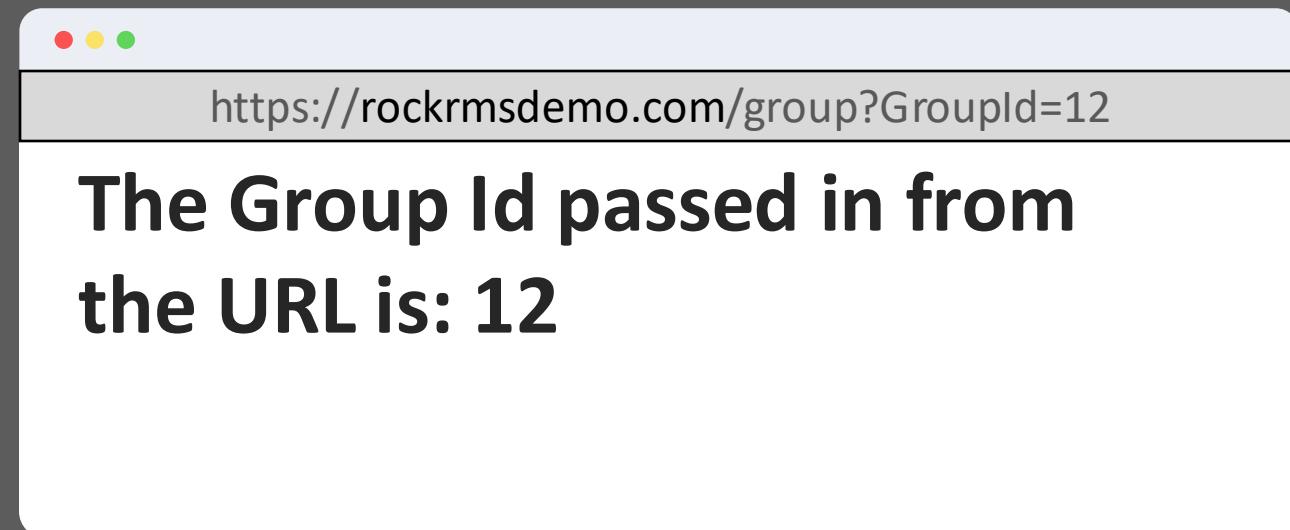
1. **Note Type Id:** or a comma delimited string of type ids.
2. **Sort Order:** asc or desc
3. **Count:** How many items to return.





# PageParameter

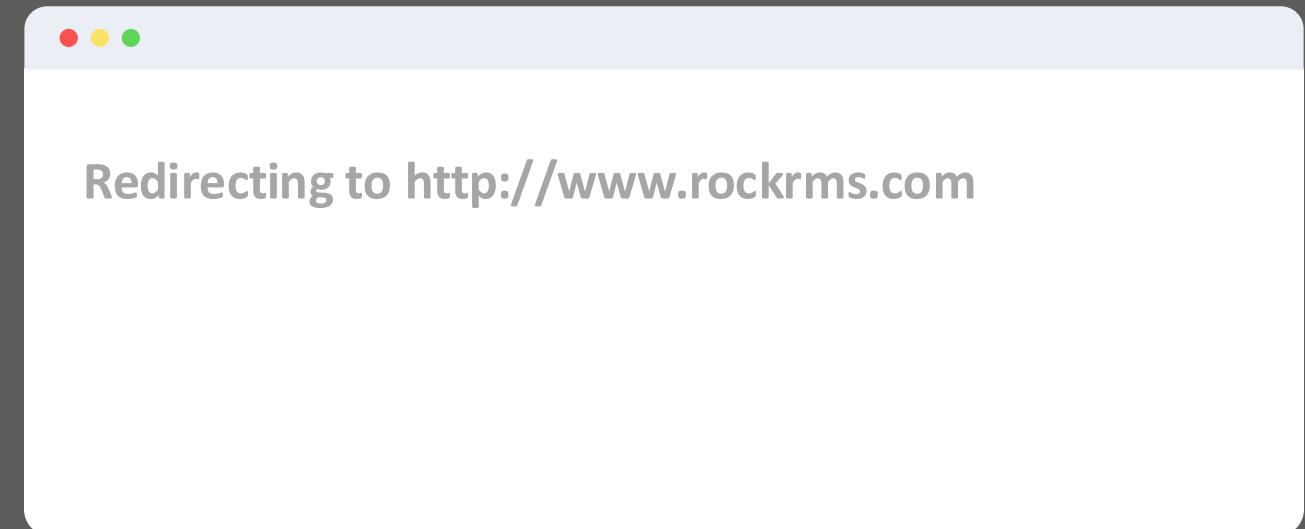
```
1 <p>The Group Id passed in from the URL is: {{ 'Global' | PageParameter:'GroupId' }}
```





# PageRedirect

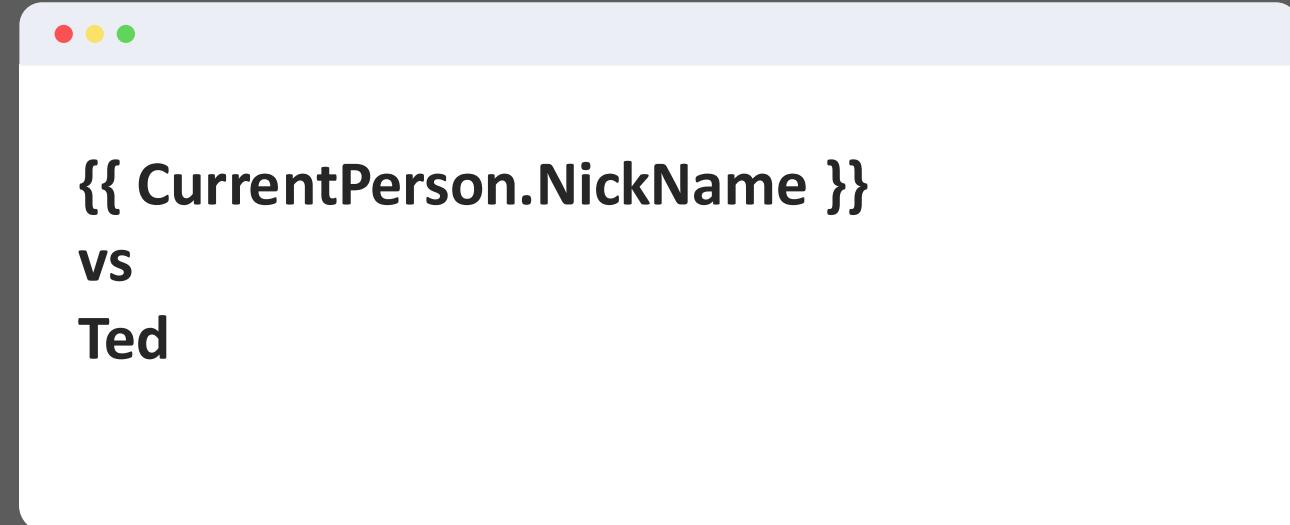
```
1 {{ 'http://www.rockrms.com' | PageRedirect }}
```





# RunLava

```
1 {{ Item.Content }}  
2 vs  
3 {{ Item.Content | RunLava }}
```





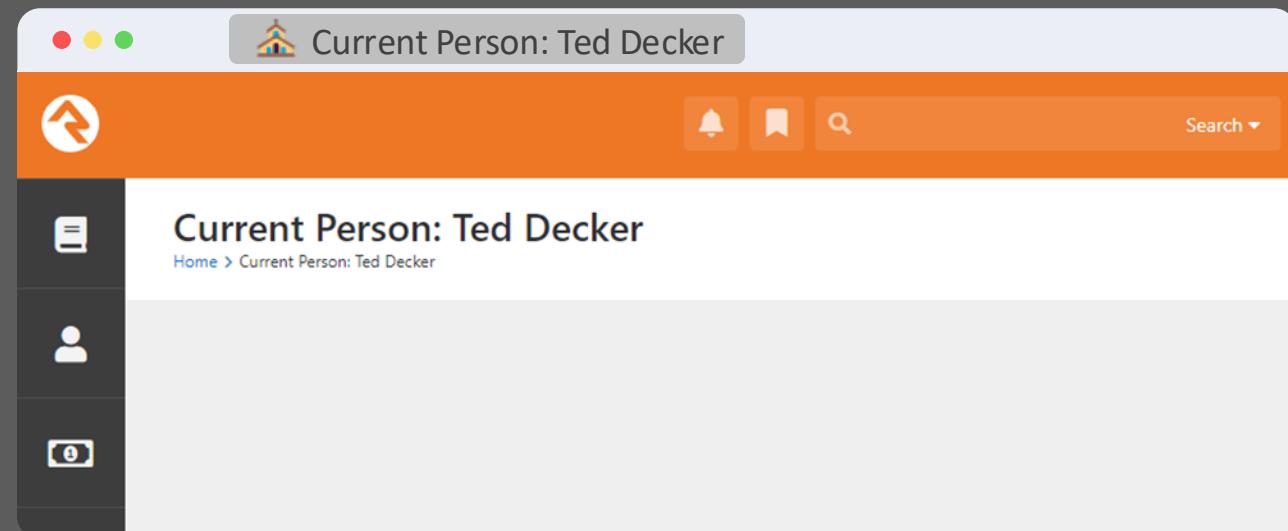
# SetPageTitle

```
1 {{ 'Current Person: ' | Append:CurrentPerson.FullName | SetPageTitle }}
```

## Additional Options

### 1. Title Location

- BrowserTitle
- PageTitle
- (Leave blank for “both”)

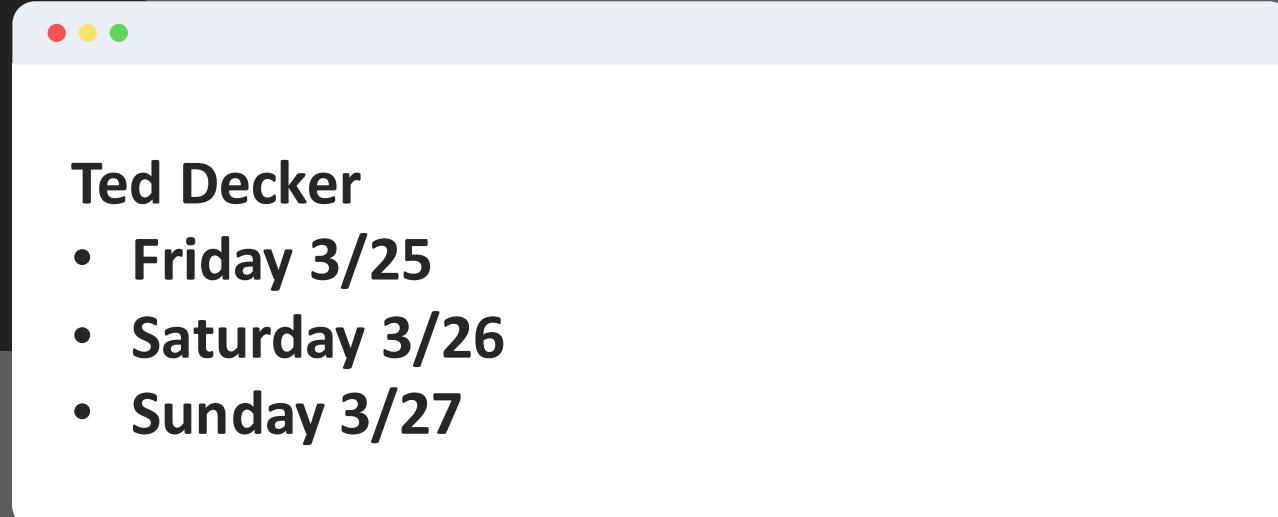




# FromJSON

```
1  {% capture customString %}  
2  {  
3      "Name": "Ted Decker",  
4      "Id": 5,  
5      "ServingTimes": [  
6          {  
7              "Date": "Friday 3/25",  
8              "GroupName": "Greeters",  
9              "Service": "6:00pm Service"  
10         },  
11         {  
12             "Date": "Saturday 3/26",  
13             "GroupName": "Preschool Volunteers",  
14             "Service": "First Service"  
15         },  
16         {  
17             "Date": "Sunday 3/27",  
18             "GroupName": "Tech Team",  
19             "Service": "Contemporary Service"  
20         }  
21     ]  
22 }  
23 {% endcapture %}
```

```
24  {% assign customObject = customString | FromJSON %}  
25  
26  {{ customObject.Name }}  
27  <ul>  
28      {% for servingTime in customObject.ServingTimes %}  
29          <li>{{ servingTime.Date }}</li>  
30      {% endfor %}  
31  <ul>
```





# PersistedDataset

```
1  {% assign data = 'tedschedule' | PersistedDataset %}  
2  {{ data.Name }}  
3  <ul>  
4      {% for servingTime in data.ServingTimes %}  
5          <li>{{ servingTime.Date }}</li>  
6      {% endfor %}  
7  <ul>
```





# ToJSON

```
1 {{ CurrentPerson | ToJSON }}
```

## Output

```
1 {  
2   "NickName": "Ted",  
3   "LastName": "Decker"  
4 }
```



# Url

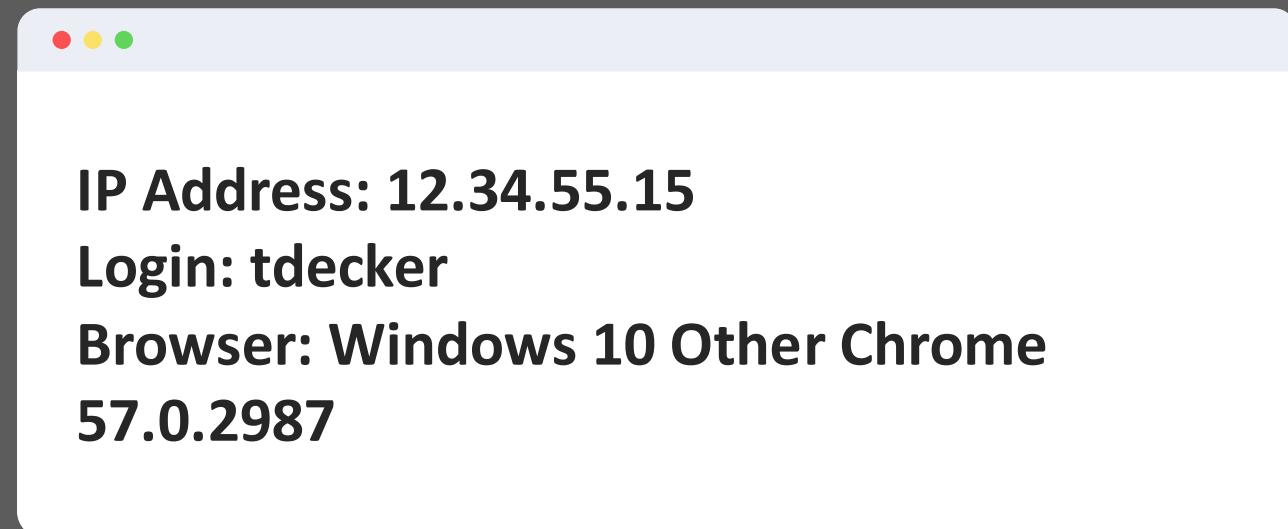
```
1  {% assign url = 'https://www.rockrms.com/WorkflowEntry/35?PersonId=2' %}  
2  Testing URL: {{ url }}  
3  Host: {{ url | Url:'host' }}  
4  Port: {{ url | Url:'port' }}  
5  Segments: {{ url | Url:'segments' | ToJSON }}  
6  Scheme: {{ url | Url:'scheme' }}  
7  Protocol: {{ url | Url:'protocol' }}  
8  Localpath: {{ url | Url:'localpath' }}  
9  Pathandquery: {{ url | Url:'pathandquery' }}  
10 Queryparameter no key:  
11 Queryparameter with key:  
12 Url: {{ url | Url:'url' }}
```

```
1  Testing URL https://www.rockrms.com/WorkflowEntry/35?PersonId=2  
2  Host: www.rockrms.com  
3  Port: 443  
4  Segments: [ "/", "WorkflowEntry/", "35" ]  
5  Scheme: https  
6  Protocol: https  
7  Localpath: /WorkflowEntry/35  
8  Pathandquery: /WorkflowEntry/35?PersonId=2  
9  Queryparameter no key:  
10 Queryparameter with key: 2  
11 Url: https://www.rockrms.com/WorkflowEntry/35?PersonId=2
```



# Client

```
1 IP Address: {{ 'Global' | Client:'ip' }} <br />
2 Login: {{ 'Global' | Client:'login' }} <br />
3 Browser: {{ 'Global' | Client:'browser' }} <br />
```





# Page

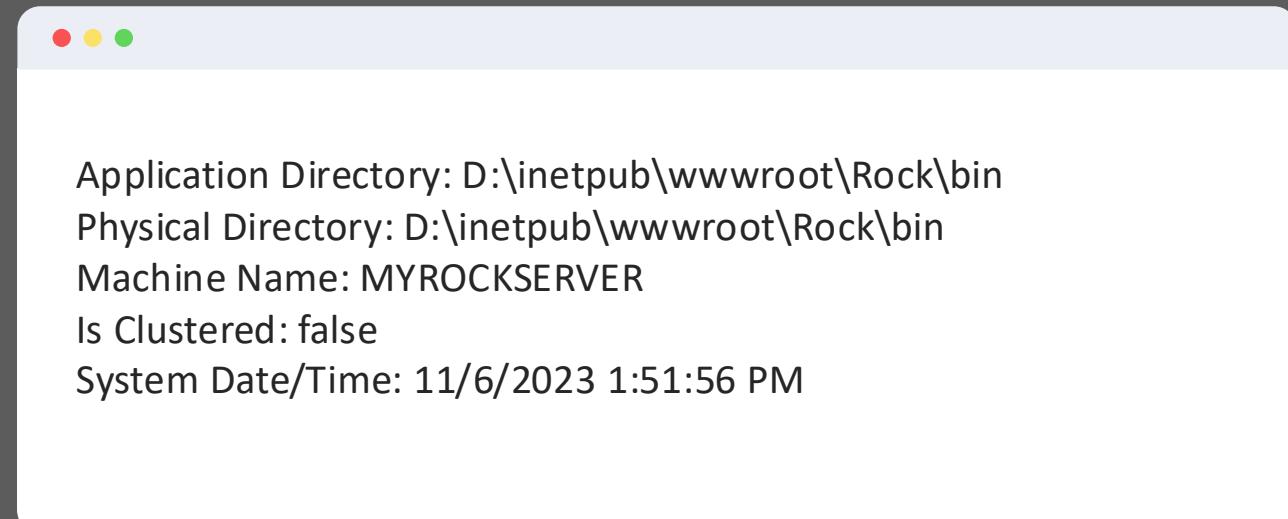
```
1 Title: {{ 'Global' | Page:'Title' }} <br/>
2 BrowserTitle: {{ 'Global' | Page:'BrowserTitle' }}<br/>
3 Description: {{ 'Global' | Page:'Description' }}<br/>
4 URL: {{ 'Global' | Page:'Url' }} <br/>
5 Page Id: {{ 'Global' | Page:'Id' }} <br/>
6 Host: {{ 'Global' | Page:'Host' }} <br/>
7 Path: {{ 'Global' | Page:'Path' }} <br/>
8 Site Name: {{ 'Global' | Page:'SiteName' }} <br/>
9 Site Id: {{ 'Global' | Page:'SiteId' }} <br/>
10 Theme: {{ 'Global' | Page:'Theme' }} <br/>
11 Layout: {{ 'Global' | Page:'Layout' }} <br/>
12 Scheme: {{ 'Global' | Page:'Scheme' }} <br/>
13 Cookies: {{ 'Global' | Page:'Cookies' }}<br/>
```

```
1 Title: Home
2 BrowserTitle: Home of the Browser's Title
3 Description: This is the page description
4 URL: http://localhost:6229/page/1
5 Page Id: 1
6 Host: localhost
7 Path: /page/1
8 Site Name: External Website
9 Site Id: 3
10 Theme: Stark
11 Layout: Homepage
12 Scheme: http
```



# RockInstanceConfig

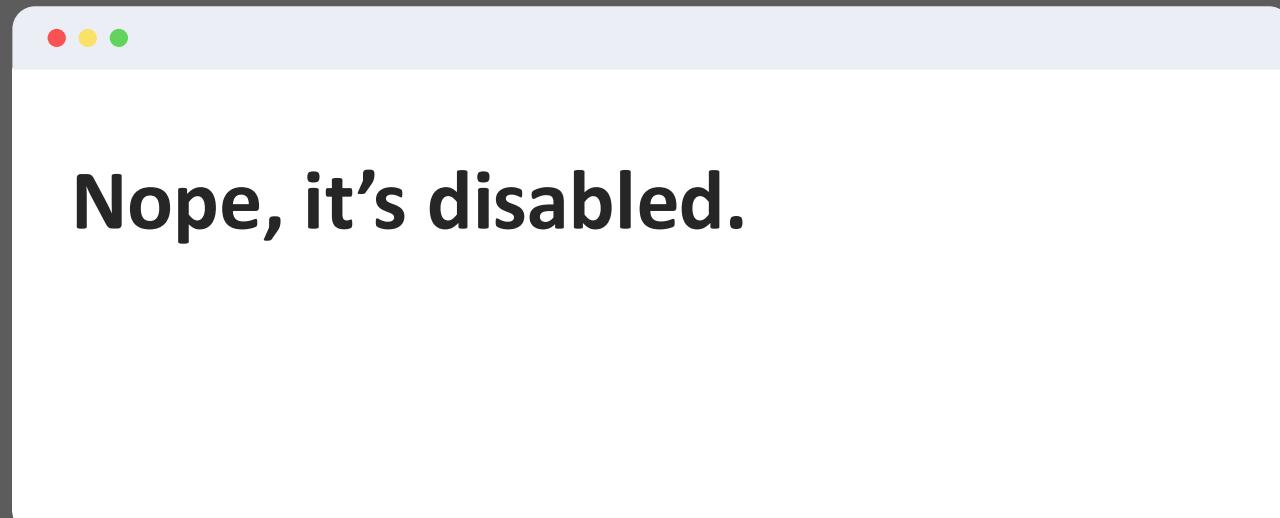
```
1 Application Directory: {{ 'ApplicationDirectory' | RockInstanceConfig }}  
2 Physical Directory: {{ 'PhysicalDirectory' | RockInstanceConfig }}  
3 Machine Name: {{ 'MachineName' | RockInstanceConfig }}  
4 Is Clustered: {{ 'IsClustered' | RockInstanceConfig }}  
5 System Date/Time: {{ 'SystemDateTime' | RockInstanceConfig }}
```





# AsBoolean

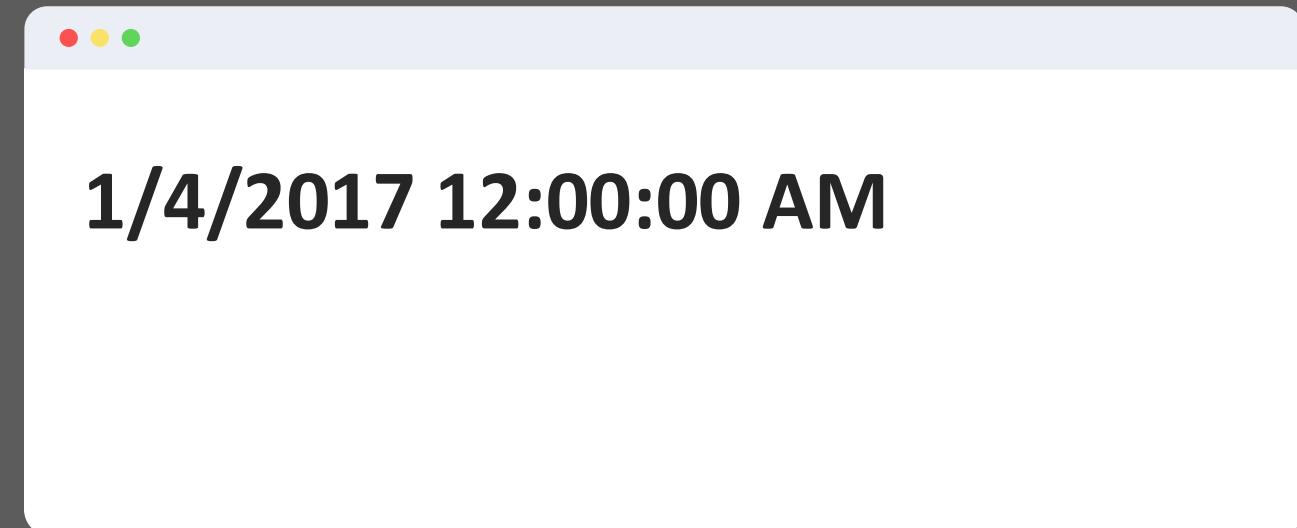
```
1  {% assign isEnabled = Workflow | Attribute:'Option' | AsBoolean %}  
2  
3  {% if isEnabled %}  
4      It is enabled.  
5  {% else %}  
6      Nope, it's disabled.  
7  {% endif %}
```





# AsDateTime

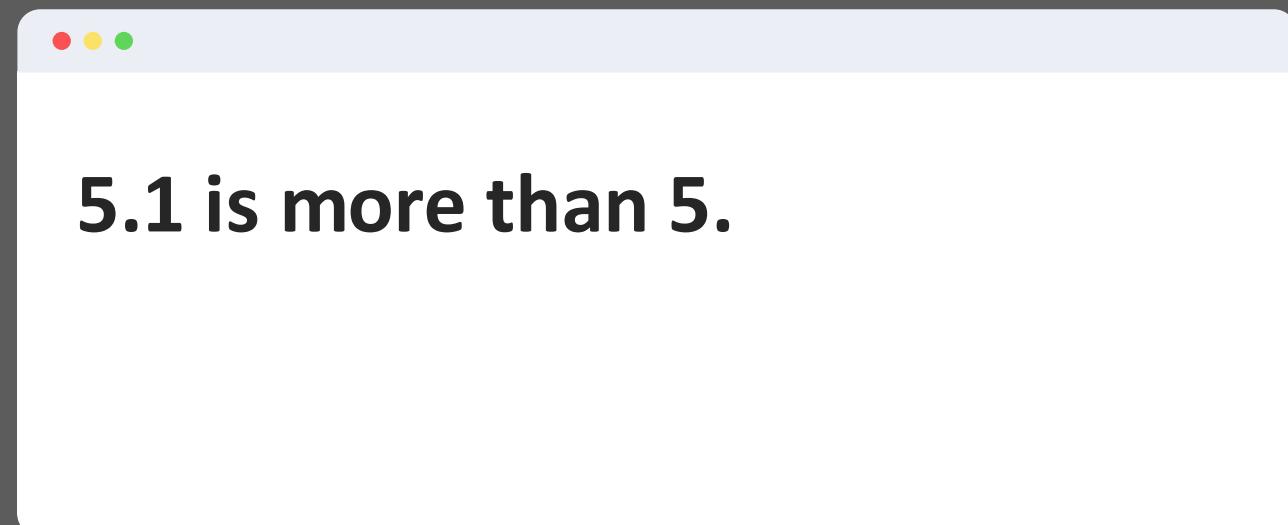
```
1 {{ '1/1/2017' | AsDateTime | DateAdd:3,'d' }}
```





# AsDecimal

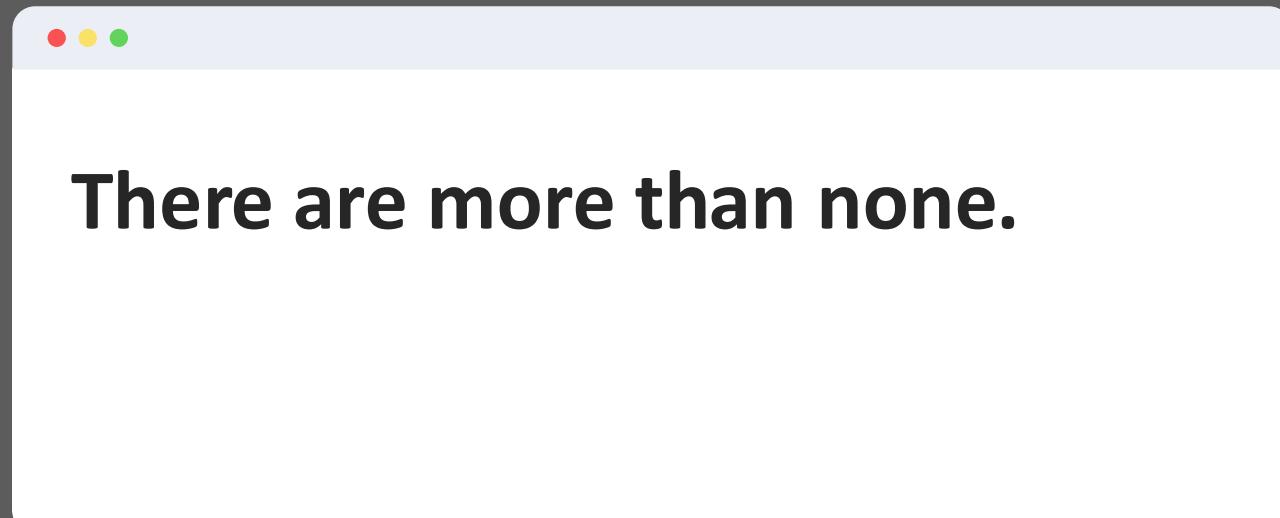
```
1  {% assign miles = Workflow | Attribute:'Miles' | AsDecimal %}  
2  
3  {% if miles > 5.0 %}  
4      {{ miles }} is more than 5.  
5  {% else %}  
6      Less than 5 miles.  
7  {% endif %}
```





# AsInteger

```
1  {% assign quantity = Workflow | Attribute:'Quantity' | AsInteger %}  
2  
3  {% if quantity > 0 %}  
4      There are more than none.  
5  {% else %}  
6      There are none.  
7  {% endif %}
```





# Other Filters

- HmacSha1
- HmacSha256
- Md5
- Sha1
- Sha256
- Base64Encode
- FromBase64