```
!pip install turicreate
```

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

Fire up Turi Create

We always start with this line before using any part of Turi Create

```
import turicreate
```

Load a tabular data set

```
sf = turicreate.SFrame('/content/gdrive/My Drive/Turicreate/Week 1/people-example.csv')
```

```
Finished parsing file /content/gdrive/My Drive/Turicreate/Week 1/people-example.csv
Parsing completed. Parsed 7 lines in 0.030245 secs.
------------------------------------------------------
Inferred types from first 100 line(s) of file as
column_type_hints=[str,str,str,int]
If parsing fails due to incorrect types, you can correct
the inferred type list above and pass it to read_csv in
the column_type_hints argument
------------------------------------------------------
Read 7 lines. Lines per second: 539.749
Finished parsing file /content/gdrive/My Drive/Turicreate/Week 1/people-example.csv
Parsing completed. Parsed 7 lines in 0.013309 secs.
```

SFrame basics

```
sf #we can view first few lines of table
```

First Name	Last Name	Country	age
Bob	Smith	United States	24
Alice	Williams	Canada	23
Malcolm	Jone	England	22
Felix	Brown	USA	23
Alex	Cooper	Poland	23
Tod	Campbell	United States	22
Derek	Ward	Switzerland	25

[7 rows x 4 columns]

```
sf.tail()  # view end of the table
```

```
sf.tail()  # view end of the table
```

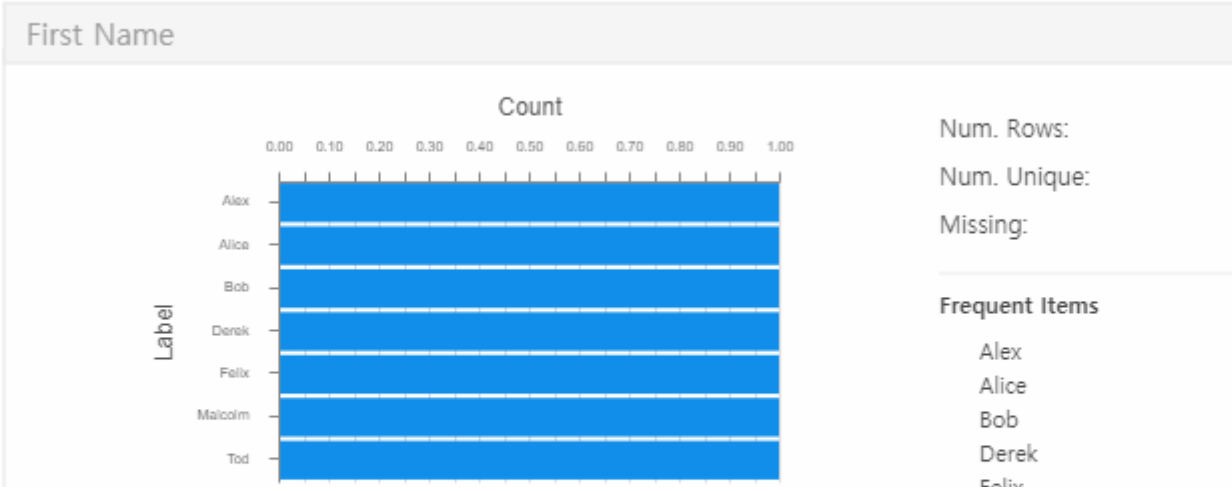
First Name	Last Name	Country	age
Bob	Smith	United States	24
Alice	Williams	Canada	23
Malcolm	Jone	England	22
Felix	Brown	USA	23
Alex	Cooper	Poland	23
Tod	Campbell	United States	22
Derek	Ward	Switzerland	25

[7 rows x 4 columns]

Turi Create visualization

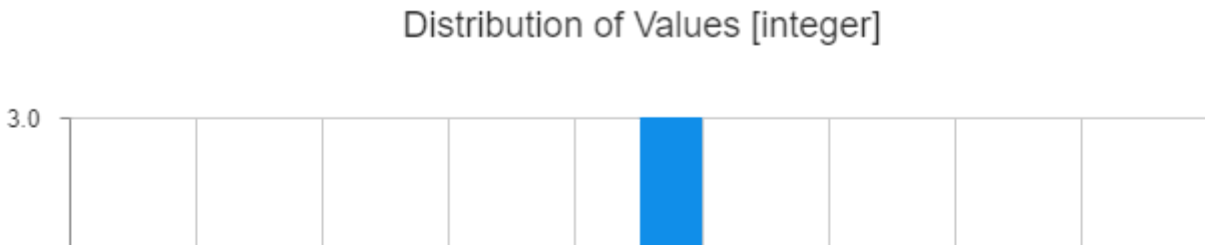
```
# .show() visualizes any data structure in Turi Create
sf.show()
```

```
Materializing SFrame
```

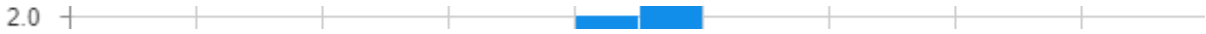



```
sf['age'].show()
```

```
Materializing SArray
```



Inspect columns of dataset




```
sf['Country']
```

```
dtype: str
Rows: 7
['United States', 'Canada', 'England', 'USA', 'Poland', 'United States', 'Switzerland']
```

```
sf['age']
```

```
dtype: int
Rows: 7
[24, 23, 22, 23, 23, 22, 25]
```

Some simple columnar operations


```
sf['age'].mean()
```

```
23.14285714285714
```

```
sf['age'].max()
```

```
25
```

Create new columns in our SFrame

```
sf
```

First Name	Last Name	Country	age
Bob	Smith	United States	24
Alice	Williams	Canada	23
Malcolm	Jone	England	22
Felix	Brown	USA	23
Alex	Cooper	Poland	23
Tod	Campbell	United States	22
Derek	Ward	Switzerland	25

[7 rows x 4 columns]

```
sf['Full Name'] = sf['First Name'] + ' ' + sf['Last Name']
```

```
sf
```

First Name	Last Name	Country	age	Full Name
Bob	Smith	United States	24	Bob Smith
Alice	Williams	Canada	23	Alice Williams
Malcolm	Jone	England	22	Malcolm Jone
Felix	Brown	USA	23	Felix Brown
Alex	Cooper	Poland	23	Alex Cooper
Tod	Campbell	United States	22	Tod Campbell
Derek	Ward	Switzerland	25	Derek Ward

[7 rows x 5 columns]

```
sf['age'] * sf['age']
```

```
dtype: int
Rows: 7
[576, 529, 484, 529, 529, 484, 625]
```

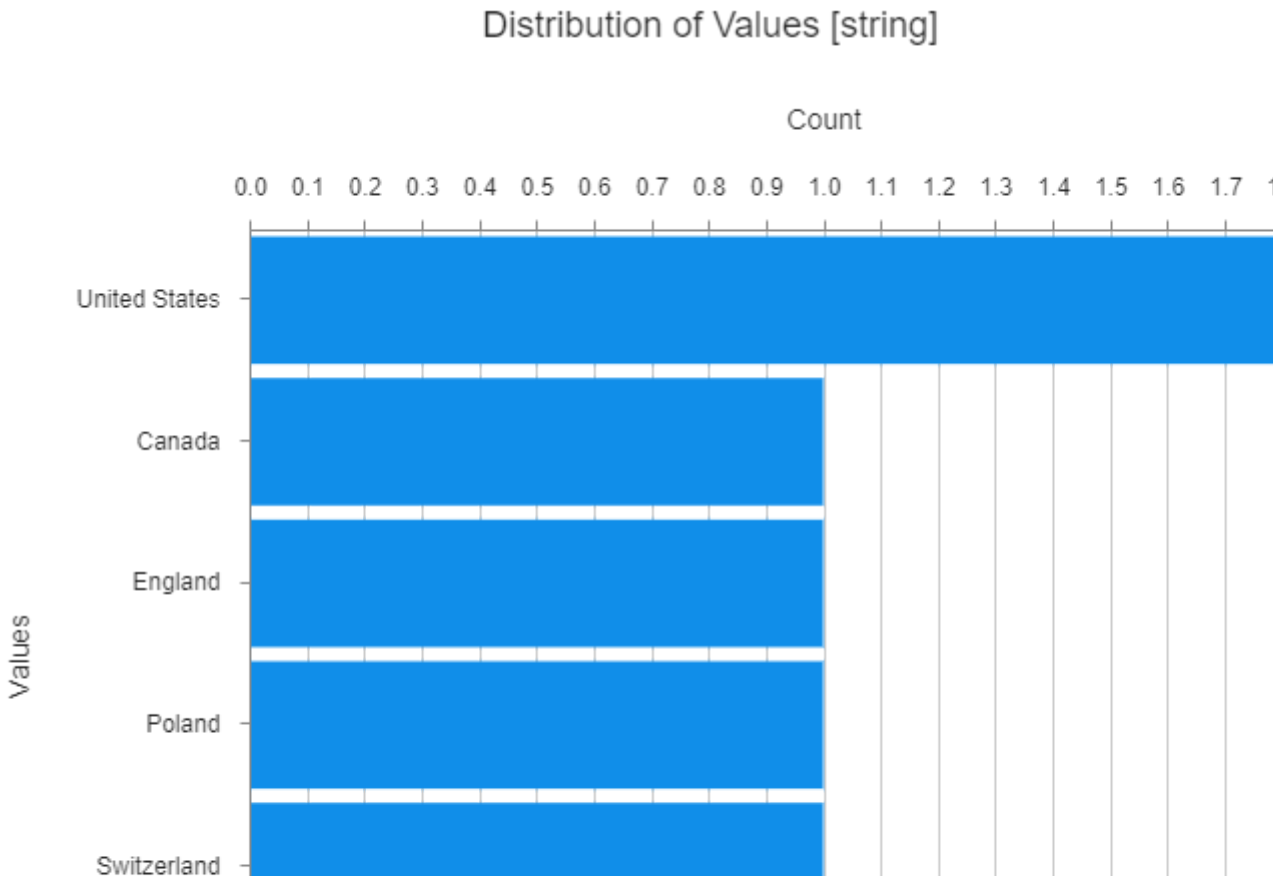
Use the apply function to do a advance transformation of our data

```
sf['Country']
```

```
dtype: str
Rows: 7
['United States', 'Canada', 'England', 'USA', 'Poland', 'United States', 'Switzerland']
```

```
sf['Country'].show()
```

```
Materializing SArray
```




```
def transform_country(country):
    if country == 'USA':
        return 'United States'
    else:
        return country


transform_country('Brazil')
```

```
'Brazil'
```

```
transform_country('Brasil')
```

```
'Brasil'
```

```
transform_country('USA')
```

```
'United States'
```

```
sf['Country'].apply(transform_country)
```

```
dtype: str
Rows: 7
['United States', 'Canada', 'England', 'United States', 'Poland', 'United States', 'Switzerla
```

```
sf['Country'] = sf['Country'].apply(transform_country)
```

```
sf
```

First Name	Last Name	Country	age	Full Name
Bob	Smith	United States	24	Bob Smith
Alice	Williams	Canada	23	Alice Williams
Malcolm	Jone	England	22	Malcolm Jone
Felix	Brown	United States	23	Felix Brown
Alex	Cooper	Poland	23	Alex Cooper
Tod	Campbell	United States	22	Tod Campbell
Derek	Ward	Switzerland	25	Derek Ward

[7 rows x 5 columns]

```
sf = turicreate.SFrame('/content/gdrive/My Drive/Turicreate/Week 1/people_wiki.sframe')
```

```
sf
```

URI	name	text
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...

[59071 rows x 3 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.