

Plotly 기본

학습 내용

- 반응형 브라우저 기반 시각화 라이브러리(plotly)를 소개하고 실습해 본다.

plotly로 시각화하기

- cufflinks와 iplot()을 활용. pandas.plot()와 같이 판다스 데이터 시각화
- plotly.express 라이브러리 활용
- cufflinks 는 무엇인가?
 - 판다스 데이터 프레임과 Plotly를 연결하여 사용자가 판다스로부터 직접 시각화를 할 수 있는 라이브러리

01 시작하기 - 설치(Plotly and Cufflinks)

- pip install plotly
- pip install cufflinks
- 자료 실행 버전
 - plotly 5.6.0
 - cufflinks 0.17.3
 - python 3.9.12
- 버전 지정 설치 시,
 - pip install plotly==4.10.0
 - conda install -c plotly plotly==4.1.0

In [1]:

```
import plotly
import cufflinks as cf
import pandas as pd
import numpy as np
import sys
```

프로그램 버전 확인

In [3]:

```
print(sys.version)
print(plotly.__version__)
print(cf.__version__)
print(pd.__version__)
print(np.__version__)
```

```
3.9.12 (main, Apr  4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
5.6.0
0.17.3
1.4.2
1.21.5
```

In [4]:

```
# 오프라인 모드에서도 인터랙티브한 그래픽을 가능하도록 하기
# Enabling the offline mode for interactive plotting locally
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected=True)
cf.go_offline()
```

데이터 생성 및 plot

In [36]:

```
# 데이터 만들기
dat = np.random.randn(50,4) # 50개 4개 컬럼
df = pd.DataFrame(dat, columns='A B C D'.split())
print(df.shape)
df.head()
```

(50, 4)

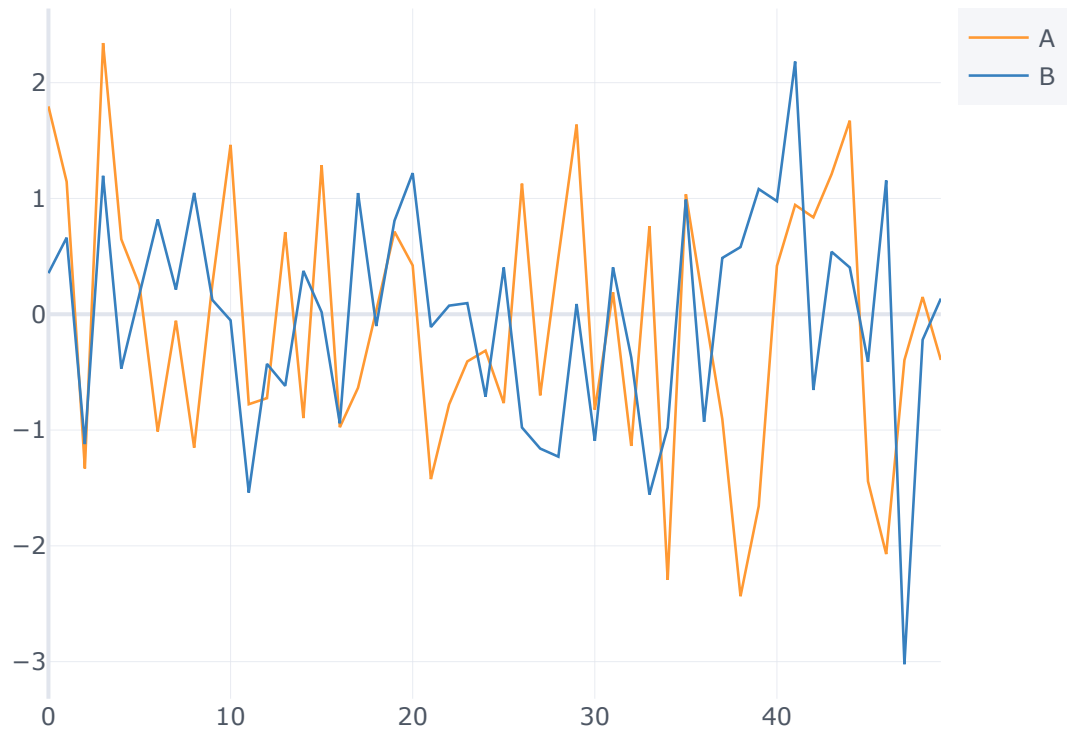
Out[36]:

| | A | B | C | D |
|---|-----------|-----------|-----------|-----------|
| 0 | 1.795314 | 0.354644 | 0.208492 | -0.853836 |
| 1 | 1.144684 | 0.661491 | -1.096486 | 0.157183 |
| 2 | -1.331844 | -1.121284 | 0.004359 | -1.015948 |
| 3 | 2.342148 | 1.195937 | -0.743732 | -0.580941 |
| 4 | 0.647516 | -0.469113 | 1.490221 | -0.584984 |

02 Line Plot 그려보기

In [37]:

```
df[ ['A', 'B'] ].iplot(kind='line')
```

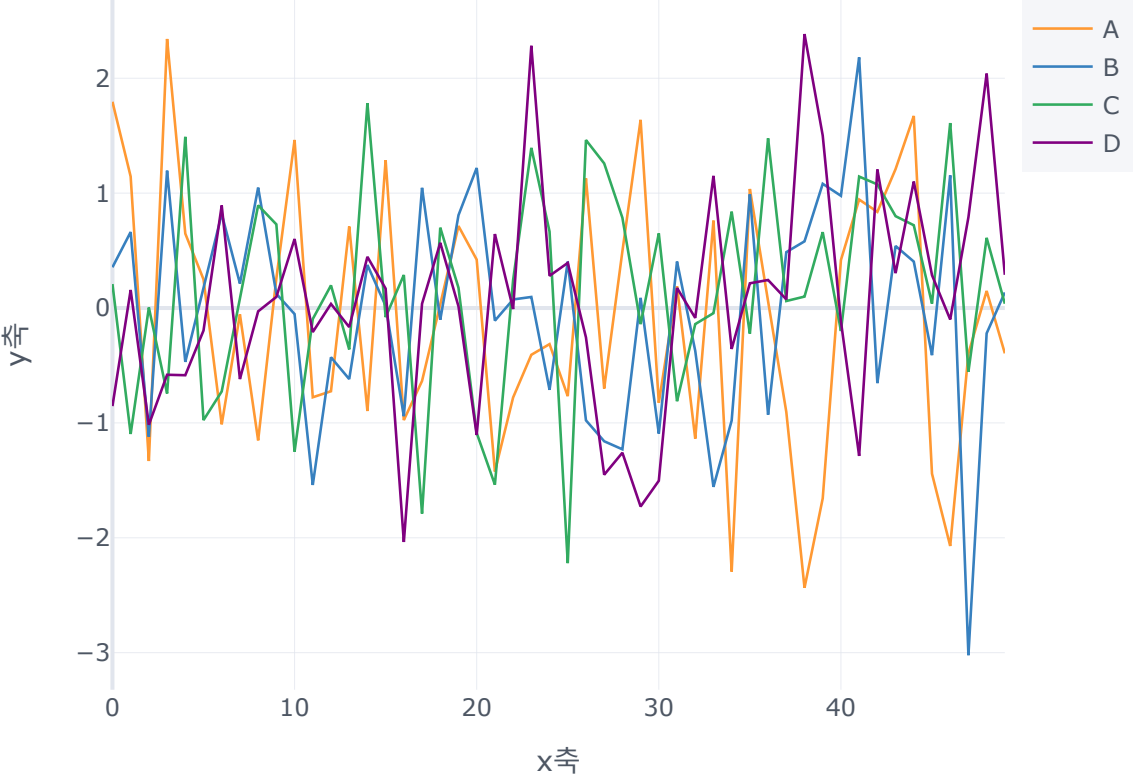


[Export to plot.ly](#)

In [38]:

```
df.iplot(kind='line', xTitle="x축", yTitle="y축", title="데이터 제목")
```

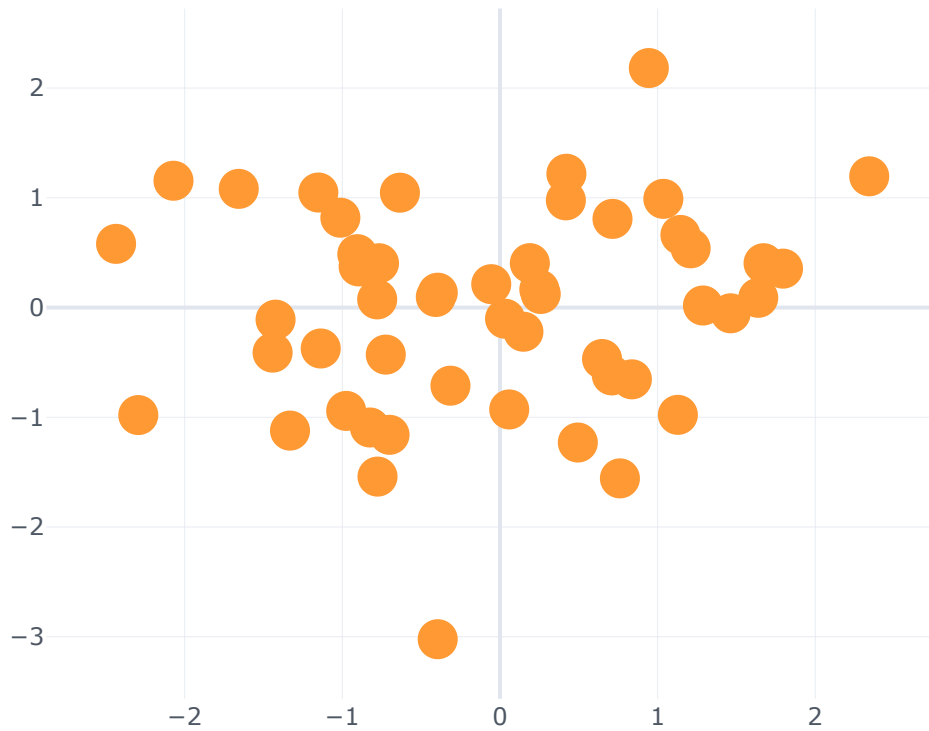
데이터 제목



Export to plotly

In [39]:

```
df.iplot(kind='scatter', x='A', y='B', mode='markers', size=20)
```



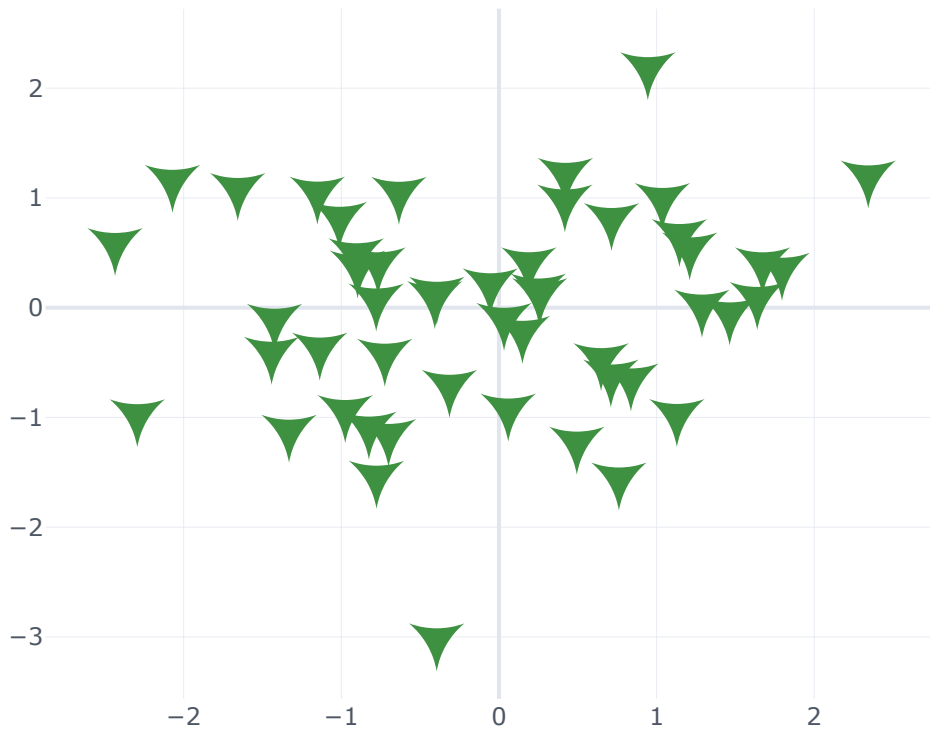
[Export to plot.ly](#)

Scatter Plot

- scatter Plot을 위한 mode
 - lines
 - markers
 - lines+markers
 - lines+text
 - markers+text
 - lines+markers+text

In [43]:

```
df.iplot(kind='scatter',  
         x='A',y='B',  
         mode='markers',  
         size=20,      # 크기  
         color='#3f9142', # 색  
         symbol=20)    # 점의 표시
```



[Export to plot.ly](#)

In [44]:

```
df2 = pd.DataFrame({'items':['bag','apple','cap'],  
                   'Values':[32,43,50,]})  
df2
```

Out[44]:

| | items | Values |
|---|-------|--------|
| 0 | bag | 32 |
| 1 | apple | 43 |
| 2 | cap | 50 |

Bar Plot

In [45]:

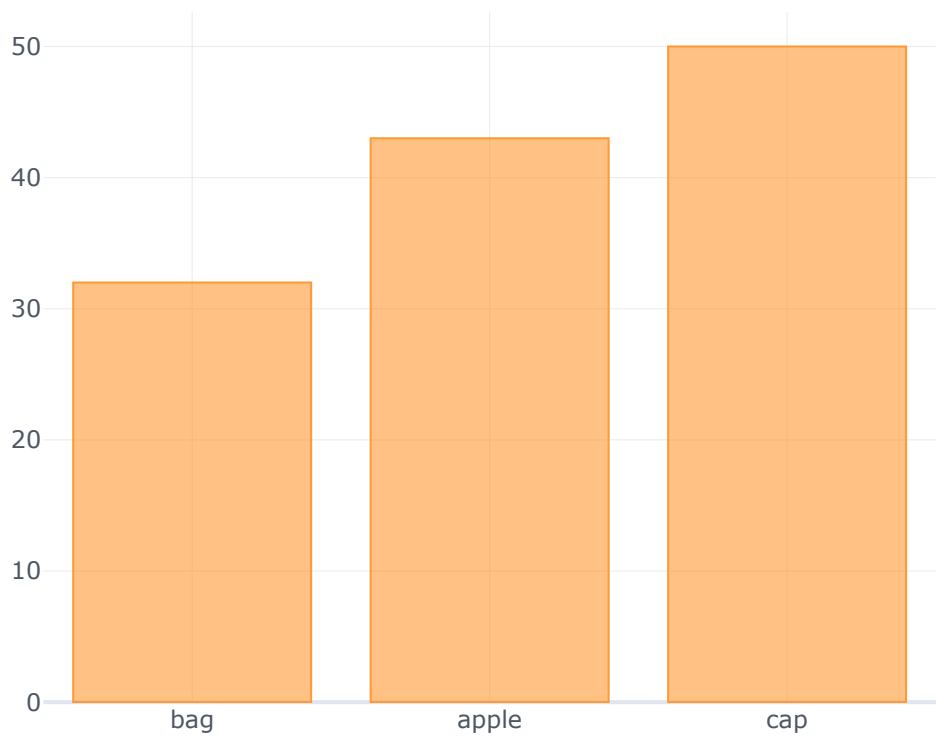
```
df2
```

Out[45]:

| | items | Values |
|---|-------|--------|
| 0 | bag | 32 |
| 1 | apple | 43 |
| 2 | cap | 50 |

In [46]:

```
df2.iplot(kind='bar',x='items',y='Values')
```



[Export to plotly](#)

In [47]:

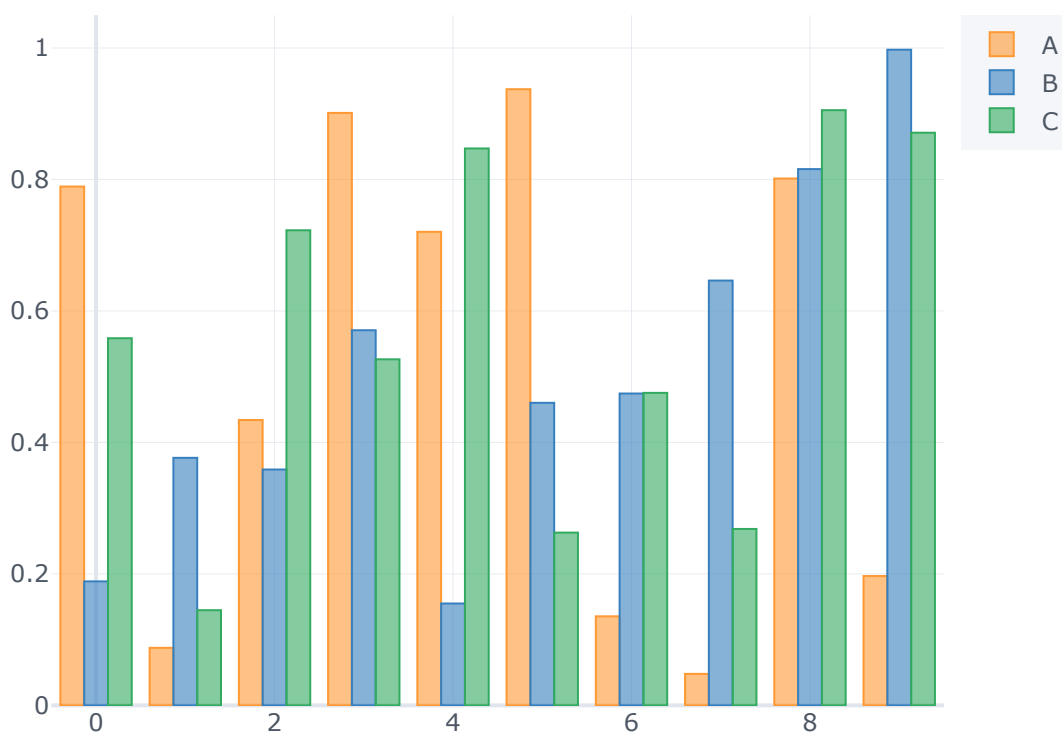
```
df = pd.DataFrame(np.random.rand(10,3),  
                  columns=['A', 'B', 'C'])  
df.head()
```

Out[47]:

| | A | B | C |
|---|----------|----------|----------|
| 0 | 0.789214 | 0.188631 | 0.558508 |
| 1 | 0.087561 | 0.376575 | 0.144793 |
| 2 | 0.434154 | 0.358793 | 0.722729 |
| 3 | 0.901232 | 0.570671 | 0.526448 |
| 4 | 0.720390 | 0.154980 | 0.847210 |

In [48]:

```
# 0,1,2,3,4에 대한 A,B,C의 값  
df.iplot(kind='bar')
```

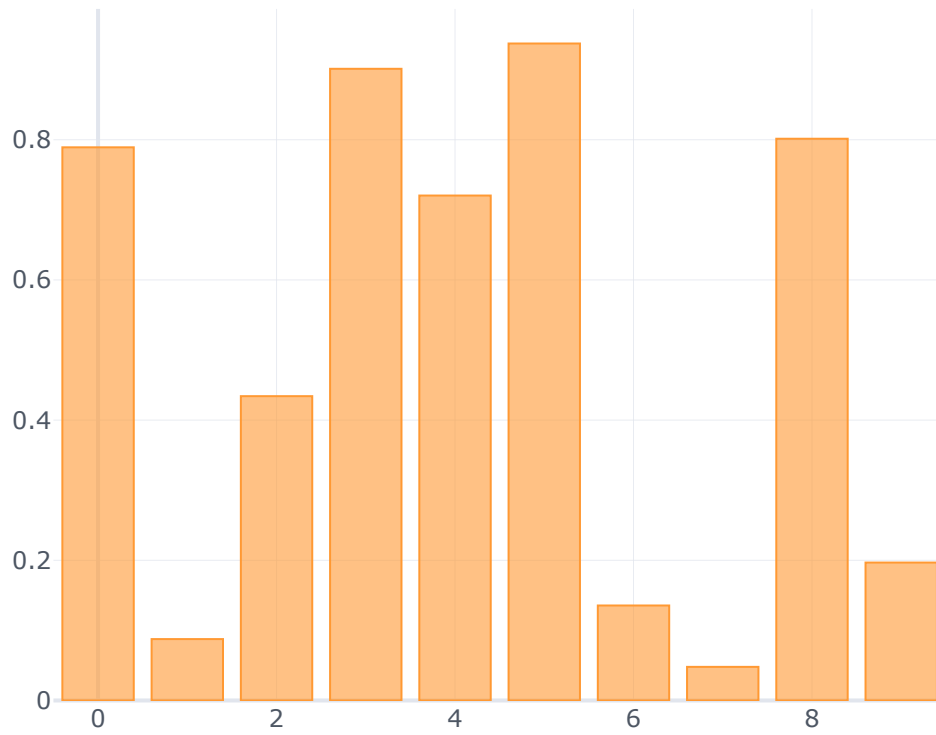


[Export to plotly](#)

A컬럼만 보기

In [49]:

```
df['A'].iplot(kind='bar')
```

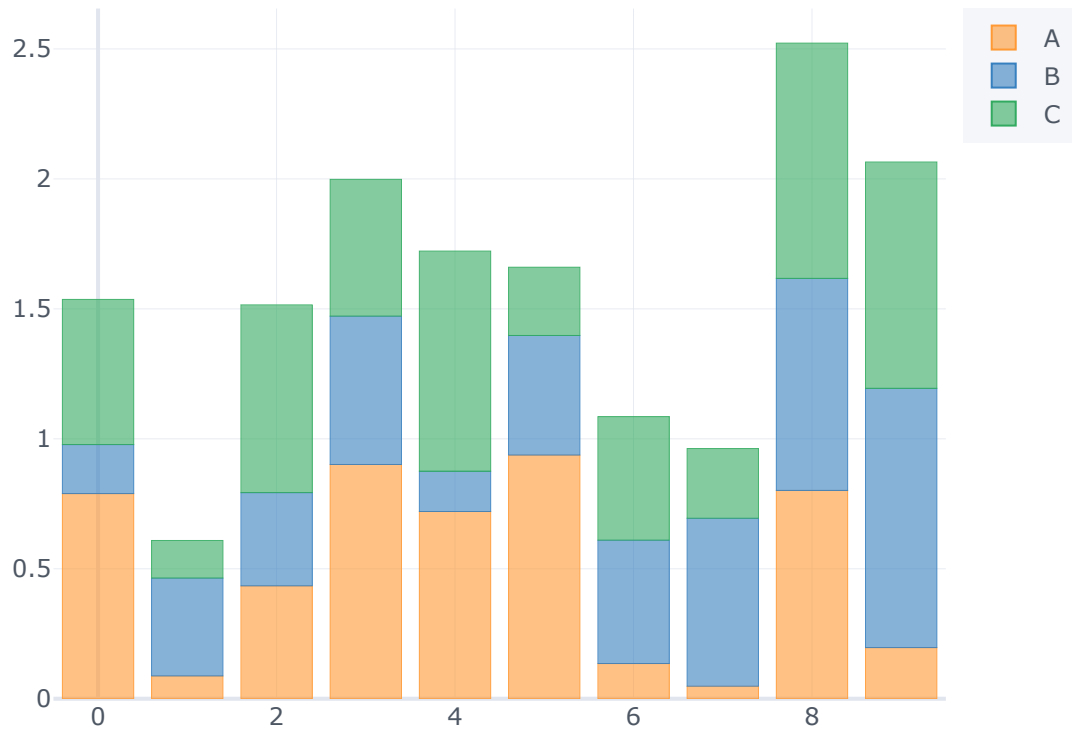


[Export to plotly](#)

Stack plot

In [50]:

```
df.iplot(kind='bar', barmode='stack')
```

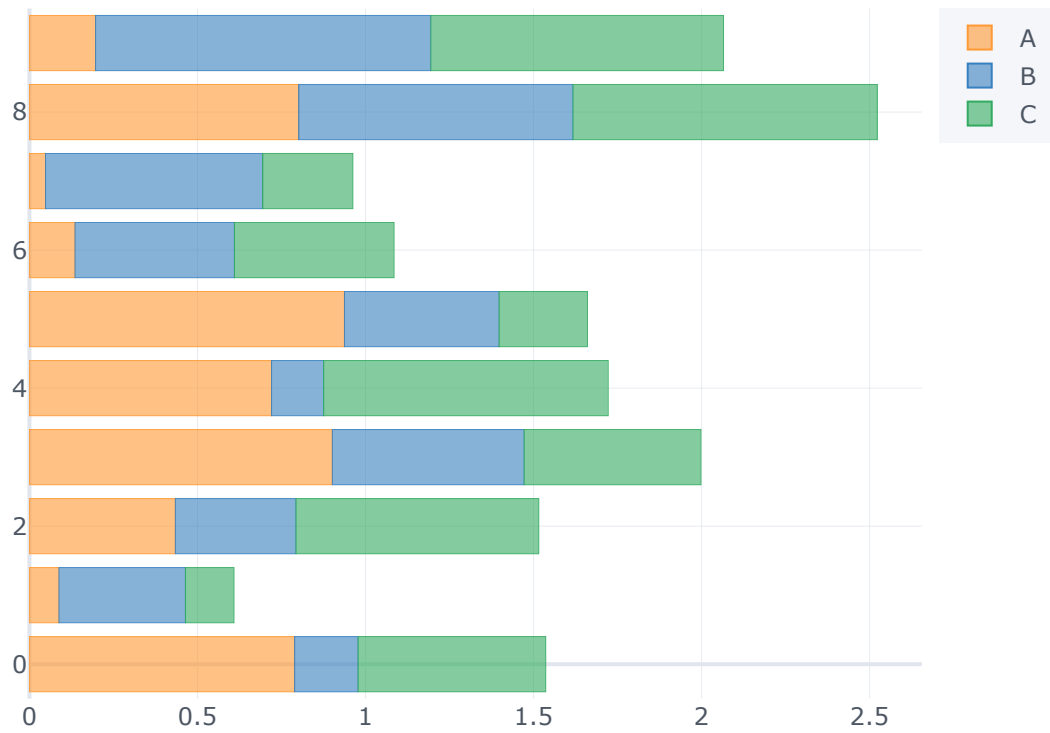


[Export to plot.ly](#)

수평 막대 그래프

In [52]:

```
df.iplot(kind='barh', barmode='stack')
```

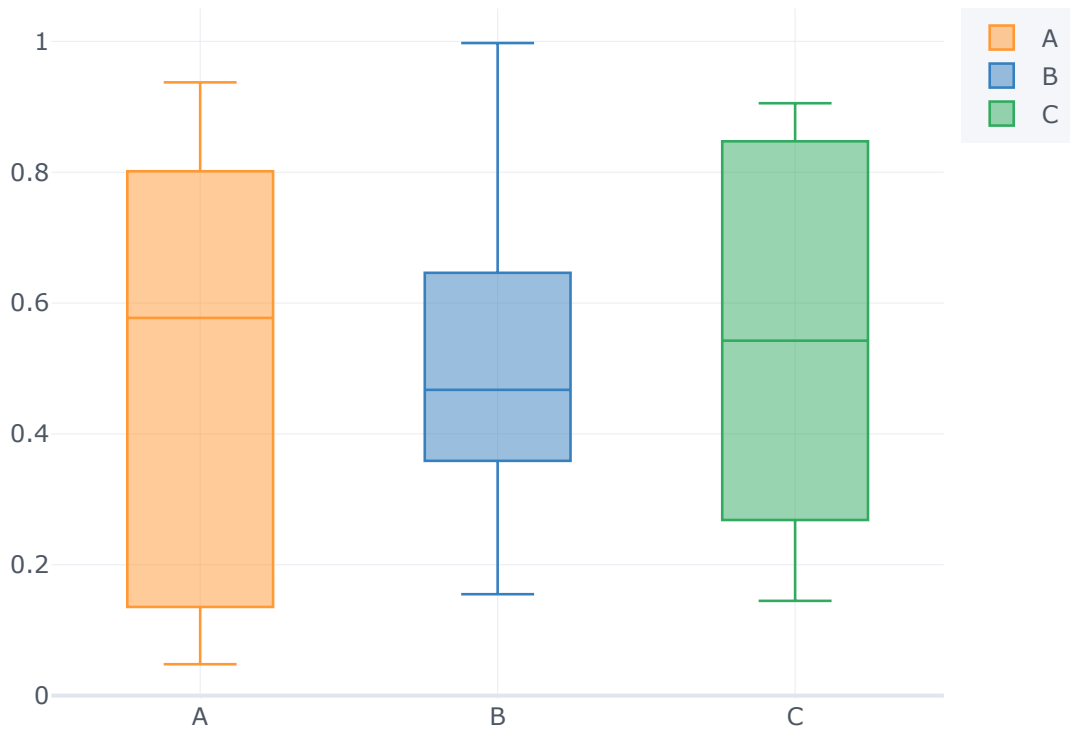


[Export to plotly](#)

Boxplot

In [53]:

```
df.iplot(kind='box')
```



[Export to plotly](#)

3D Surface Plot

In [54]:

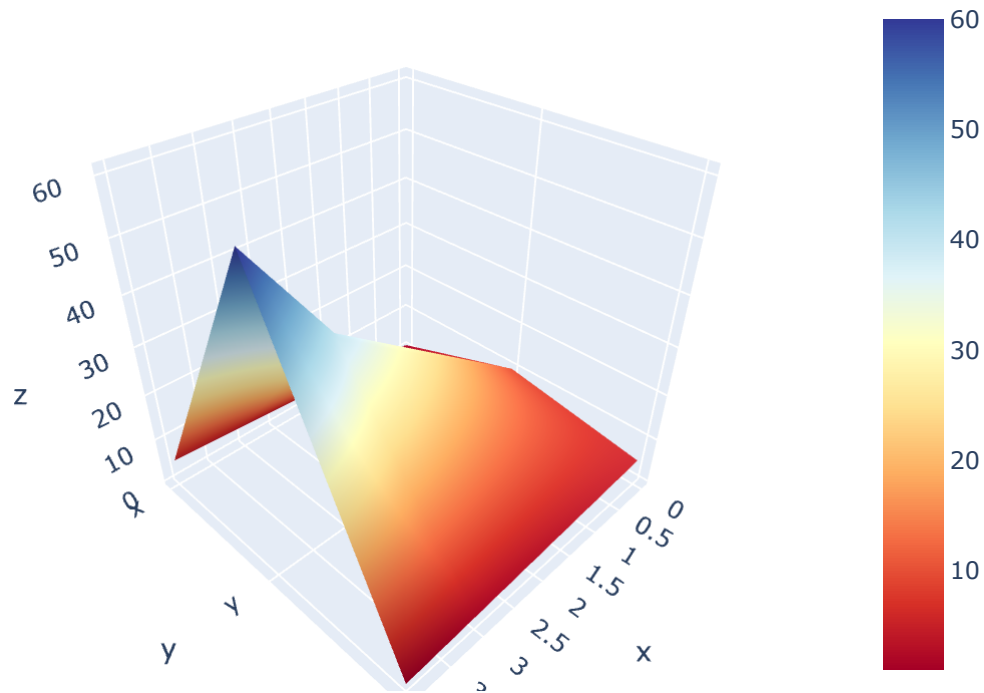
```
df3 = pd.DataFrame({'x':[1,2,3,4,5],  
                    'y':[10,20,30,40,60],  
                    'z':[5,4,3,2,1]})  
df3
```

Out[54]:

| | x | y | z |
|---|---|----|---|
| 0 | 1 | 10 | 5 |
| 1 | 2 | 20 | 4 |
| 2 | 3 | 30 | 3 |
| 3 | 4 | 40 | 2 |
| 4 | 5 | 60 | 1 |

In [55]:

```
df3.ipyplot(kind='surface',colorscale='rdylbu')
```



[Export to plotly](#)

cufflinks.datagen module

- ref : <https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>
(<https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>)
- datagen.lines : scatter(lines) plot을 위한 데이터 프레임 반환
- cufflinks.datagen.lines(n_traces=5, n=100, ...)
 - n_traces:int -> traces의 수
 - n : 각각의 점의 수

Line Charts

In [56]:

```
df = cf.datagen.lines()  
df.shape
```

Out[56]:

(100, 5)

In [57]:

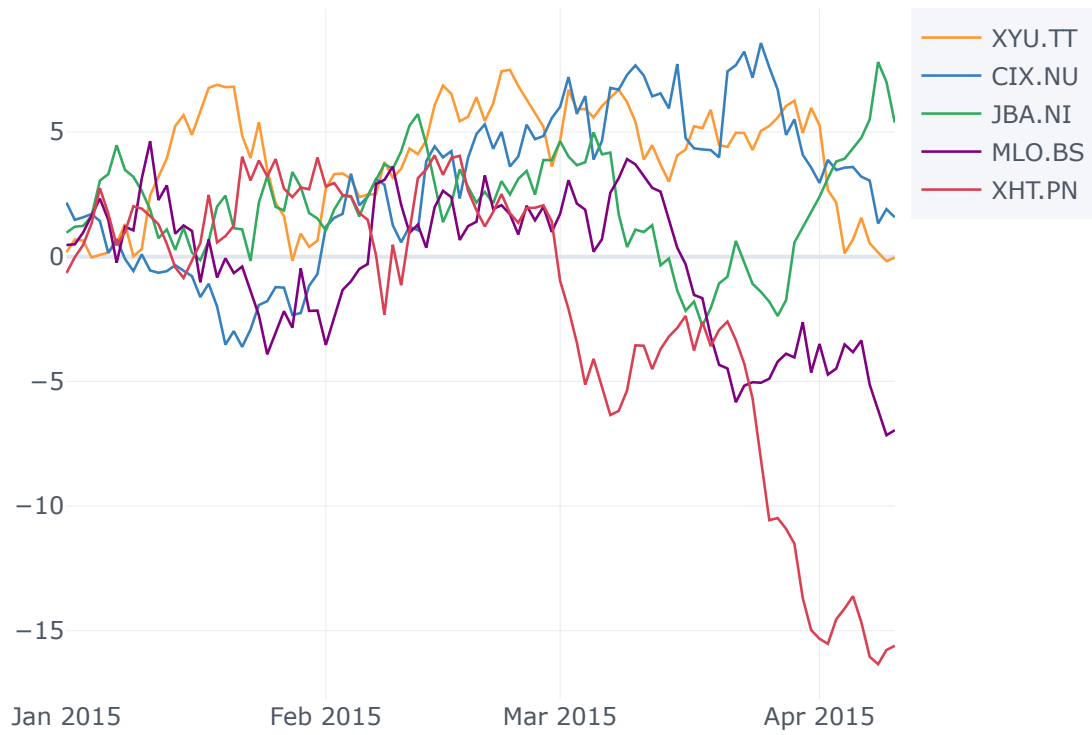
```
df.head(13)
```

Out[57]:

| | XYU.TT | CIX.NU | JBA.NI | MLO.BS | XHT.PN |
|------------|-----------|-----------|----------|-----------|-----------|
| 2015-01-01 | 0.184927 | 2.171877 | 0.964836 | 0.470304 | -0.655137 |
| 2015-01-02 | 0.689656 | 1.479998 | 1.206208 | 0.499690 | -0.020399 |
| 2015-01-03 | 0.649151 | 1.582318 | 1.235998 | 0.971387 | 0.475011 |
| 2015-01-04 | -0.029342 | 1.714909 | 1.639274 | 1.691653 | 1.336021 |
| 2015-01-05 | 0.075288 | 1.429631 | 3.053898 | 2.331488 | 2.750114 |
| 2015-01-06 | 0.170091 | 0.152632 | 3.314450 | 1.480077 | 1.732689 |
| 2015-01-07 | 0.544495 | 0.710863 | 4.471726 | -0.238790 | 0.461243 |
| 2015-01-08 | 1.312028 | -0.088384 | 3.482218 | 1.203644 | 0.990346 |
| 2015-01-09 | 0.018649 | -0.573471 | 3.206584 | 1.058521 | 2.023526 |
| 2015-01-10 | 0.308142 | 0.097888 | 2.642304 | 3.135890 | 1.931340 |
| 2015-01-11 | 2.460953 | -0.551436 | 1.893967 | 4.630392 | 1.629478 |
| 2015-01-12 | 3.275820 | -0.642213 | 0.742785 | 2.274207 | 1.300788 |
| 2015-01-13 | 3.948288 | -0.578341 | 1.094794 | 2.862953 | 0.520975 |

In [58]:

```
df.iplot(kind='line')
```



[Export to plot.ly](#)

In [59]:

```
print(df.shape)
df.head(10)
```

(100, 5)

Out[59]:

| | XYU.TT | CIX.NU | JBA.NI | MLO.BS | XHT.PN |
|-------------------|-----------|-----------|----------|-----------|-----------|
| 2015-01-01 | 0.184927 | 2.171877 | 0.964836 | 0.470304 | -0.655137 |
| 2015-01-02 | 0.689656 | 1.479998 | 1.206208 | 0.499690 | -0.020399 |
| 2015-01-03 | 0.649151 | 1.582318 | 1.235998 | 0.971387 | 0.475011 |
| 2015-01-04 | -0.029342 | 1.714909 | 1.639274 | 1.691653 | 1.336021 |
| 2015-01-05 | 0.075288 | 1.429631 | 3.053898 | 2.331488 | 2.750114 |
| 2015-01-06 | 0.170091 | 0.152632 | 3.314450 | 1.480077 | 1.732689 |
| 2015-01-07 | 0.544495 | 0.710863 | 4.471726 | -0.238790 | 0.461243 |
| 2015-01-08 | 1.312028 | -0.088384 | 3.482218 | 1.203644 | 0.990346 |
| 2015-01-09 | 0.018649 | -0.573471 | 3.206584 | 1.058521 | 2.023526 |
| 2015-01-10 | 0.308142 | 0.097888 | 2.642304 | 3.135890 | 1.931340 |

테마설정

In [60]:

```
themes = cf.getThemes()
themes
```

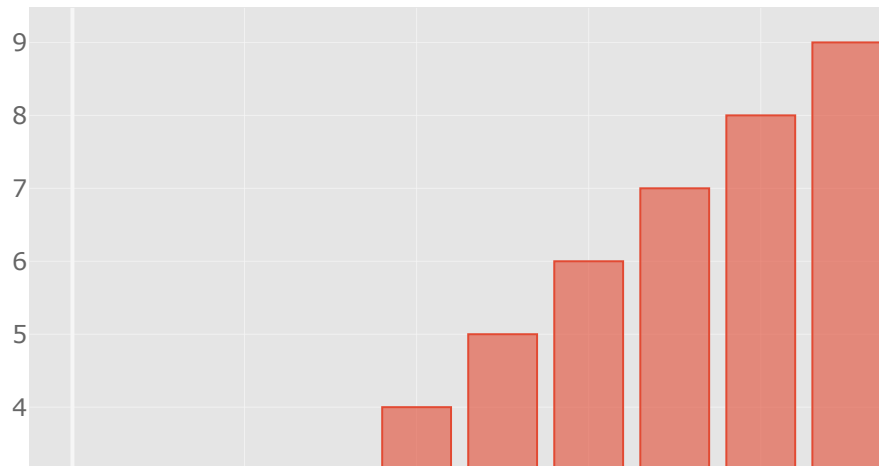
Out[60]:

['ggplot', 'pearl', 'solar', 'space', 'white', 'polar', 'henanigans']

In [61]:

```
data = pd.Series(range(10))  
for theme in themes:  
    data.iplot(kind='bar', theme=theme, title=theme)
```

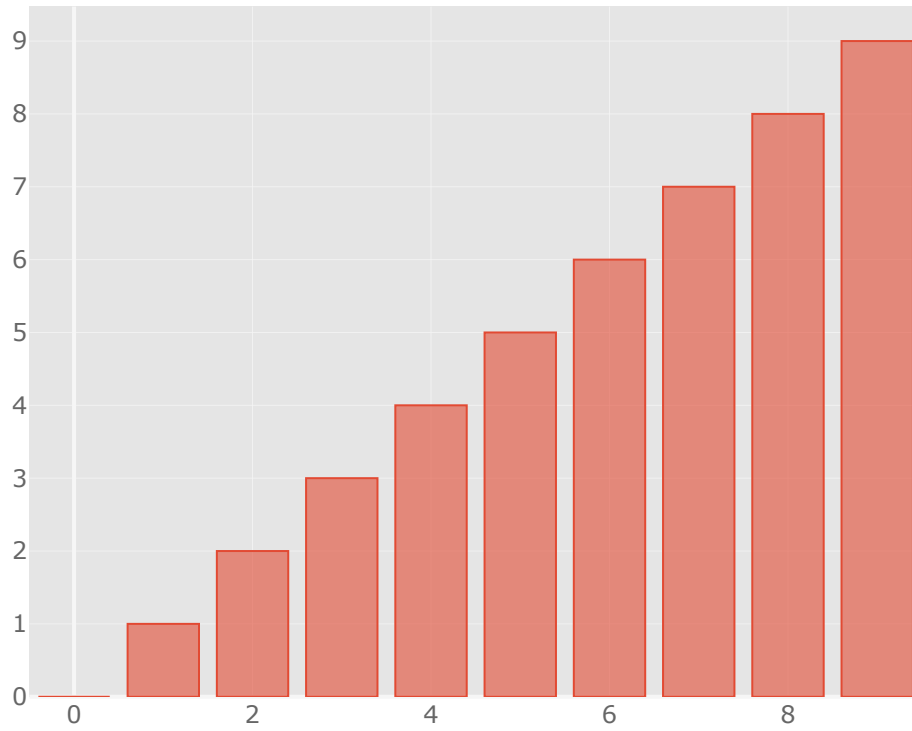
ggplot



In [62]:

```
data.iplot(kind='bar', theme="ggplot", title="ggplot")
```

ggplot

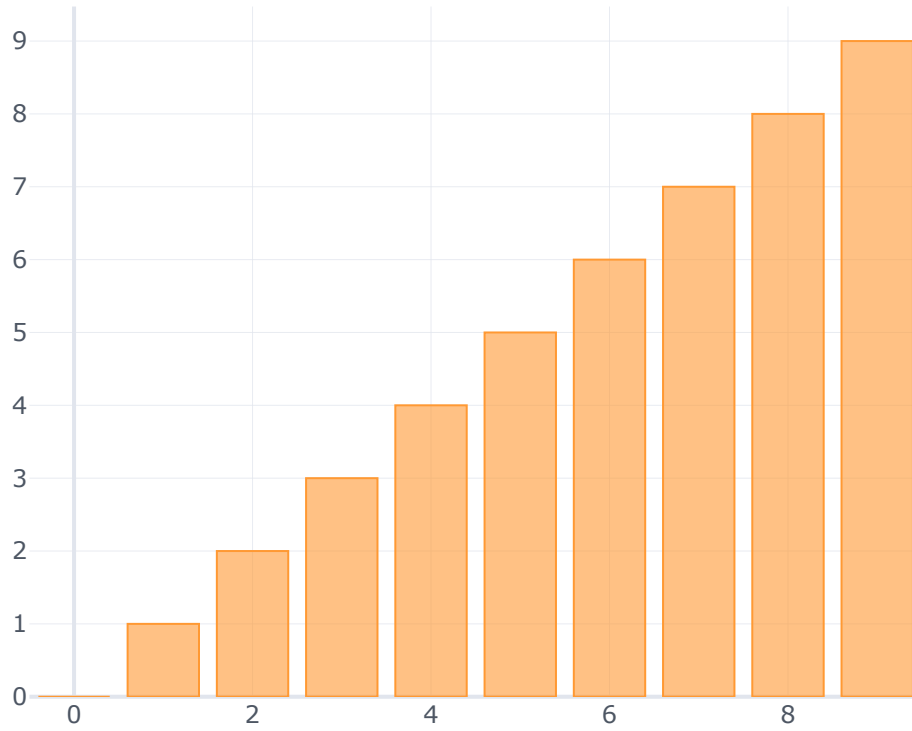


[Export to plot.ly](#)

In [63]:

```
data.iplot(kind='bar', theme="pearl", title="pearl")
```

pearl

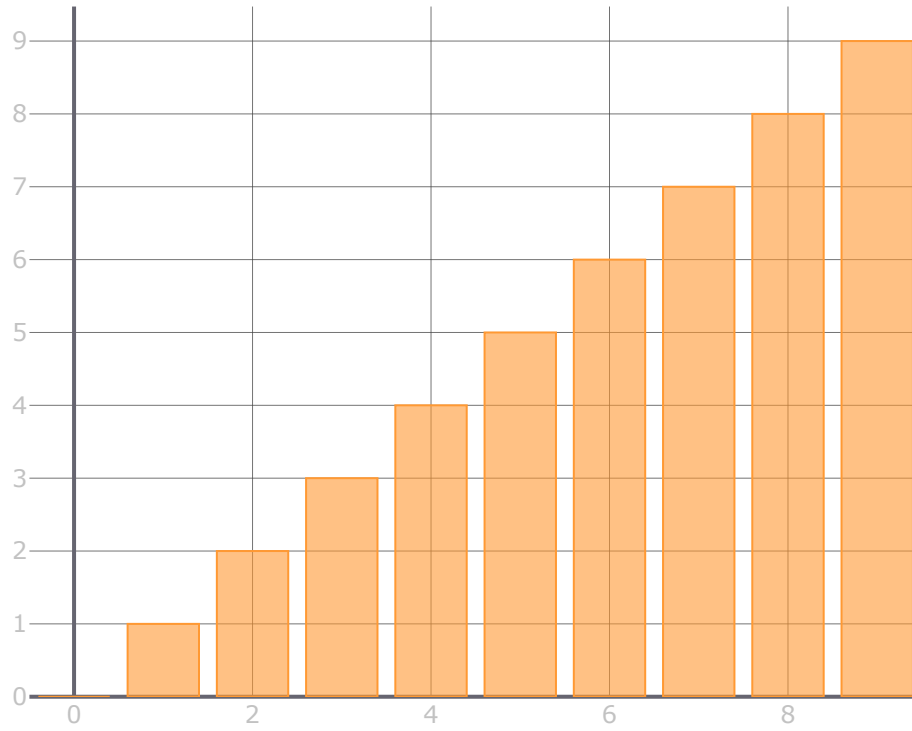


[Export to plotly](#)

In [64]:

```
data.iplot(kind='bar', theme="solar", title="solar")
```

solar

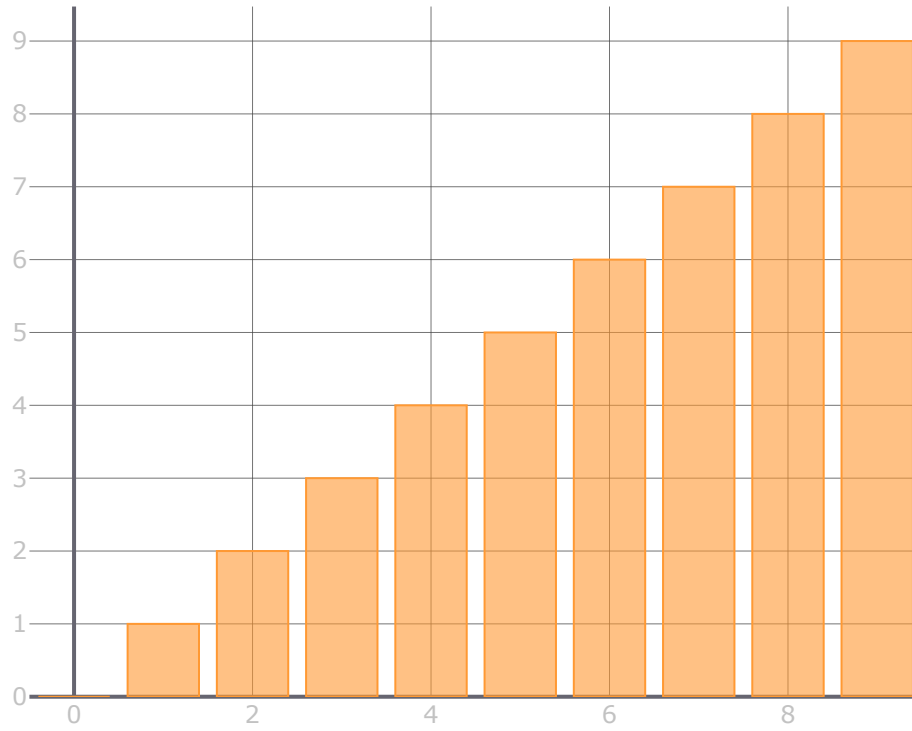


[Export to plot.ly](#)

In [65]:

```
data.iplot(kind='bar', theme="space", title="space")
```

space

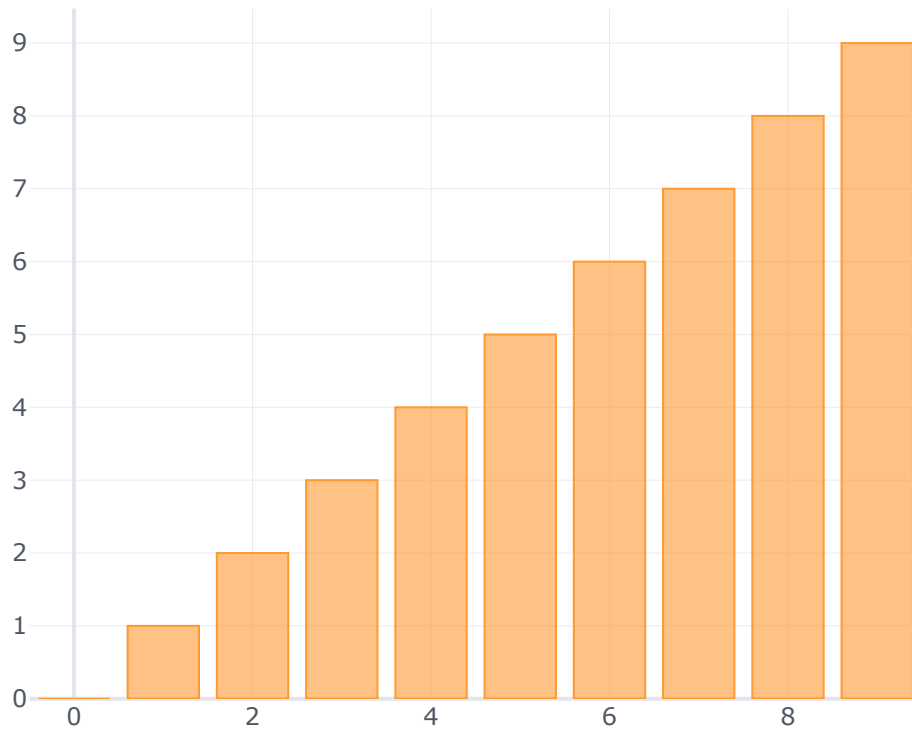


[Export to plot.ly](#)

In [66]:

```
data.ipplot(kind='bar', theme="white", title="white")
```

white

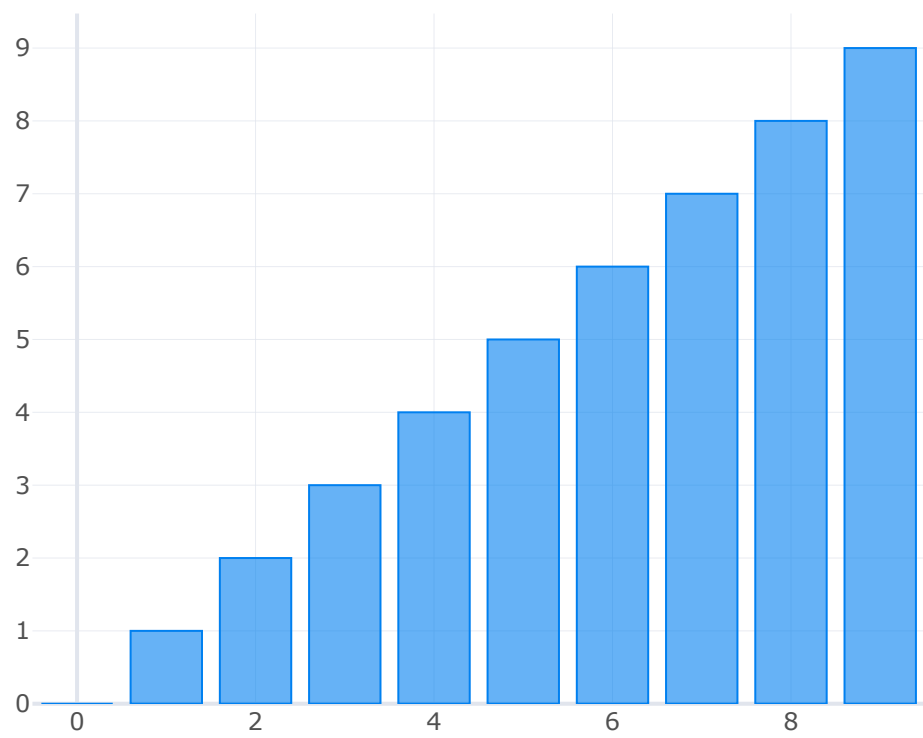


[Export to plot.ly](#)

In [67]:

```
data.ipplot(kind='bar', theme="polar", title="polar")
```

polar

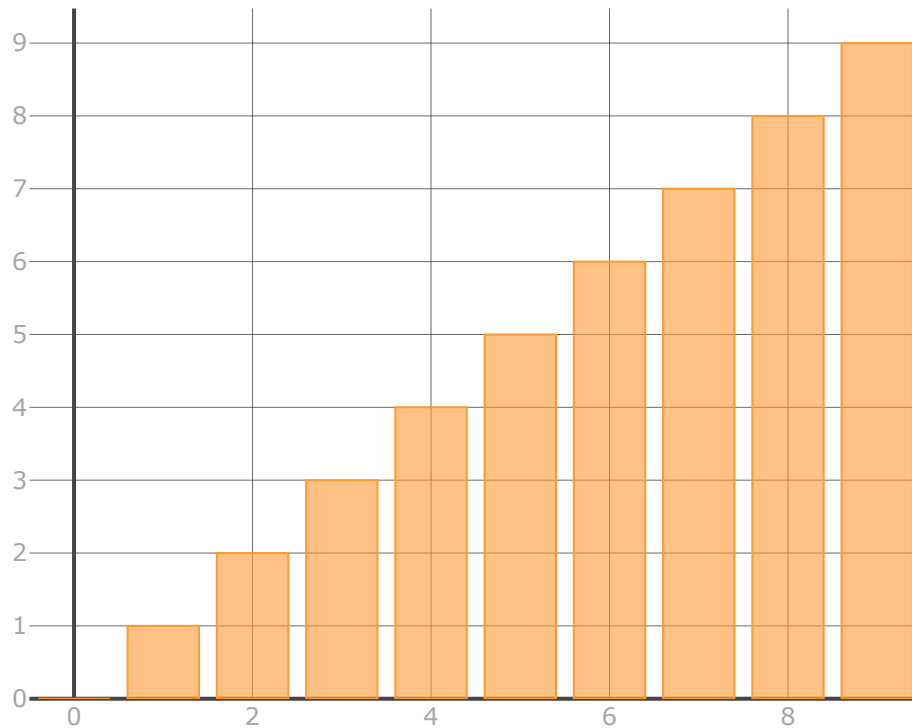


[Export to plot.ly](#)

In [68]:

```
data.ipplot(kind='bar', theme="henanigans", title="henanigans")
```

henanigans



Export to plot.ly

REF

- cufflinks.datagen module
- <https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>
(<https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>)
(<https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>)
(<https://jpoles1.github.io/cufflinks/html/cufflinks.datagen.html>)