```
#n = 1000
#1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,.....,1000

def generate_natural_numbers(n):
    numbers=[]
    for i in range(1,n+1):
        numbers.append(i)
    return numbers


generate_natural_numbers(10)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

```
x= generate_natural_numbers(10000000)#dhereei data deyo bhane computer nai freeze hunchha


# so we use generator for performance optimization


#generator vanekei iterator ho yesko value directly access hudenaa ki ta list lagayeraa or next,next gareraa or for loop layeraa  balla acces


def hello_fun(a,b):
    x = a+b
    yield x #generator => does not exist / teminate
    y=a-b
    yield y


hello_fun(4,2)
```

```
<generator object hello_fun at 0x0000026D4FE9C6C0>
```

```
list(hello_fun(4,2))
```

```
[6, 2]
```

```
for i in hello_fun(4,2):
    print(i)
```

```
6
2
```

```
x=hello_fun(4,2)


next(x)
```

```
6
```

```
next(x)
```

```
2
```

```
 # generator ma directly value aaudena kita loop lagaunu paryo kita next garnu paryo kitaa list lagaunu paryo


def hello_fun(a,b): #lazy loading (jati belaa chaiyo teti belaa value denchhaa)
    x = a+b
    yield x #generator => does not exist / teminate
    y=a-b
    yield y


x=hello_fun(4,2)



next(x)
```

```
6
```

```
print("hello world")
```

```
hello world
```

```
#bichma hello world print garemm aaba feri next (x ) gardaa yield y ko valuee denchhaa jati bellaa chaiyo teti belaa value denye vayo
```

```
next(x)
```

```
2
```

```
print("hiiii")
```

```
hiiii
```

```
#n = 1000
#1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,.....,1000 #using generator
```

```
def generate_natural_numbers(n):
    for i in range(1,n+1):
         yield i
```

```
x = generate_natural_numbers(5673874983)
```

```
next(x)
```

```
1
```

```
next(x)
```

```
2
```

```
#yesari value chaiyeko belama next(x) gardei load garnaa melchhaa
```

```
#pandas basic
```

```
#stock prices for a week
import pandas as pd

stock_prices = pd.Series([150,152,153,155,157],index = ['Monday','Tuesday','Wednesday','Thursday','Friday'])
print(stock_prices)
```

```
Monday       150
Tuesday      152
Wednesday    153
Thursday     155
Friday       157
dtype: int64
```

```
#creating dataframes
```

```
import pandas as pd
#creating a dataframe from dictionary
data = {
    'Name':['Alice','Bob','charlie'],
    'Age':[25,30,35],
    'city':['New York','Los Angeles','Chicago']
}
df = pd.DataFrame(data)
```

```
df
```

	Name	Age	city
0	Alice	25	New York
1	Bob	30	Los Angeles
2	charlie	35	Chicago

```
#selecting a single column
df['Name']
```

```
0      Alice
1        Bob
2    charlie
Name: Name, dtype: object
```

```
#selecting multiple column
df[['Name','city']]
```

	Name	city
0	Alice	New York
1	Bob	Los Angeles
2	charlie	Chicago

```
#selecting row by index
df.iloc[0]
```

```
Name       Alice
Age           25
city    New York
Name: 0, dtype: object
```

```
df.iloc[1:3] #Rows 1 to 2
```

	Name	Age	city
1	Bob	30	Los Angeles
2	charlie	35	Chicago

```
#Filtering data
#filter rows where age > 25
filtered_df = df [df['Age']>25]
filtered_df
```

	Name	Age	city
1	Bob	30	Los Angeles
2	charlie	35	Chicago

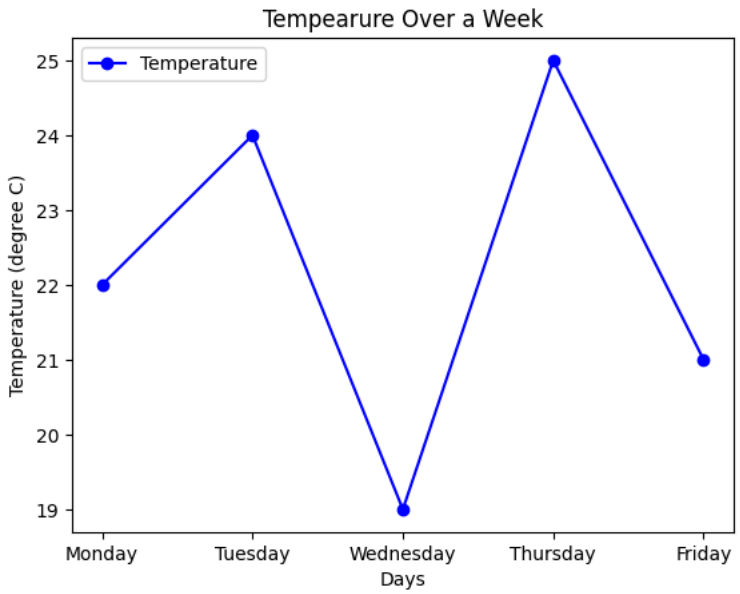
```
#Adding a new column
df['Country']= ['USA','USA','USA']
df
```

	Name	Age	city	Country
0	Alice	25	New York	USA
1	Bob	30	Los Angeles	USA
2	charlie	35	Chicago	USA

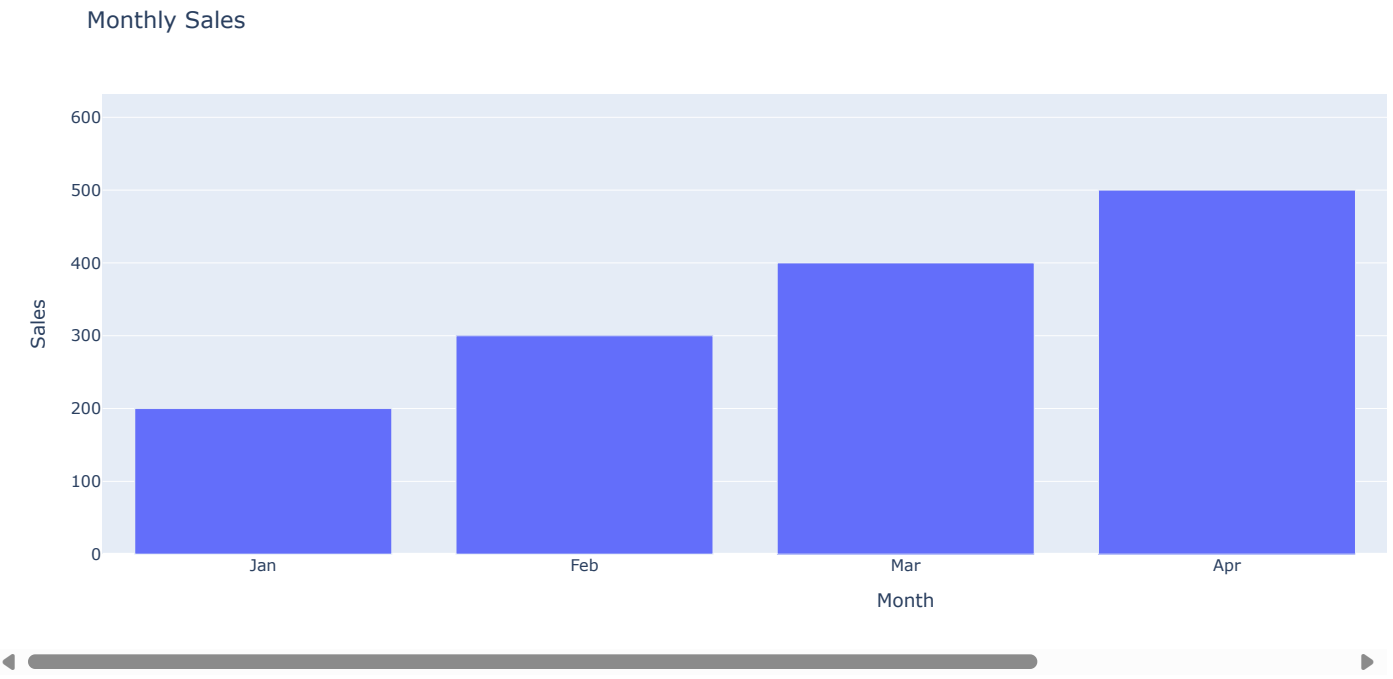
```
#save dataframe to a csv file
df.to_csv('output.csv',index = False)
```

```
import matplotlib.pyplot as plt
#data
days = ['Monday','Tuesday','Wednesday','Thursday','Friday']
temperatures = [22,24,19,25,21]

#create the plot
plt.plot(days,temperatures, marker='o',color='blue',label ='Temperature')

#Add title and lebels
plt.title('Tempearure Over a Week')
plt.xlabel('Days')
plt.ylabel('Temperature (degree C)')
```

```
plt.legend()

#show the plot
plt.show()
```




```
import plotly.express as px
import pandas as pd

#sample data
data = {
    'Month':['Jan','Feb','Mar','Apr','May'],
    'Sales':[200,300,400,500,600]
}
df = pd.DataFrame(data)

#Create a line plot
fig = px.bar(df, x='Month',y='Sales',title='Monthly Sales')
fig.show()
```




```
fruits = ['apple','banana','mango','orange']
for i in enumerate(fruits):
    print(i)
```

```
(0, 'apple')
(1, 'banana')
(2, 'mango')
(3, 'orange')
```

Start coding or generate with AI.