

1. BASIC OPERATORS

LOGICAL OPERATORS	DESCRIPTION	SYNTAX
logical_and	True if both operands are true	logical_and(x1,x2)
logical_or	True if one operands are true	logical_or(x1, x2)
logical_not	True if operand is false	logical_not(x)
logical_xor	Compute truth value of x1 XOR x2	logical_xor(x1, x2)

LOGIC FUNCTIONS	DESCRIPTION	SYNTAX
1. Truth value testing		
all	All array elements along given axis are True	all(a[, axis, out, keepdims])
any	Any array elements along given axis are True	any(a[, axis, out, keepdims])
2. Array contents		
isfinite	Test element-wise finiteness	isfinite(x[,f, out, where, casting, order])
isinf	Test element-wise for positive or negative infinity	isinf(x[,f, out, where, casting, order])
isnan	Test element-wise for NaN and return result as a boolean array	isnan(x[,f, out, where, casting, order])
isneginf	Test element-wise for negative infinity	isneginf(x[,out])
isposinf	Test element-wise for positive infinity	isposinf(x[,out])
3. Array type testing		
iscomplex	Return True if input element is complex	iscomplex(x)
iscomplexobj	Check for a complex number array	iscomplexobj(x)
isfortran	Return True if the array is fortran contiguous	isfortran(a)
isreal	Return True if input element is real	isreal(x)
isrealobj	Return True if x is a not complex	isrealobj(x)
isscalar	Return True if the type of num is scalar type	isscalar(num)
4. Comparison		
allclose	Return True if two array are element wise equal	allclose(a1, a2)
isclose	Return a boolean array if two array are element wise equal	isclose(a1, a2)
array_equal	True if two arrays have same shape and elements	array_equal(a1, a2)
array_equiv	Return True if input array are shape consistent and all elements equal	array_equiv(a1, a2)
greater	Return True if (x1>x2)	greater(x1, x2)
greater_equal	Return True if (x1>=x2)	greater_equal(x1, x2)
less	Return True if (x1<x2)	less(x1, x2)
less_equal	Return True if (x1<=x2)	less_equal(x1, x2)
equal	Return True if (x1==x2)	equal(x1, x2)
not_equal	Return True if (x1!=x2)	not_equal(x1, x2)

2. BUILT-IN FUNCTIONS IN NUMPY

FUNCTION	DESCRIPTION
1. Basic function	
ndim()	Return number of axes or rank of the array
shape()	Return tuple containing dimensionality of dataframe
size()	Total number of elements
2. Trigonometric function	
sin()	Trigonometric sine
cos()	Trigonometric cosine
tan()	Trigonometric tangent
arcsin()	Inverse sine
arccos()	Inverse cosine
arctan()	Inverse tangent
hypot()	Returns hypotenuse of right angled triangle
degree()	Convert angles from radians to degrees
radianns()	Convert angles from degrees to radians
3. Rounding	
around()	Evenly round to the given number of decimal
round()	Round an array to the given number of decimals
floor()	Return floor of input
ceil()	Return the ceiling of the input
truncate()	Return truncate value of the input before and after some index
4. Sums, products, differences	
prod()	Return product of array elements
sum()	Return sum of array elements
nanprod()	Return product of array elements over a given axis treating a not a number as ones
nansum()	Returns sum of array elements over a given axis treating not a number as zero
diff()	Calculate nth discrete difference along given axis
gradient	Return gradient of an n- dimensional array
cross	Returns cross product of two vectors
5. Exponents and logarithms	
exp()	Calculate exponential of all elements
log()	Natural logarithm element- wise
6. Arithmetic operations	
add()	Add arguments
reciprocal()	Return reciprocal of arguments
negative()	Numerical negative
multiply()	Multiply arguments
divide()	Divide arguments element wise
power()	First array element raised to powers from second array
subtract()	Subtract arguments element wise

3. BUILT IN FUNCTIONS IN PANDAS

FUNCTION	DESCRIPTION
add()	Returns addition of dataframe and other, element-wise (binary operator add)
sub()	Returns subtraction of dataframe and other, element-wise (binary operator sub)
mul()	Returns multiplication of dataframe and other, element-wise (binary operator mul)
div()	Returns floating division of dataframe and other, element-wise (binary operator truediv)
unique()	Method extracts the unique values in the dataframe
nunique()	Returns count of the unique values in the dataframe
value_counts()	Method counts the number of times each unique value occurs within the Series
columns()	Returns the column labels of the dataframe
axes()	Returns a list representing the axes of the dataframe
isnull()	Method creates a Boolean Series for extracting rows with null values
notnull()	Method creates a Boolean Series for extracting rows with non-null values
between()	Method extracts rows where a column value falls in between a predefined range
isin()	Method extracts rows from a dataframe where a column value exists in a predefined collection
dtypes()	Returns a Series with the data type of each column. The result's index is the original dataframe's columns
astype()	Method converts the data types in a Series
values()	Returns only the values in the DataFrame will be returned, the axes labels will be removed
sort_values()- Set1, Set2	Method sorts a data frame in Ascending or Descending order of passed Column
sort_index()	Method sorts the values in a dataframe based on their index positions or labels instead of their values
loc[]	Method retrieves rows based on index label
iloc[]	Method retrieves rows based on index position
ix[]	Method retrieves dataframe rows based on either index label or index position.
rename()	Method is called on a dataframe to change the names of the index labels or column names
columns()	Method is an alternative attribute to change the column name
drop()	Method is used to delete rows or columns from a dataframe
pop()	Method is used to delete rows or columns from a dataframe
sample()	Method pulls out a random sample of rows or columns from a dataframe
nsmallest()	Method pulls out the rows with the smallest values in a column
nlargest()	Method pulls out the rows with the largest values in a column
shape()	Returns a tuple representing the dimensionality of the DataFrame
ndim()	Returns an 'int' representing the number of axes / array dimensions. Returns 1 if Series, otherwise returns 2 if DataFrame
dropna()	Method allows the user to analyze and drop Rows/Columns with Null values in different ways
fillna()	Method manages and let the user replace NaN values with some value of their own
rank()	Values in a Series can be ranked in order with this method
query()	Method is an alternate string-based syntax for extracting a subset from a DataFrame
copy()	Method creates an independent copy of a pandas object
uplicated()	Method creates a Boolean Series and uses it to extract rows that have duplicate values
drop_duplicates()	Method is an alternative option to identifying duplicate rows and removing them through filtering
set_index()	Method sets the DataFrame index (row labels) using one or more existing columns
reset_index()	Method resets index of a Data Frame. This method sets a list of integer ranging from 0 to length of data as index
where()	Method is used to check a Data Frame for one or more condition and return the result accordingly.
merge	Merge dataframe or series by join
concat()	Concatenate pandas objects along a particular axis
get_dummies()	Convert categorical variables into dummy variables
isna()	Detect missing values for an array
isnull()	Detect missing values for an array
count()	Number of non-NA observations
mean()	Mean of values
median ()	Arithmetic median of values
max()	Maximum number
min()	Minimum number

4. READING FILES FROM SOURCES

FUNCTION	DESCRIPTION
read_csv	Load delimited data from a file, URL, or file-like object; use comma as default delimiter
read_table	Load delimited data from a file, URL, or file-like object; use tab ('\t') as default delimiter
read_excel	Read tabular data from an Excel XLS or XLSX file
read_hdf	Read HDF5 files written by pandas
read_html	Read all tables found in the given HTML document
read_json	Read data from a JSON (JavaScript Object Notation) string representation
read_sql	Read the results of a SQL query (using SQLAlchemy) as a pandas DataFrame

5. SUMMARY STATISTICS

STATISTIC	DESCRIPTION	SYNTAX
Mean	The sum of a collection of numbers divided by the count of numbers in the collection.	statistics.mean()
Median	The middle most value in the list of numbers	statistics.median()
Mode	The value in the list of numbers which has the highest frequency	statistics.mode()
Variance	Measure of dispersion of a set of values from the mean.	statistics.variance()
Standard Deviation	Measure of the amount of variation or dispersion of a set of values from the mean. (square of v	statistics.stdev()
Skewness	Skewness is a measure of the symmetry in a distribution.	scipy.stats.skew()
Kurtosis	Measure of peakedness (or flatness)	scipy.stats.kurtosis()
Correlation	Finds correlation of numeric features	corr()

6. VISUALIZATION

Matplotlib

import matplotlib.pyplot as plt

CHART	DESCRIPTION
acorr	Plot the autocorrelation of x
annotate	Annotate the point xy with text text
arrow	Add an arrow to the axes
axes	Add an axes to the current figure and make it the current axes
axhline	Add a horizontal line across the axis
axhspan	Add a horizontal span (rectangle) across the axis
axis	Convenience method to get or set some axis properties
axvline	Add a vertical line across the axes
axvspan	Add a vertical span (rectangle) across the axes.
bar	Make a bar plot.
barbs	Plot a 2D field of barbs.
barh	Make a horizontal bar plot.
box	Turn the axes box on or off on the current axes.
boxplot	Make a box and whisker plot.
broken_barh	Plot a horizontal sequence of rectangles.
clf	Clear the current figure.
clim	Set the color limits of the current image.
close	Close a figure window.
cohere	Plot the coherence between x and y.
colorbar	Add a colorbar to a plot.
eventplot	Plot identical parallel lines at the given positions.
figlegend	Place a legend on the figure.
fignum_exists	Return whether the figure with the given id exists.
figtext	Add text to figure.
figure	Create a new figure.
gca	Get the current Axes instance on the current figure matching the given keyword args, or create one.
gcf	Get the current figure.
grid	Configure the grid lines.
hexbin	Make a hexagonal binning plot.
hist	Plot a histogram.
hist2d	Make a 2D histogram plot.
legend	Place a legend on the axes.
locator_params	Control behavior of major tick locators.
loglog	Make a plot with log scaling on both the x and y axis.
magnitude_spectrum	Plot the magnitude spectrum.
margins	Set or retrieve auto scaling margins.
matshow	Display an array as a matrix in a new figure window.
minorticks_off	Remove minor ticks from the axes.
minorticks_on	Display minor ticks on the axes.
pause	Pause for interval seconds.
pcolor	Create a pseudocolor plot with a non-regular rectangular grid.
pcolormesh	Create a pseudocolor plot with a non-regular rectangular grid.
phase_spectrum	Plot the phase spectrum.
pie	Plot a pie chart.
plot	Plot y versus x as lines and/or markers.
plot_date	Plot data that contains dates.
savefig	Save the current figure.
sca	Set the current Axes instance to ax.
scatter	A scatter plot of y vs x with varying marker size and/or color.
show	Display a figure.
stackplot	Draw a stacked area plot.
stem	Create a stem plot.
step	Make a step plot.
subplot	Add a subplot to the current figure.
subplots	Create a figure and a set of subplots.
subplots_adjust	Tune the subplot layout.
supplitle	Add a centered title to the figure.
text	Add text to the axes.
thetagrids	Get or set the theta grid lines on the current polar plot.
tick_params	Change the appearance of ticks, tick labels, and gridlines.
ticklabel_format	Change the ScalarFormatter used by default for linear axes.
tight_layout	Automatically adjust subplot parameters to give specified padding.
title	Set a title for the axes.
twinx	Make and return a second axes that shares the x-axis.
twiny	Make and return a second axes that shares the y-axis.
violinplot	Make a violin plot.
vlines	Plot vertical lines.
xcorr	Plot the cross correlation between x and y.
xlabel	Set the label for the x-axis.
xlim	Get or set the x limits of the current axes.
xscale	Set the x-axis scale.
xticks	Get or set the current tick locations and labels of the x-axis.
ylabel	Set the label for the y-axis.
ylim	Get or set the y-limits of the current axes.
yscale	Set the y-axis scale.
yticks	Get or set the current tick locations and labels of the y-axis.

Seaborn**import seaborn as sns**

Chart	DESCRIPTION
catplot()	Figure-level interface for drawing categorical plots onto a FacetGrid.
stripplot()	Draw a scatterplot where one variable is categorical.
swarmplot()	Draw a categorical scatterplot with non-overlapping points.
boxplot()	Draw a box plot to show distributions with respect to categories.
violinplot()	Draw a combination of boxplot and kernel density estimate.
boxenplot()	Draw an enhanced box plot for larger datasets.
pointplot()	Show point estimates and confidence intervals using scatter plot glyphs.
barplot()	Show point estimates and confidence intervals as rectangular bars.
countplot()	Show the counts of observations in each categorical bin using bars.
relplot()	Figure-level interface for drawing relational plots onto a FacetGrid.
scatterplot()	Draw a scatter plot with possibility of several semantic groupings.
lineplot()	Draw a line plot with possibility of several semantic groupings.
jointplot()	Draw a plot of two variables with bivariate and univariate graphs.
pairplot()	Plot pairwise relationships in a dataset.
distplot()	Flexibly plot a univariate distribution of observations.
kdeplot()	Fit and plot a univariate or bivariate kernel density estimate.
rugplot()	Plot datapoints in an array as sticks on an axis.
lmpplot()	Plot data and regression model fits across a FacetGrid.
regplot()	Plot data and a linear regression model fit.
residplot()	Plot the residuals of a linear regression.
heatmap()	Plot rectangular data as a color-encoded matrix.
clustermap()	Plot a matrix dataset as a hierarchically-clustered heatmap.

Plotly**Import plotly as plotly**

Chart	DESCRIPTION
import plotly.express as px	
scatter()	Plots a scatter plot
line()	Plots a line plot
bar()	Plots a bar plot
violin()	Plots a violin plot of variables
box()	Plots a box plot
strip()	Plots a strip plot
histogram()	Plots a histogram
pie()	Plots a pie chart
import plotly.graph_objects as go	
Scatter()	Plots a scatter plot
Bar()	Plots a bar plot
Pie()	Plots a pie chart
Heatmap()	Plots a heatmap
Box()	Plots a box plot
Violin()	Plots a violin plot of variables
Histogram()	Plots a histogram