

「현업에 바로 적용할 수 있는 코드 90종 가이드」				
<div>▶ 제공되는 시나리오별 Ipython에 있는 기능들을 현업에 바로 사용하기 위해 기능 설명과 주요 명령어, 패키지를 기록하였습니다.</div> <div>▶ 현업에 필요한 기능이 있을 시 필터(Ctrl+F)를 통해 Chapter와 Session을 확인하시고 해당 Chapter00.ipynb 파일을 load하여 따라하며 현업에 적용 하시기 바랍니다.</div> <div>(※ 각 Chapter별 중복으로 사용되는 기능들은 한 번 만 기재하였습니다.)</div>				
Chapter	Session	기능(Function)	주요 명령어(Command)	패키지(Package)
01	01	· Data 형태 확인	· df.shape, df.info, df.describe, df.isnull().sum()	· import pandas as pd
		· Numreic(연속형) 변수 분포 확인	· sns.displot(), sns.distplot()	· import seaborn as sns
		· Plot size 조절	· plt.gcf().set_size_inches()	· import matplotlib.pyplot as plt
		· Unique한 Value별 카운팅	· df[].value_counts()	· import pandas as pd
	02	· 산점도(Scatter plot) 그리기	· sns.scatterplot()	· import seaborn as sns
	03	· Train/Test set 분할	· train_test_split()	· from sklearn.model_selection import train_test_split
		· 모델 학습 및 예측	· RandomForestClassifier()	· from sklearn.ensemble import RandomForestClassifier
		· 이진분류 모델 성능 확인	· classification_report()	· from sklearn.metrics import classification_report
		· 하이퍼 파라미터 튜닝	· GridSearchCV()	· from sklearn.model_selection import GridSearchCV
		· 중요 변수 파악(Feature Importance)	· model.feature_importances_	· from sklearn.ensemble import RandomForestClassifier
· 모델 Save & Read		· pickle.dumps(), pickle.loads()	· import pickle	
02	01	· 특정 값 해당하는 DataFrame 출력	· df[df[] ==]	· import pandas as pd
		· DataFrame 특정값 치환	· df.replace()	· import pandas as pd
		· Null 값 이전 값으로 채워넣기	· df.fillna(method='ffill')	· import pandas as pd
		· DataFrame 특정 col만 가져오기	· df[[]]	· import pandas as pd
		· 기본 line plot 그리기	· plt.plot()	· import matplotlib.pyplot as plt
	02	· for문 활용 distplot 다중 출력	· for i in range() : plt.subplot() sns.distplot() ...	· import matplotlib.pyplot as plt · import seaborn as sns
		· 다중 값 해당하는 DataFrame 출력	· df[(df[] >=) & (df[] <=)]	· import pandas as pd
	03	· Dataframe 유니크한 값 list 출력	· df[].unique()	· import pandas as pd
		· Datetype(시간형)으로 변경하기	· pd.to_datetime()	· import pandas as pd
		· DataFrame Index col값으로 만들기	· reset_index()	· import pandas as pd
03	01	· 이중 축 그래프 그리기	· ax.twinx()	· import matplotlib.pyplot as plt
		· 오름차순, 내림차순 정렬	· df[].sort_values(by=[], ascending=)	· import pandas as pd
		· 특정값이 포함된 Data 찾기	· str.contains()	· import pandas as pd
		· 상관계수 값 출력	· stats.pearsonr()	· import scipy.stats as stats
	02	· pairplot 상관관계 분석	· sns.pairplot()	· import seaborn as sns
		· Heat map 상관관계 분석	· sns.heatmap()	· import seaborn as sns
	03	· Regressor(회귀) 모델 학습 및 평가	· RandomForestRegressor() · mean_absolute_error(), r2_score()	· from sklearn.ensemble import RandomForestRegressor · from sklearn.metrics import mean_absolute_error, r2_score
04	01	· sep 활용 csv 파일 읽기	· pd.read_csv(, sep=' ')	· import pandas as pd
		· 특정 조건 만족하는 값, 변경하기	· np.where()	· import numpy as np
	02	· 표준화 및 PCA 차원축소	· StandardScaler(), PCA()	· from sklearn.decomposition import PCA, StandardScaler
		· 전처리 규칙 파이프라인 만들기	· make_pipeline()	· from sklearn.pipeline import make_pipeline
05	03	· light gbm 활용 모델링	· lgb.Dataset(), lgb.train()	· import lightgbm as lgb
		· 선형회귀 상관계수 확인	· model.coef_	· from sklearn.linear_model import LinearRegression
		· 선형회귀활용 모델링	· LinearRegression()	· from sklearn.linear_model import LinearRegression
06	01	· 연속형, 범주형 변수 list 나누기	· for i in df.columns : if df[i].dtypes == 'O' : categoical_list.append(i) else : numeric_list.append(i)	· import numpy as np
	02	· catplot 그리기	· sns.catplot()	· import seaborn as sns
		· groupby 활용 카운팅	· df[].groupby().value_counts()	· import pandas as pd
		· pivot table 활용 데이터 처리	· pd.pivot_table()	· import pandas as pd
07	01	· Excel 파일 읽기	· pd.read_excel()	· import pandas as pd
		· inf(무한대) 데이터 null 처리	· replace([np.inf, -np.inf], np.nan)	· import pandas as pd
	03	· AUROC score 출력하기	· roc_auc_score()	· from sklearn.metrics import roc_auc_score
08	02	· Unique한 데이터 숫자세기	· nunique()	· import pandas as pd
	03	· 누적 막대 그래프로 비중 표현	· get_bbox().bounds, ax.annotate	· import matplotlib.pyplot as plt
09	01	· apply lambda 활용 데이터 split	· df[].apply(lambda x: x.split()[])	· import pandas as pd
	03	· Lable encoder 활용 범주형 데이터 처리	· LabelEncoder()	· from sklearn.preprocessing import LabelEncoder
10	02	· 연속형 변수의 구간화	· np.where(, , np.where())	· import numpy as np
	03	· ROC 커브 그리기	· roc_curve_plot()	· from sklearn.metrics import roc_curve

「현업에 바로 적용할 수 있는 코드 90종 가이드」				
Chapter	Session	기능(Function)	주요 명령어(Command)	패키지(Package)
11	01	· 캔들스틱 주식 데이터 그리기	· candlestick2_ohlc()	· from mpl_finance import candlestick2_ohlc
	02	· 그래프 특정 값에 색상 입히기	· plt.bar(color=)	· import matplotlib.pyplot as plt
	03	· lag 데이터 생성	· shift(-1)	· import pandas as pd
12	01	· 중복 데이터 처리	· drop_duplicates()	· import pandas as pd
	03	· 회귀예측 문제 실제값, 예측값 fitting 확인	· sns.scatterplot()	· import seaborn as sns
13	01	· 한글 깨질시 데이터 읽기	· pd.read_csv(, encoding='cp949')	· import pandas as pd
		· 문자열 데이터 앞 공백 제거	· str.lstrip()	· import pandas as pd
14	01	· 위도, 경도 활용 특정 좌표 Mapping	· folium.Map()	· import folium
		· 위도, 경도 특정 좌표 Marker 그리기	· folium.Marker()	· import folium
		· Marker 클러스터 생성	· MarkerCluster()	· from folium.plugins import MarkerCluster
	02	· 데이터 left join	· pd.merge()	· import pandas as pd
	03	· Circle Marker 그리기	· folium.CircleMarker()	· from folium import Marker
15	01	· Recency(최근성) 데이터 생성	· apply(lambda x : (df[].max() - x).days)	· import pandas as pd
		· groupby.agg 활용 타겟률 추출	· groupby.agg()	· import pandas as pd
16	01	· 연관규칙 분석을 위한 list 데이터처리	· for i in range(len(df)): records.append([str(df.values[i,j]) for j in range(len(df.columns)) if not pd.isna(df.values[i,j])])	· import pandas as pd
		· Transaction Encoder 활용 전처리	· TransactionEncoder()	· from mlxtend.preprocessing import TransactionEncoder
	02	· 연관규칙 생성	· apriori()	· from mlxtend.frequent_patterns import apriori
	02	· 연관규칙 분석	· association_rules()	· from mlxtend.frequent_patterns import association_rules
17	03	· 다항회귀(비선형 패턴) 모델링	· PolynomialFeatures()	· from sklearn.preprocessing import PolynomialFeatures
		· 선형 Fitting 확인 및 그래프 그리기	· sns.scatterplot(), plt.xlim, plt.ylim	· import matplotlib.pyplot as plt · import seaborn as sns
18	01	· 특정 Col 기준으로 Drop	· df.dropna(subset=[])	· import pandas as pd
	02	· RFM 분석	· apply(lambda x : (df[].max() - x).days)	· import pandas as pd
	03	· min-max scale 활용 표준화	· minmax_scale()	· from sklearn.preprocessing import minmax_scale
		· 날짜 데이터 형식 변경	· dt.strftime("%Y-%m")	· import datetime
19	01	· index 순서 변경하기	· pd.DataFrame(, index=[])	· import pandas as pd
20	02	· 시계열 분해	· seasonal_decompose()	· from statsmodels.tsa.seasonal import seasonal_decompose
	03	· Seasonal Arima 활용 시계열 예측	· sm.tsa.statespace.SARIMAX()	· import statsmodels.api as sm
21	01	· list 중복 없애기	· set()	· import pandas as pd
	02	· for문 활용 dataframe 생성	· list.append()	· import pandas as pd
24	01	· barplot 순서에 맞게 출력하기	· sns.barplot(order=[])	· import matplotlib.pyplot as plt
25	01	· pandas dataframe 시각적 표현 Tip	· df.style.background_gradient(cmap='Reds')	· import pandas as pd
28	03	· t-test 활용 A/B test	· stats.ttest_ind()	· from scipy import stats
		· chi2 test 활용 A/B test	· stats.chi2_contingency()	· from scipy import stats
29	02	· isin, notnull 활용 데이터 찾기	· isin(), nounull()	· import pandas as pd