In [2]:	Requirement already satisfied: matplotlib in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (3.8.2) Requirement already satisfied: cycler>=0.10 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (0.11.0) Requirement already satisfied: pillow>=8 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (3.0.10) Requirement already satisfied: pyparsing>=2.3.1 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (3.0.9) Requirement already satisfied: packaging>=2.0.0 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (22.0) Requirement already satisfied: fonttools>=4.22.0 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (2.2.0) Requirement already satisfied: python-dateutils=2.7 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (2.8.2) Requirement already satisfied: kiwisolver>=1.3.1 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (1.4.4) Requirement already satisfied: numpy<2,>=1.21 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (1.2.3.5) Requirement already satisfied: contourpy>=1.0.1 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (1.0.5) Requirement already satisfied: six>=1.5 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (1.0.5) Requirement already satisfied: six>=1.5 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib) (1.0.6) Note: you may need to restart the kernel to use updated packages. %matplotlib inline은 Jupyter Notebook에서 자주 사용되는 매직 명령어로, 이 명령어는 matplotlib으로 생성한 그래프를 노트북 안에 인라인으로 표시되도록 합니다. 이를 통해 별도의 창을 열지 않고도 노트북 내에서 바로 그래프를 확인할 수 있습니다.
In []: In []: In [3]:	%matplotlib inline import matplotlib.pyplot as plt matplotlib을 이용할 때 매번 한글 폰트 설정을 해줘야하는 불편함이 있습니다. 그러면 "koreanize_matplotlib" 라이브러리를 설치하시고 불어주시면 됩니다.
In [2]: In [2]:	Requirement already satisfied: fonttools>=4.22.0 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib->koreanize-matplotlib) (4.25.0) Requirement already satisfied: python-dateutil>=2.7 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from matplotlib->koreanize-matplotlib) (2.8.2) Requirement already satisfied: six>=1.5 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from python-dateutil>=2.7->matplotlib->koreanize-matplotlib) (1.16.0) Note: you may need to restart the kernel to use updated packages. **matplotlib* inline import matplotlib.pyplot as plt import koreanize_matplotlib import pandas as pd import numpy as np
In []: In [4]:	categories = ['2021', '2022', '2023'] group_a = [5, 6, 4] group_b = [3, 2, 5] group_c = [2, 4, 3] # DataFrame 생성 # DataFrame 생성 # Comparison of the state of
Out[4]:	Department Revenue 0 HR 250 1 IT 300 2 Sales 400 3 Marketing 350 4 Finance 450
	PHI III Sales Marketing Finance
<pre>In [9]: Out[9]: In [11]:</pre>	
	plt.figure(flastze = (10.7)) plt.bar(df.stacked['vactegorias'],df.stacked['group_a'],color = 'bloe', label = 'a') plt.bar(df.stacked['vactegorias'],df.stacked['group_b'],bottom = df_stacked['group_a'],color = 'green',label = 'b') plt.bar(df.stacked['vactegorias'],df.stacked['group_c'],bottom = df_stacked['group_b'],color = 'green',label = 'c') plt.lagend() #4 다른 축성 보려되어 b c 8
<pre>In []: In [12]: Out[12]: In [13]:</pre>	그룹화 된 막대 차트 df_stacked
	stack[('group_a', 'group_b', 'group_c']].plot(kind = 'bar')#.plot()等於 kind等 bar(學術) 章 發音 plt.show() 6 6 7 7 8 8 8 6 6 6 7 8 8 8 6 6 6 7 8 8 8 8 6 6 7 8 8 8 8 8 8 8 8 8 8 8 8
In []: In [14]: Out[14]:	선 플롯 df_sales
In [26]:	Monthly Sales 400 350 250 200
In [15]:	January February March April May June Month
	Monthly Sales 400 350 250
In []:	200 January February March April May June Month
<pre>In [28]: In [10]: In [31]:</pre>	import numpy as np # 랜덤 시드 설정 (재현성을 위해) np.random.seed(42) # 대이터 생성 num_employees = 100 departments = ['HR', 'IT', 'Sales', 'Marketing', 'Finance'] salaries = { 'bepartment': np.random.choice(departments, num_employees), 'Salary': np.random.normal(50000, 15000, num_employees).astype(int) # 평균 50000, 표준편차 15000 } # DataFrame 생성 df_salaries = pd.DataFrame(salaries)
	Histogram of Employee Salaries 10 8 4 2
In [12]:	2000 3000 4000 5000 6000 7000 8000 9000 pit. Tigure(Tigsize = (10,7)) pit. hist(uf. salaries(Salary'), bins = 28, color = 'skyblue', edgecolor = 'black') pit. Jabel ('Salary') pit. Histogram of Employee Salaries") pit. show() Histogram of Employee Salaries
In [4]:	# 데이터셋 생성 np.random.seed(0) # 재현성을 위해 시드 고정 np.points = 100 x = np.linspace(0, 10, n_points) y = 2.5 * x + np.random.randn(n_points) * 2 # y는 x에 노이즈를 더한 값 colors = np.random.rand(n_points) # 생상 sizes = 100 * np.random.rand(n_points) # 37
In [5]:	#alpha는 도명 호텔 # cmap은 예상 저동 plt.scatter(x,y,c= colors, s=sizes, alpha = 0.5, cmap = 'viridis') plt.vlabel('\ % 값') plt.ylabel('\ % \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \
In [33]:	Subplots # 캠링 시트 설정 np.random.seed(42) # 데이터 생성 num_employees = 100 departments = ['HR', 'IT', 'Sales', 'Marketing', 'Finance'] salaries = {
In 「^	# 월벌 판매량 데이터셋 months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December'] sales = pg.random.randint(200, 500, len(months)) df_sales = pd.DataFrame({'Month': months, 'Sales': sales}) #제품별 시장 접유율 데이터셋 products = ['Product A', 'Product B', 'Product C', 'Product E'] market_share = pp.random.randint(10, 40, len(products)) df_market_share = pd.DataFrame({'Product': products, 'Market Share': market_share}) # 직원 연령 분포 데이터셋 ages = pp.random.normal(35, 10, num_employees).astype(int) df_ages = pd.DataFrame({'Age': ages}) fig, axs = plt.subplots(2, 2, figsize=(14, 10))
ın [34]:	fig, axs = plt.subplots(2, 2, figsize=(14, 10)) # 1. 무서별 영병 분포 하스토그램 (앤 원목의 원목) axs[0, 0].hist(df_salaries['salary'], bins=20, color='skyblue', edgecolor='black') axs[0, 0].set_title('Histogram of Employee Salaries') axs[0, 0].set_title('Histogram of Employee Salaries') axs[0, 0].set_ylabel('Frequency') # 2. 원발 판매한 라인 플로 (앤 워의 오르쪽) axs[0, 1].plot(df_sales['Month'], df_sales['Sales'], marker='o', linestyle='-', color='green') axs[0, 1].set_xlabel('Wonth') axs[0, 1].set_xlabel('Wonth') axs[0, 1].set_xlabel('Wonth') axs[0, 1].set_xlabel('Bales') # 3. 제품별 시장 점임을 파이 플론엔 아래의 왼쪽) axs[1, 0].set_title('Morket Share'), labels=df_market_share['Product'], autopct='%1.1f%%', colors=['gold', 'yellowgreen', 'lightcoral', 'lightskyblue', 'purple' axs[1, 0].set_title('Market Share by Product') # 4. 직원 연령 분포 박스물론(펜 아래의 오른쪽) axs[1, 1].set_title('Boxplot of Employee Ages') axs[1, 1].set_xlabel('Employees') axs[1, 1].set_xlabel('Employees') axs[1, 1].set_xlabel('Mage') plt.tight_layout() plt.tight_layout() plt.tight_layout()
	450 - 425 - 400 - 375 -
	Product A 9.8% 27.9% 50 - 50 - 50 - 50 - 50 - 50 - 50 - 50