**ONLINE ADVERTISING PERFORMANCE DATASET**

* **Sharon Blessy J V**

1. **What is the overall trend in user engagement throughout the campaign period?**

**Ans.**

***Code Snippet:***

*print(ad['user\_engagement'].value\_counts())*

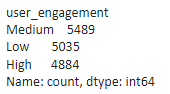
*sns.countplot(x='user\_engagement', hue='month', data=ad)*

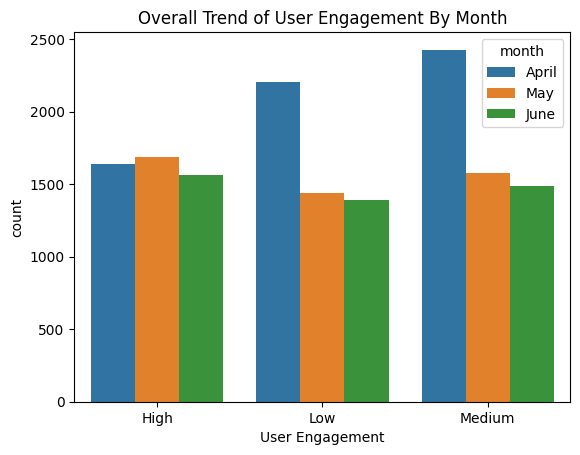
*plt.xlabel('User Engagement')*

*plt.title('Overall Trend of User Engagement By Month')*

*plt.show()*

**Output:**





1. **How does the size of the ad (banner) impact the number of clicks generated?**

**Ans.**

***Code Snippet:***

*click\_size\_count = ad.groupby('banner')['clicks'].count()*

*print(click\_size\_count)*

*sns.countplot(x='banner', data=ad)*

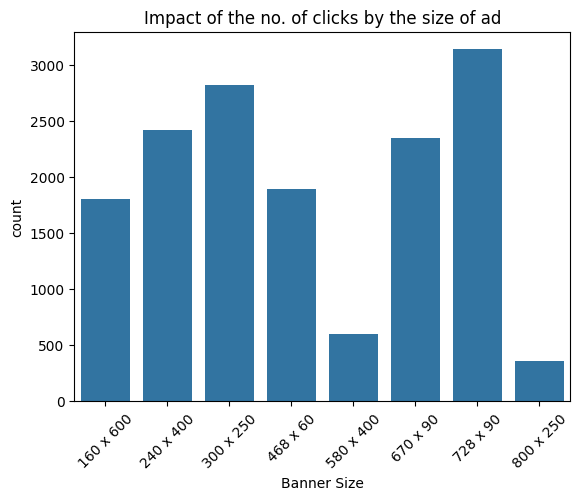
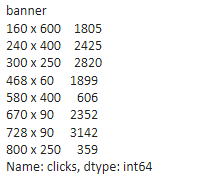
*plt.xlabel('Banner Size')*

*plt.title('Impact of the no. of clicks by the size of ad')*

*plt.xticks(rotation=45)*

*plt.show()*

**Output:**



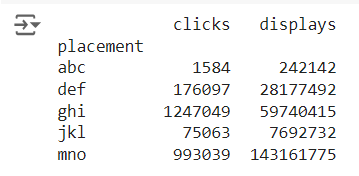
1. **Which publisher spaces (placements) yielded the highest number of displays and clicks?**

**Ans.**

***Code Snippet:***

*placement\_stats = ad.groupby('placement')[['clicks', 'displays']].sum()*

*print(placement\_stats)*

****MNO has the highest displays while GHI has the highest clicks.

1. **Is there a correlation between the cost of serving ads and the revenue generated from clicks?**

**Ans.**

***Code Snippet:***

*corr\_coeff = ad['cost'].corr(ad['revenue'])*

*print(f'Correlation coefficient of cost and revenue: {corr\_coeff}')*

*plt.scatter(ad['cost'], ad['revenue'])*

*plt.xlabel('Cost')*

*plt.ylabel('Revenue')*

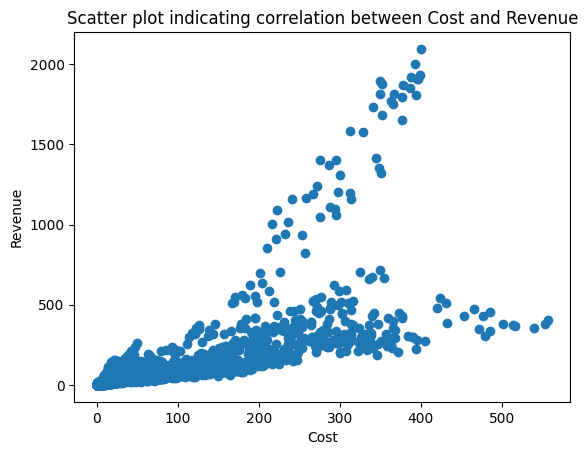
*plt.title('Scatter plot indicating correlation between Cost and Revenue')*

*plt.show()*

**Output:**



This indicates a strong positive relationship.

****

1. **What is the average revenue generated per click for Company X during the campaign period?**

**Ans.**

***Code Snippet:***

*tot\_revenue = ad['revenue'].sum()*

*tot\_clicks = ad['clicks'].sum()*

*avg\_revenue\_per\_click = tot\_revenue / tot\_clicks*

*print("Average revenue generated per click:", avg\_revenue\_per\_click)*

**Output:**

1. **Which campaigns had the highest post-click conversion rates?**

**Ans.**

***Code Snippet:***

*filter\_ad = ad[ad['clicks'] != 0]*

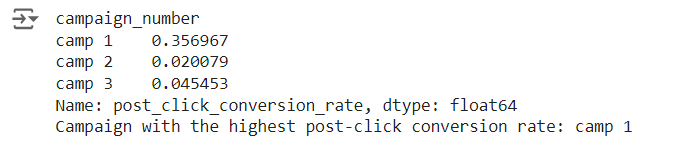
*filter\_ad['post\_click\_conversion\_rate'] = filter\_ad['post\_click\_conversions'] / filter\_ad['clicks']*

*campaign\_conversion\_rates = filter\_ad.groupby('campaign\_number')['post\_click\_conversion\_rate'].mean()*

*print(campaign\_conversion\_rates)*

*highest\_conversion\_campaign = campaign\_conversion\_rates.idxmax()*

*print("Campaign with the highest post-click conversion rate:", highest\_conversion\_campaign)*

**Output:**

1. **Are there any specific trends or patterns in post-click sales amounts over time?**

**Ans.**

***Code Snippet:***

*monthly\_sales = ad.groupby('month')['post\_click\_sales\_amount'].mean()*

*print(monthly\_sales)*

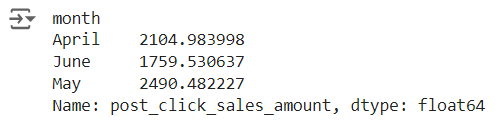
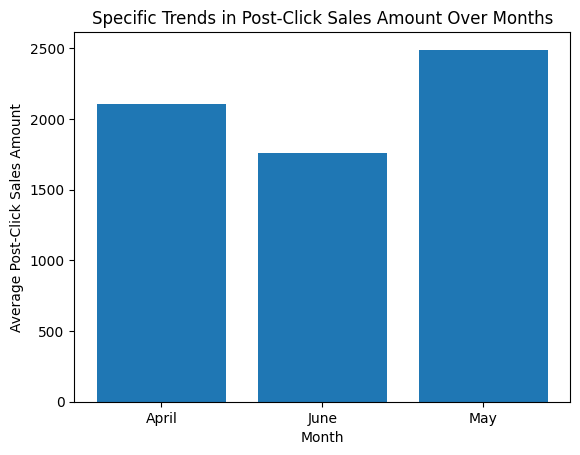
*plt.bar(monthly\_sales.index, monthly\_sales.values)*

*plt.xlabel('Month')*

*plt.ylabel('Average Post-Click Sales Amount')*

*plt.title('Specific Trends in Post-Click Sales Amount Over Months')*

*plt.show()*

****

1. **How does the level of user engagement vary across different banner sizes?**

**Ans.**

***Code Snippet:***

*engagement\_mapping = {'Low': 1, 'Medium': 2, 'High': 3}*

*ad['user\_engagement\_val'] = ad['user\_engagement'].map(engagement\_mapping)*

*engagement\_banner = ad.groupby('banner')['user\_engagement\_val'].mean()*

*print(engagement\_banner)*

*plt.bar(engagement\_banner.index, engagement\_banner.values)*

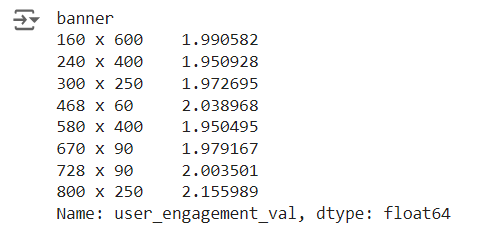
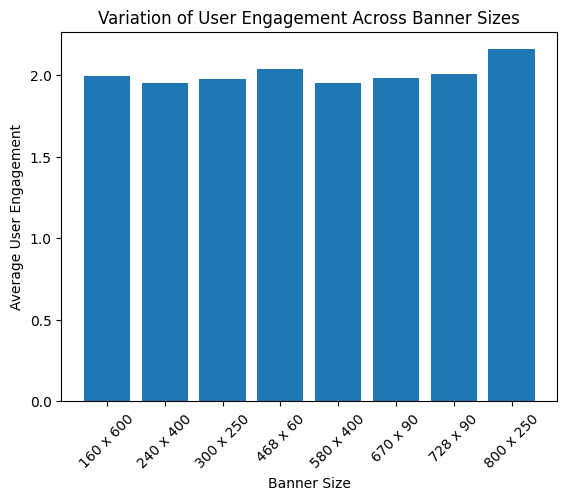
*plt.xlabel('Banner Size')*

*plt.ylabel('Average User Engagement')*

*plt.title('Variation of User Engagement Across Banner Sizes')*

*plt.xticks(rotation=45)*

*plt.show()*

****

1. **Which placement types result in the highest post-click conversion rates?**

**Ans.**

***Code Snippet:***

*filter\_ad = ad[ad['clicks'] != 0]*

*filter\_ad['post\_click\_conversion\_rate'] = filter\_ad['post\_click\_conversions'] / filter\_ad['clicks']*

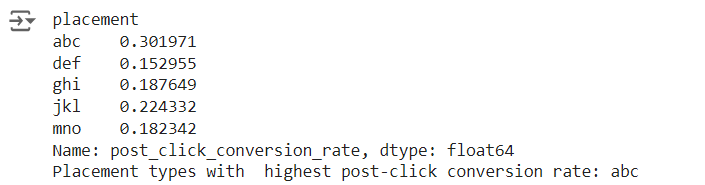
*placement\_conversion\_rates = filter\_ad.groupby('placement')['post\_click\_conversion\_rate'].mean()*

*print(placement\_conversion\_rates)*

*highest\_conversion\_placement = placement\_conversion\_rates.idxmax()*

*print("Placement types with  highest post-click conversion rate:", highest\_conversion\_placement)*

**Output:**



1. **Can we identify any seasonal patterns or fluctuations in displays and clicks throughout the campaign period?**

**Ans.**

***Code Snippet:***

*camp\_data = ad.groupby('month')[['displays', 'clicks']].sum()*

*print(camp\_data)*

*plt.plot(camp\_data.index, camp\_data['displays'], label='Displays')*

*plt.plot(camp\_data.index, camp\_data['clicks'], label='Clicks')*

*plt.xlabel('Month')*

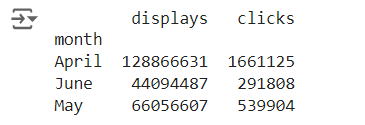
*plt.ylabel('Count')*

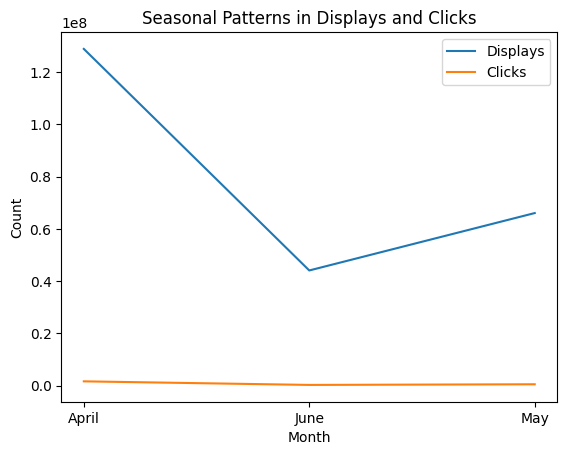
*plt.title('Seasonal Patterns in Displays and Clicks')*

*plt.legend()*

*plt.show()*

**Output:**





1. **Is there a correlation between user engagement levels and the revenue generated?**

**Ans.**

***Code Snippet:***

*corr\_rev\_user = ad.groupby('user\_engagement')['revenue'].sum()*

*print(corr\_rev\_user)*

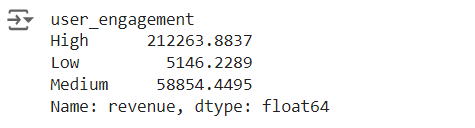
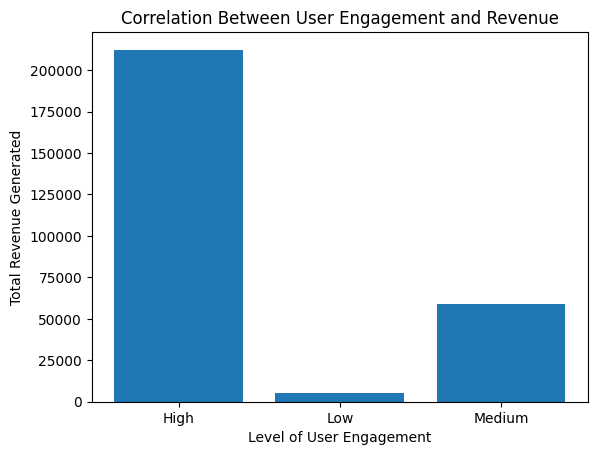
*plt.bar(corr\_rev\_user.index, corr\_rev\_user.values)*

*plt.xlabel('Level of User Engagement')*

*plt.ylabel('Total Revenue Generated')*

*plt.title('Correlation Between User Engagement and Revenue')*

*plt.show()*

****

1. **Are there any outliers in terms of cost, clicks, or revenue that warrant further investigation?**

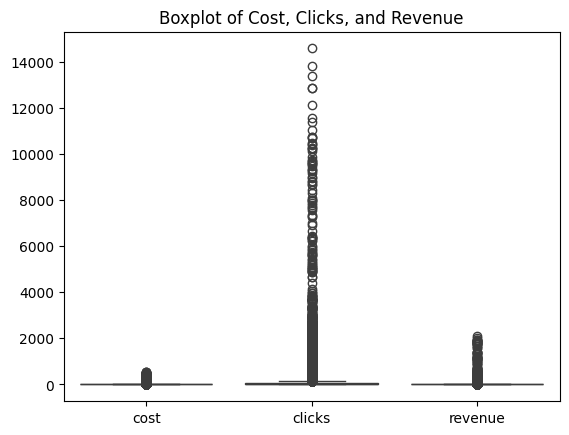
**Ans.**

***Code Snippet:***

*sns.boxplot(data=ad[['cost', 'clicks', 'revenue']])*

*plt.title('Boxplot of Cost, Clicks, and Revenue')*

*plt.show()*

****

1. **How does the effectiveness of campaigns vary based on the size of the ad and placement type?**

**Ans.**

***Code Snippet:***

*filter\_ad = ad[ad['clicks'] != 0]*

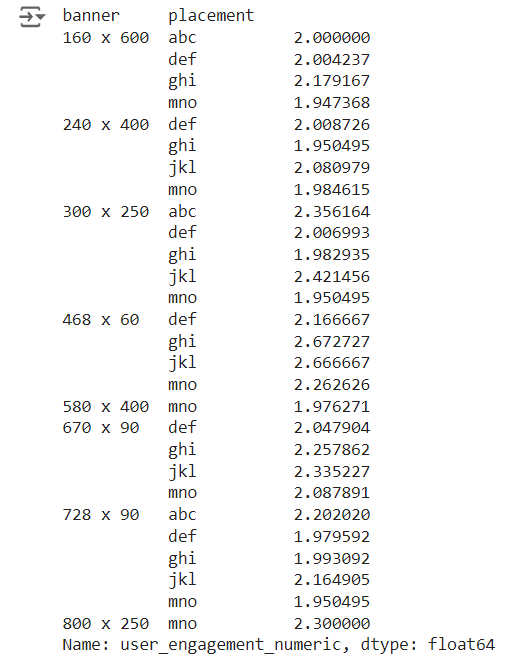
*post\_click\_conversion\_rate = filter\_ad['post\_click\_conversions'] / filter\_ad['clicks']*

*user\_engagement\_mapping = {'High': 3, 'Medium': 2, 'Low': 1}*

*ad['user\_engagement\_numeric'] = ad['user\_engagement'].map(user\_engagement\_mapping)*

*effectiveness = filter\_ad.groupby(['banner', 'placement'])['user\_engagement\_numeric'].mean()*

*print(effectiveness)*



1. **Are there any specific campaigns or banner sizes that consistently outperform others in terms of ROI?  
   Ans.**

***Code Snippet:***

*filter\_ad = ad[ad['cost'] != 0]*

*filter\_ad['ROI'] = (filter\_ad['revenue'] - filter\_ad['cost']) / filter\_ad['cost']*

*roi\_by\_campaign\_banner = filter\_ad.groupby(['campaign\_number', 'banner'])['ROI'].mean().reset\_index()*

*top\_campaigns = roi\_by\_campaign\_banner.sort\_values('ROI', ascending=False).head(10)*

*print("Top Performing Campaigns and Banner Sizes by ROI:")*

*print(top\_campaigns)*

*plt.figure(figsize=(12, 6))*

*sns.barplot(x='campaign\_number', y='ROI', hue='banner', data=roi\_by\_campaign\_banner)*

*plt.title('ROI by Campaign and Banner Size')*

*plt.xlabel('Campaign Number')*

*plt.ylabel('Return on Investment (ROI)')*

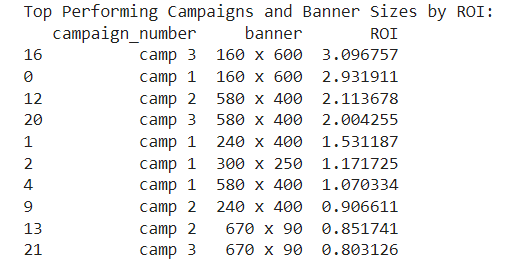
*plt.xticks(rotation=45)*

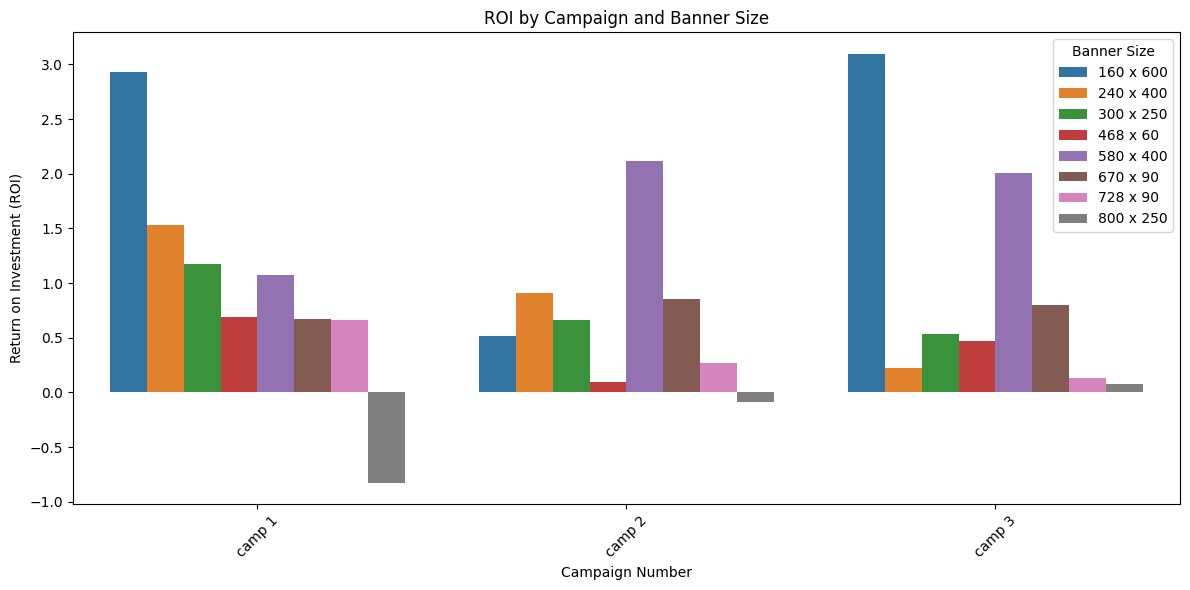
*plt.legend(title='Banner Size')*

*plt.tight\_layout()*

*plt.show()*

**Output:**



****

1. **What is the distribution of post-click conversions across different placement types?**

**Ans.**

***Code Snippet:***

*conv\_placement = ad.groupby('placement')['post\_click\_conversions'].sum()*

*print(conv\_placement)*

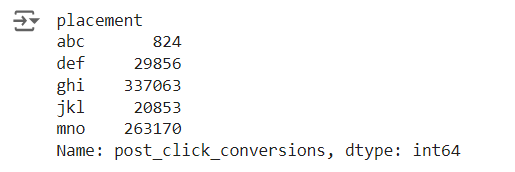
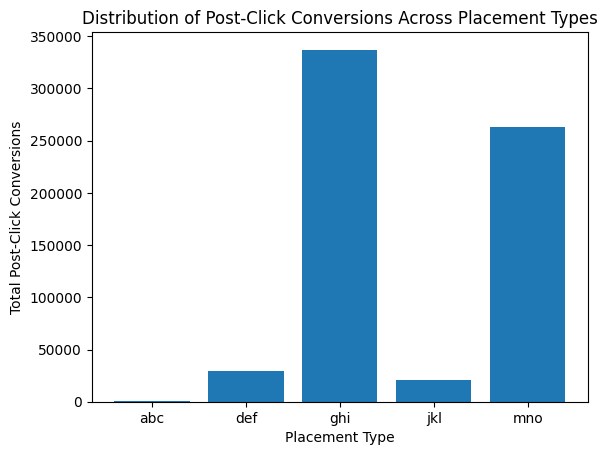
*plt.bar(conv\_placement.index, conv\_placement.values)*

*plt.xlabel('Placement Type')*

*plt.ylabel('Total Post-Click Conversions')*

*plt.title('Distribution of Post-Click Conversions Across Placement Types')*

*plt.show()*

****

1. **Are there any noticeable differences in user engagement levels between weekdays and weekends?**

**Ans.**

***Code Snippet:***

*month\_map = {'April':'04', 'May':'05', 'June' : '06'}*

*ad['month'] = ad['month'].apply(lambda x: month\_map[x])*

*ad['date'] = ad['month'].astype(str) + '/' + ad['day'].astype(str) + '/2020'*

*ad['date'] = pd.to\_datetime(ad['date'], format='%m/%d/%Y')*

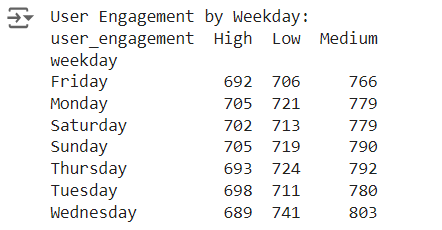
*ad['weekday'] = ad['date'].dt.day\_name()*

*engagement\_weekday = ad.groupby('weekday')['user\_engagement'].value\_counts().unstack().fillna(0)*

*print("User Engagement by Weekday:")*

*print(engagement\_weekday)*

**Output:**



1. **How does the cost per click (CPC) vary across different campaigns and banner sizes?**

**Ans.**

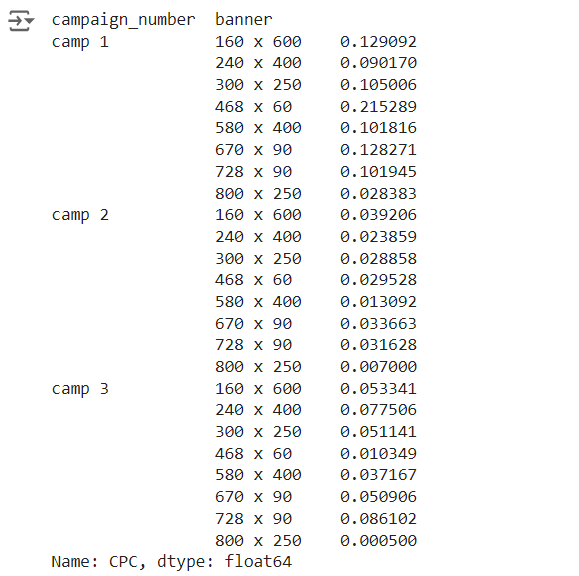
***Code Snippet:***

*filter\_ad = ad[ad['clicks'] != 0]*

*filter\_ad['CPC'] = filter\_ad['cost'] / filter\_ad['clicks']*

*cpc\_range = filter\_ad.groupby(['campaign\_number', 'banner'])['CPC'].mean()*

*print(cpc\_range)*



1. **Are there any campaigns or placements that are particularly cost-effective in terms of generating post-click conversions?**

**Ans.**

***Code Snippet:***

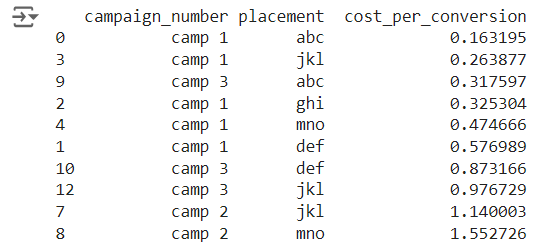
*filter\_ad = ad[ad['post\_click\_conversions'] != 0]*

*filter\_ad['cost\_per\_conversion'] = filter\_ad['cost'] / filter\_ad['post\_click\_conversions']*

*cost\_effectiveness = filter\_ad.groupby(['campaign\_number', 'placement'])['cost\_per\_conversion'].mean().reset\_index()*

*cost\_effectiveness = cost\_effectiveness.sort\_values('cost\_per\_conversion')*

*print(cost\_effectiveness.head(10))*



1. **Can we identify any trends or patterns in post-click conversion rates based on the day of the week?**

**Ans.**

***Code Snippet:***

*month\_map = {'April':'04', 'May':'05', 'June' : '06'}*

*ad['date'] = ad['month'].astype(str) + '/' + ad['day'].astype(str) + '/2020'*

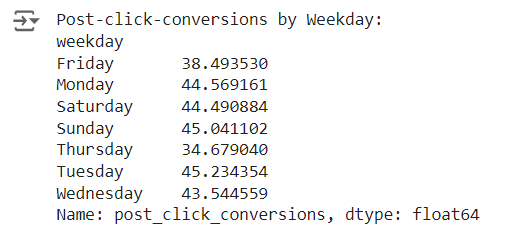
*ad['date'] = pd.to\_datetime(ad['date'], format='%m/%d/%Y')*

*ad['weekday'] = ad['date'].dt.day\_name()*

*conv\_weekday = ad.groupby('weekday')['post\_click\_conversions'].mean().fillna(0)*

*print("Post-click-conversions by Weekday:")*

*print(conv\_weekday)*



1. **How does the effectiveness of campaigns vary throughout different user engagement types in terms of post-click conversions?**

**Ans.**

***Code Snippet:***

*ad.fillna({'campaign\_number': 0, 'user\_engagement': 'Unknown', 'post\_click\_conversions': 0}, inplace=True)*

*campaign\_effectiveness\_by\_engagement = ad.groupby(['campaign\_number', 'user\_engagement'])['post\_click\_conversions'].mean().reset\_index()*

*pivot\_table = campaign\_effectiveness\_by\_engagement.pivot\_table(index='campaign\_number',*

*columns='user\_engagement', values='post\_click\_conversions', fill\_value=0*

*print(pivot\_table)*

