

Департамент образования города Москвы
Государственное автономное образовательное учреждение
высшего образования города Москвы
«Московский городской педагогический университет»

Институт цифрового образования
Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

«Проектный практикум по разработке ETL-решений »

Лабораторная работа № 3.1

«Интеграция данных из нескольких источников. Обработка и согласование
данных из разных источников»

Выполнила:

Студентка группы АДЭУ-211

Кравцова Алёна Евгеньевна

Руководитель:

Босенко Т.М

Москва

2025

Цель работы: получить практические навыки интеграции, обработки и согласования данных из различных источников с использованием Python и его библиотек.

Задачи:

- Изучить методы чтения данных из разных источников;
- Освоить техники обработки и очистки данных;
- Научиться согласовывать данные из разных источников;
- Реализовать сохранение обработанных данных.

Задание 1. Построить верхнеуровневую архитектуру.

В качестве источников данных выступают: PostgreSQL, .csv, .xlsx. Данные обрабатываются посредством python в jupyter notebook.

На рисунке 1 представлена верхнеуровневая архитектура задания.

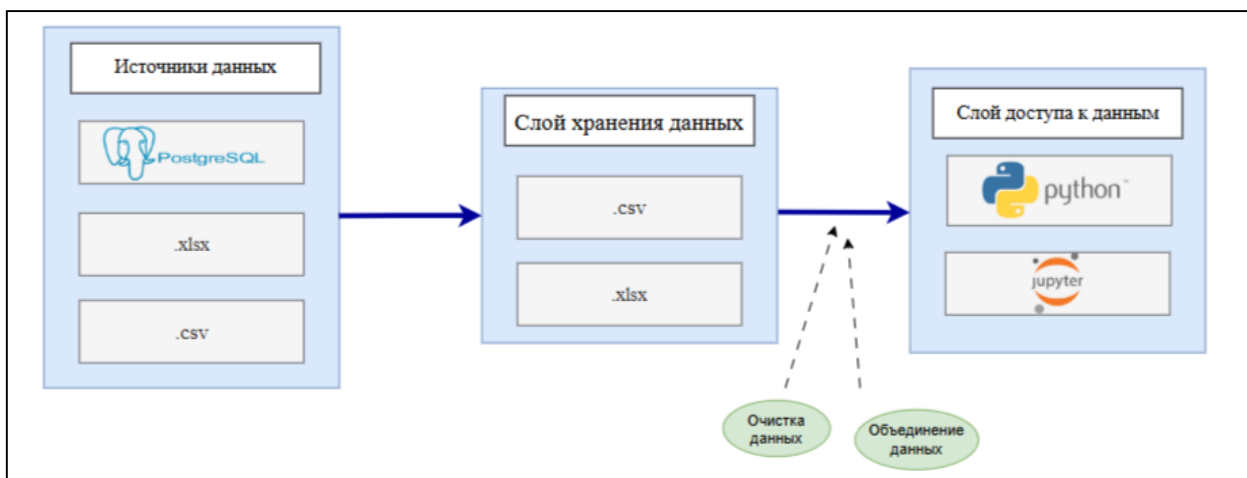


Рис. 1 –Архитектура

Задание 2. Написать код на Python для: - Чтения данных из всех источников. - Очистки и преобразования данных. - Объединения данных. - Анализа результатов.

Предварительно необходимо подготовить данные для каждого источника. Описание данных для каждого источника представлено в таблице 1.

Таблица 1 – Описание данных

Источник	Назначение данных	Поле	Тип	Описание
PostgreSQL	Данные о рекламных кампаниях	campaign_id	INTEGER	Уникальный идентификатор рекламной кампании.
		campaign_name	VARCHAR.	Название рекламной кампании.
		start_date	DATE	Дата начала кампании.
		end_date	DATE	Дата окончания кампании.
		Impressions	INTEGER	Количество показов рекламных материалов.
		clicks	INTEGER	Количество кликов на рекламный материал.
		conversions	INTEGER	Количество конверсий (например, покупок или регистраций).
		revenue	FLOAT	Доход, полученный от рекламной кампании.
		status	VARCHAR	Статус кампании (активная, завершена, приостановлена).
.csv	Данные из социальных сетей	campaign_id	INTEGER	Уникальный идентификатор кампании, в рамках которой пост
		post_id	INTEGER	Уникальный идентификатор

				поста в социальной сети.
		post_date	DATE	Дата публикации поста.
		post_type	VARCHAR	Тип контента (изображение, видео, текст).
		Post_resource	VARCHAR	Канал (Instagram, Twitter, Telegram, VK, Facebook)
		likes	INTEGER	Количество лайков на пост.
		comments	INTEGER	Количество комментариев под постом.
		shares		Количество репостов.
		Clicks	INTEGER	Количество кликов на ссылку в посте.
		Reach	INTEGER	Количество людей, увидевших пост.
.xlsx	Данные о бюджете на рекламу	campaign_id	INTEGER	Уникальный идентификатор рекламной кампании.
		date	DATE	Дата записи о бюджете.
		Budget	FLOAT	Затраты на рекламу для конкретной даты.
		source	VARCHAR	Источник выделенного бюджета (социальные сети, ТВ реклама, контекстная реклама).

На рисунках 2, 3 и 4 представлены необходимые данные по рекламным кампаниям.

campaign_id	campaign_name	start_date	end_date	impressions	clicks	conversions	revenue	status
498	Chan and Sons	2025-03-17	2025-01-13	876 782	17 246	141	16 935	Active
810	Christensen-Hodges	2025-02-13	2025-02-28	194 445	11 453	665	2 078	Comp
94	Perez, Smith and Douglas	2025-02-18	2025-01-01	417 556	6 567	281	4 949	Active
471	Daniels Ltd	2025-01-27	2025-02-07	770 040	26 261	642	6 995	Pause
194	Bailey, Cordova and Washin	2025-01-25	2025-03-11	762 357	9 143	737	17 742	Active
667	Abbott, Thompson and Hay	2025-01-29	2025-02-12	151 916	7 856	534	14 324	Comp
268	Lynch-Moore	2025-01-16	2025-02-22	274 404	25 409	658	9 413	Comp
58	Fuller, James and Reid	2025-01-05	2025-01-19	933 541	6 797	795	12 793	Pause
348	Hall-Krause	2025-03-03	2025-01-15	268 015	27 075	411	2 083	Comp
75	Johnson, Tanner and Cunni	2025-03-28	2025-03-18	549 107	6 236	126	4 089	Active
181	Johnson LLC	2025-03-30	2025-01-31	349 484	18 409	188	2 594	Pause
932	Gardner Inc	2025-02-01	2025-03-04	742 534	21 337	529	8 380	Comp
997	French, Ball and Christenser	2025-03-17	2025-04-03	667 391	8 434	569	19 100	Comp
920	Kennedy Inc	2025-01-19	2025-03-07	945 602	21 966	129	16 963	Active
706	Nelson, Miles and Luna	2025-03-16	2025-04-01	457 444	8 710	465	17 385	Comp
546	Anderson Group	2025-01-28	2025-01-17	119 878	10 688	618	15 274	Comp
755	Zavala, Maxwell and Gould	2025-02-12	2025-04-03	593 115	13 944	458	4 896	Pause
199	Padilla Inc	2025-03-27	2025-03-20	659 270	25 220	940	6 921	Active
571	Russo-James	2025-01-19	2025-01-19	476 142	10 452	969	8 589	Pause
857	Ross, Munoz and Moore	2025-02-04	2025-02-21	267 701	13 377	233	2 137	Active
198	Bryan, Robinson and Choi	2025-02-08	2025-03-18	646 697	19 052	290	10 796	Comp
473	Melendez, Knox and Anders	2025-02-17	2025-02-15	983 002	12 995	511	8 075	Pause
814	Gregory PLC	2025-01-19	2025-01-25	437 287	24 851	376	8 255	Comp
732	Smith, Sanders and Hood	2025-03-08	2025-01-31	446 236	28 853	946	8 525	Pause
105	Wood and Sons	2025-03-18	2025-01-02	367 940	22 954	981	4 229	Active
789	Martin, Gill and Valencia	2025-01-24	2025-04-02	806 864	17 602	848	18 332	Active
223	Rivers-Murphy	2025-03-24	2025-03-27	823 641	16 289	633	10 197	Comp

Рис. 2 – Данные по рекламным кампаниям

campaign_id	post_id	post_date	post_type	post_resource	likes	comments	shares	clicks	reach
498	655	2025-04-03	Text	Instagram	1520	495	110	211	81721
810	928	2025-01-28	Text	Twitter	2900	262	149	357	28815
94	928	2025-03-19	Video	Telegram	3912	489	143	187	14916
471	394	2025-04-02	Image	Twitter	4942	197	122	477	15119
194	386	2025-02-20	Text	Facebook	3403	145	194	405	59541
667	961	2025-03-29	Image	Twitter	2214	173	62	378	84877
268	546	2025-01-26	Text	Telegram	2295	270	61	296	47792
58	68	2025-02-07	Image	Facebook	2324	109	173	160	62535
348	837	2025-02-17	Text	Twitter	1533	115	146	372	32188
75	302	2025-02-19	Text	Twitter	1087	307	159	156	53806
181	538	2025-03-16	Text	Instagram	2248	277	174	223	12786
932	609	2025-02-25	Text	VK	4718	354	179	326	26582
997	599	2025-02-17	Image	Facebook	4008	219	101	221	65522
920	184	2025-01-29	Text	VK	3129	488	69	112	28543
706	100	2025-01-27	Text	VK	2223	176	113	224	75338
546	892	2025-02-07	Image	Telegram	3395	193	125	448	46682
755	260	2025-03-15	Image	Telegram	1952	196	126	53	67418
199	752	2025-03-15	Video	Twitter	1279	385	153	208	64779
571	946	2025-02-04	Video	Telegram	2439	222	137	454	51910
857	87	2025-03-26	Image	Instagram	4269	471	110	120	94487
198	509	2025-02-25	Text	Twitter	2034	197	136	353	61298
473	317	2025-03-09	Image	Facebook	1128	383	71	99	53938
814	46	2025-03-11	Image	Instagram	1810	399	133	465	36712
732	348	2025-02-21	Video	Facebook	2710	126	102	112	88379
105	335	2025-02-23	Video	Facebook	2911	405	102	309	44903

Рис. 3 – Данные социальных сетей в формате .csv

	A	B	C	D	E
1	campaign_id	date	budget	source	
2	498	16.01.25	8895	Display Ads	
3	810	20.01.25	5595	TV	
4	94	26.02.25	8293	Search Ads	
5	471	31.03.25	9194	Search Ads	
6	194	23.01.25	3573	Search Ads	
7	667	14.03.25	6366	TV	
8	268	14.03.25	9601	Social Media	
9	58	25.02.25	9893	Display Ads	
10	348	01.01.25	4097	Social Media	
11	75	26.03.25	2955	Search Ads	
12	181	01.01.25	4306	TV	
13	932	18.02.25	8537	Display Ads	
14	997	06.02.25	3678	Social Media	
15	920	16.03.25	3426	Display Ads	
16	706	19.01.25	6460	TV	
17	546	27.01.25	5129	Search Ads	
18	755	06.02.25	6497	Social Media	
19	199	08.01.25	2274	Social Media	
20	571	21.01.25	8920	TV	

Рис. 4 – Данные о бюджетах на рекламу в формате .xlsx

После подготовки данных приступим к их загрузке в jupyter notebook. Создадим подключение к БД, после успешного подключения сделаем запрос к таблице и сохраним данные. Далее соединение необходимо закрыть (Рис. 5).

```
import psycopg2
import pandas as pd

DB_NAME = "postgres"
DB_USER = "postgres"
DB_PASSWORD = "1"
DB_HOST = "localhost"
DB_PORT = "5432"

# Подключение к базе
conn = psycopg2.connect(
    dbname=DB_NAME,
    user=DB_USER,
    password=DB_PASSWORD,
    host=DB_HOST,
    port=DB_PORT
)

cursor = conn.cursor()

query = "SELECT * FROM campaigns;"
campaigns = pd.read_sql(query, conn)

campaigns.head()
```

C:\Users\alnys\AppData\Local\Temp\ipykernel_21288\422664654.py:2: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.

```
campaigns = pd.read_sql(query, conn)
```

	campaign_id	campaign_name	start_date	end_date	impressions	clicks	conversions	revenue	status
0	498	Chan and Sons	2025-03-17	2025-01-13	876782	17246	141	16935.0	Active
1	810	Christensen-Hodges	2025-02-13	2025-02-28	194445	11453	665	2078.0	Completed
2	94	Perez, Smith and Douglas	2025-02-18	2025-01-01	417556	6567	281	4949.0	Active
3	471	Daniels Ltd	2025-01-27	2025-02-07	770040	26261	642	6995.0	Paused
4	194	Bailey, Cordova and Washington	2025-01-25	2025-03-11	762357	9143	737	17742.0	Active

```
cursor.close()
conn.close()
```

Рис. 5 – Получение данных из PostgreSQL

Далее загрузим данные о социальных сетях (Рис. 6).

```
#загрузка cvs
df_social_media = pd.read_csv("social-media.csv")
df_social_media.head()
```

	campaign_id	post_id	post_date	post_type	post_resource	likes	comments	shares	clicks	reach
0	498	655	2025-04-03	Text	Instagram	1520	495	110	211	81721
1	810	928	2025-01-28	Text	Twitter	2900	262	149	357	28815
2	94	928	2025-03-19	Video	Telegram	3912	489	143	187	14916
3	471	394	2025-04-02	Image	Twitter	4942	197	122	477	15119
4	194	386	2025-02-20	Text	Facebook	3403	145	194	405	59541

Рис. 6 – Загрузка данных из .csv

И загрузим данные из.xlsx (Рис. 7).

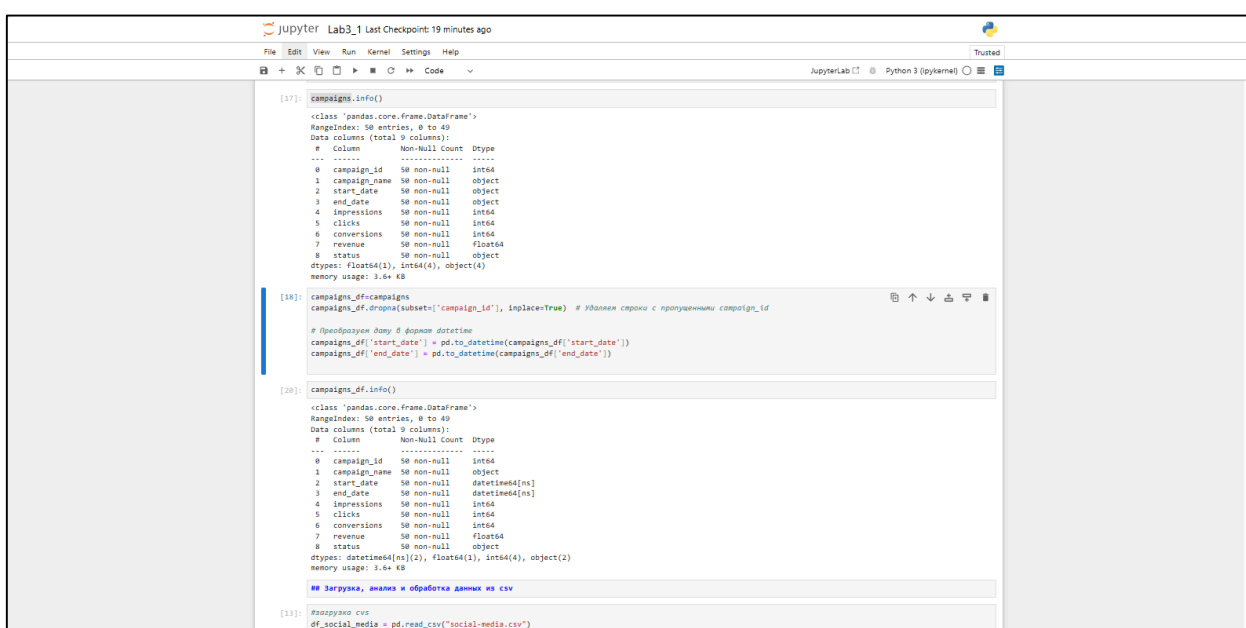
```
[16]: # загрузка данных из Excel
df_budget = pd.read_excel("budget.xlsx")
df_budget.head()

[16]:
```

	campaign_id	date	budget	source
0	498	2025-01-16	8895	Display Ads
1	810	2025-01-20	5595	TV
2	94	2025-02-26	8293	Search Ads
3	471	2025-03-31	9194	Search Ads
4	194	2025-01-23	3573	Search Ads

Рис. 7 – Загрузка данных из .xlsx

Далее необходимо провести анализ и обработку данных. Так для данных из PostgreSQL необходимо установить формат даты в соответствующих столбцах и удалить пропуски (Рис. 8).



```
[17]: campaigns.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  --
 0   campaign_id   50 non-null     int64  
 1   campaign_name 50 non-null     object  
 2   start_date    50 non-null     object  
 3   end_date      50 non-null     object  
 4   impressions   50 non-null     int64  
 5   clicks        50 non-null     int64  
 6   conversions   50 non-null     int64  
 7   revenue       50 non-null     float64 
 8   status        50 non-null     object  
dtypes: float64(1), int64(4), object(4)
memory usage: 3.6+ KB

[18]: campaigns_df=campaigns
campaigns_df.dropna(subset=['campaign_id'], inplace=True) # Удален строки с пропущенными campaign_id

# Преобразуем даты в формат datetime
campaigns_df['start_date'] = pd.to_datetime(campaigns_df['start_date'])
campaigns_df['end_date'] = pd.to_datetime(campaigns_df['end_date'])

[19]: campaigns_df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  --
 0   campaign_id   50 non-null     int64  
 1   campaign_name 50 non-null     object  
 2   start_date    50 non-null     datetime64[ns]
 3   end_date      50 non-null     datetime64[ns]
 4   impressions   50 non-null     int64  
 5   clicks        50 non-null     int64  
 6   conversions   50 non-null     int64  
 7   revenue       50 non-null     float64 
 8   status        50 non-null     object  
dtypes: datetime64[ns](2), float64(1), int64(4), object(2)
memory usage: 3.6+ KB

## Загрузка, анализ и обработка данных из csv

[11]: #загрузка csv
df_social_media = pd.read_csv("social-media.csv")
```

Рис. 8 – Настройка формата даты

Аналогичную процедуру необходимо выполнить с данными из csv (Рис. 9) и excel (Рис. 10).


```
[22]: social_media.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 10 columns):
#   Column      Non-Null Count  Dtype
---  -
0   campaign_id  50 non-null    int64
1   post_id      50 non-null    int64
2   post_date    50 non-null    object
3   post_type    50 non-null    object
4   post_resource 50 non-null    object
5   likes        50 non-null    int64
6   comments     50 non-null    int64
7   shares       50 non-null    int64
8   clicks       50 non-null    int64
9   reach        50 non-null    int64
dtypes: int64(7), object(3)
memory usage: 4.0+ KB

[24]: df_social_media=social_media

df_social_media.dropna(subset=['post_id'], inplace=True)
df_social_media['post_date'] = pd.to_datetime(df_social_media['post_date'])

[25]: df_social_media.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 10 columns):
#   Column      Non-Null Count  Dtype
---  -
0   campaign_id  50 non-null    int64
1   post_id      50 non-null    int64
2   post_date    50 non-null    datetime64[ns]
3   post_type    50 non-null    object
4   post_resource 50 non-null    object
5   likes        50 non-null    int64
6   comments     50 non-null    int64
7   shares       50 non-null    int64
8   clicks       50 non-null    int64
9   reach        50 non-null    int64
dtypes: datetime64[ns](1), int64(7), object(2)
memory usage: 4.0+ KB
```

Рис. 9 – Работа с данными из .csv

```
[27]: budget.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   campaign_id  50 non-null    int64
1   date         50 non-null    datetime64[ns]
2   budget       50 non-null    int64
3   source       50 non-null    object
dtypes: datetime64[ns](1), int64(2), object(1)
memory usage: 1.7+ KB

[28]: df_budget=budget

df_budget.dropna(subset=['budget'], inplace=True)

[29]: budget.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   campaign_id  50 non-null    int64
1   date         50 non-null    datetime64[ns]
2   budget       50 non-null    int64
3   source       50 non-null    object
dtypes: datetime64[ns](1), int64(2), object(1)
memory usage: 1.7+ KB
```

Рис. 10 – Работа с данными из .xlsx

Следующий шаг – объединение данных по общему столбцу campaign_id (Рис. 11).

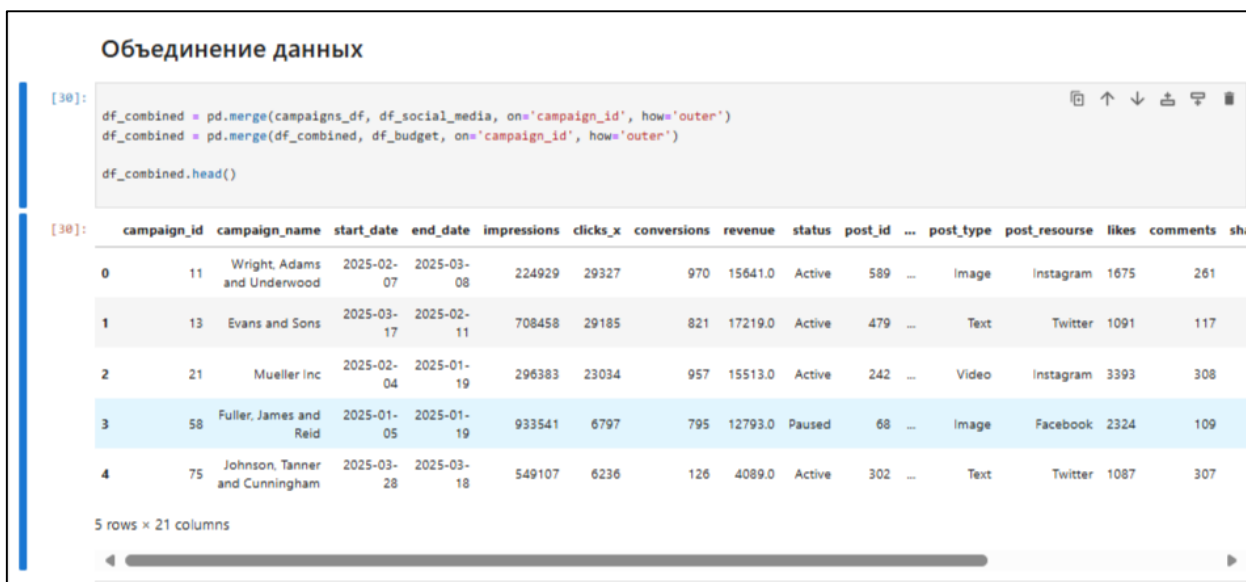


Рис. 11 – Объединение данных

Проведем анализ получившегося набора данных. Для анализа можно рассчитать метрики эффективности маркетинга (Рис. 12):

- Конверсия (Conversion Rate): $(\text{Количество конверсий} / \text{Количество показов}) * 100$
- CPC (Cost per Click): $\text{Бюджет} / \text{Количество кликов}$
- ROI (Return on Investment): $(\text{Доход от рекламы} - \text{Бюджет}) / \text{Бюджет}$

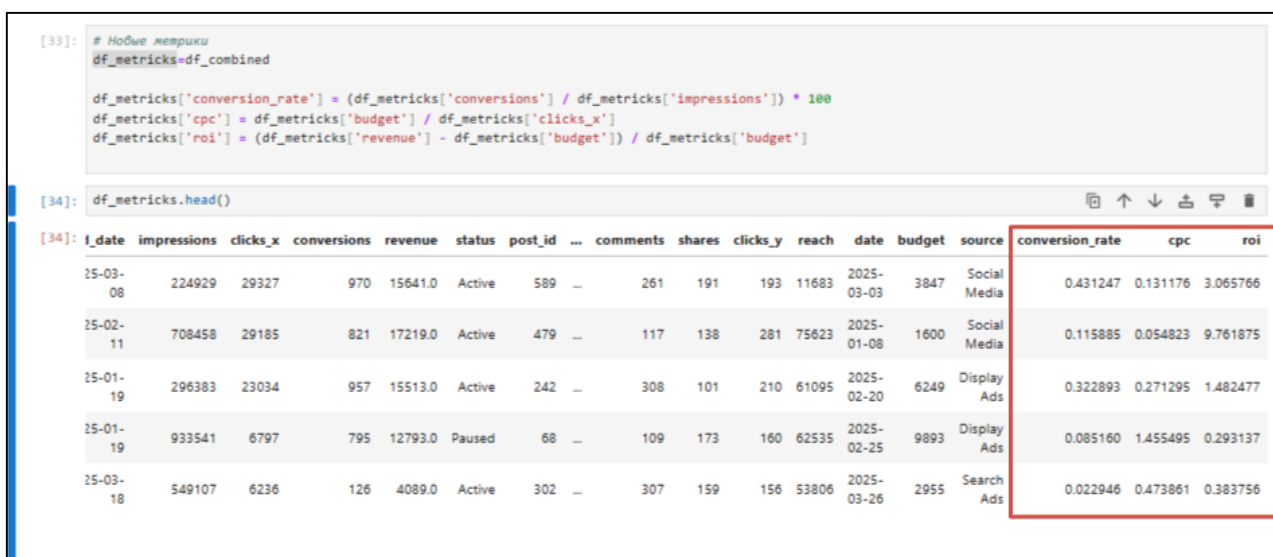


Рис. 12 – Расчёт метрик

Задание 3. Создать отчет с визуализацией.

Построим визуализацию с помощью библиотеки matplotlib и seaborn.

Распределение бюджета между рекламными кампаниями (Рис. 13) показывает, что бюджет распределяется неравномерно.

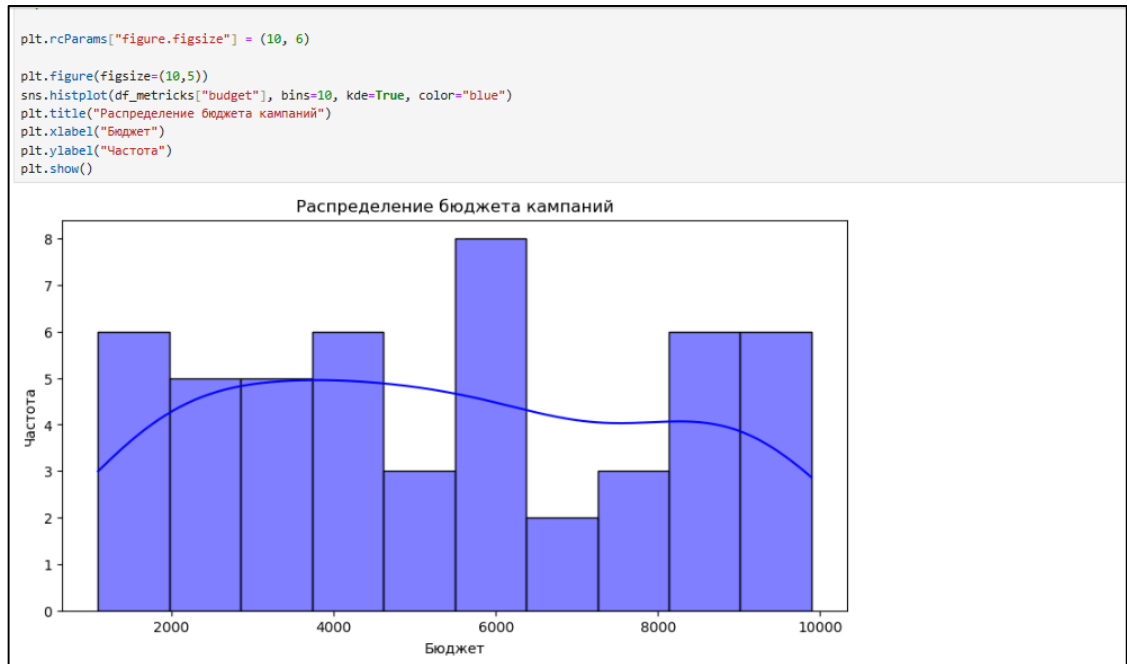


Рис. 13 – Распределение бюджета на рекламные кампании

Зависимость кликов от числа показов (Рис. 14). Возможно для каких-то каналов есть четкая зависимость, посмотрим графики по каждому каналу (Рис. 15). Разброс данных слишком большой.

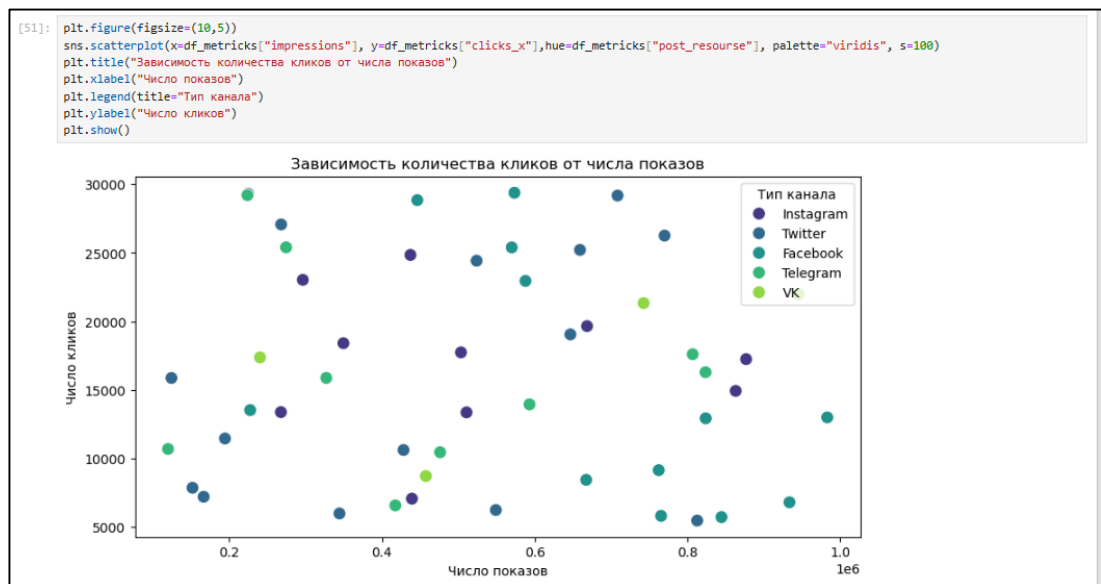


Рис. 14 – Зависимость количества кликов от числа показов

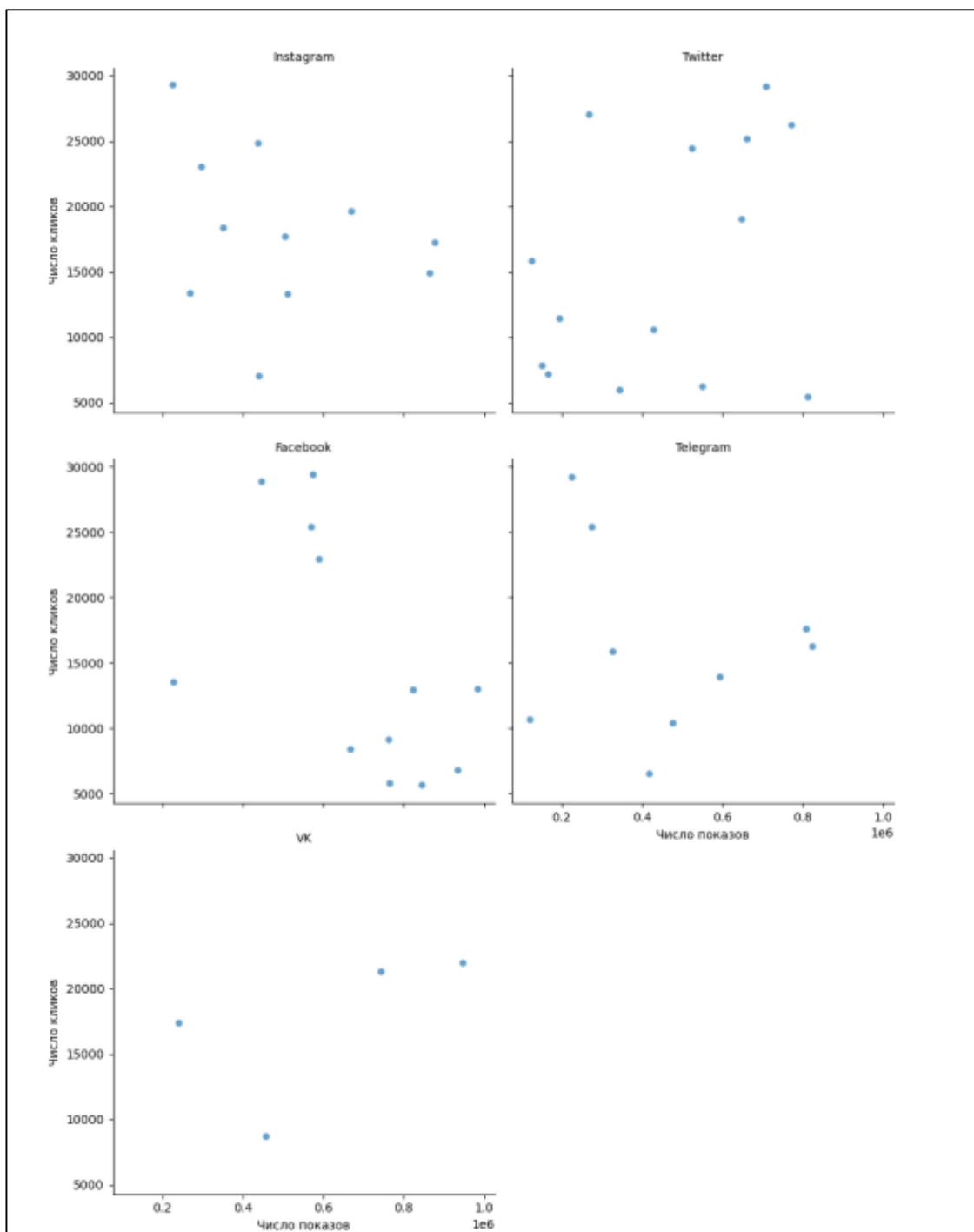


Рис. 15 – Детализированные сведения по каналам

Посмотрим конверсии в зависимости от бюджета рекламной кампании (Рис. 16). Исходя из полученного бюджета нельзя утверждать, что чем больше бюджет, тем больше конверсия.

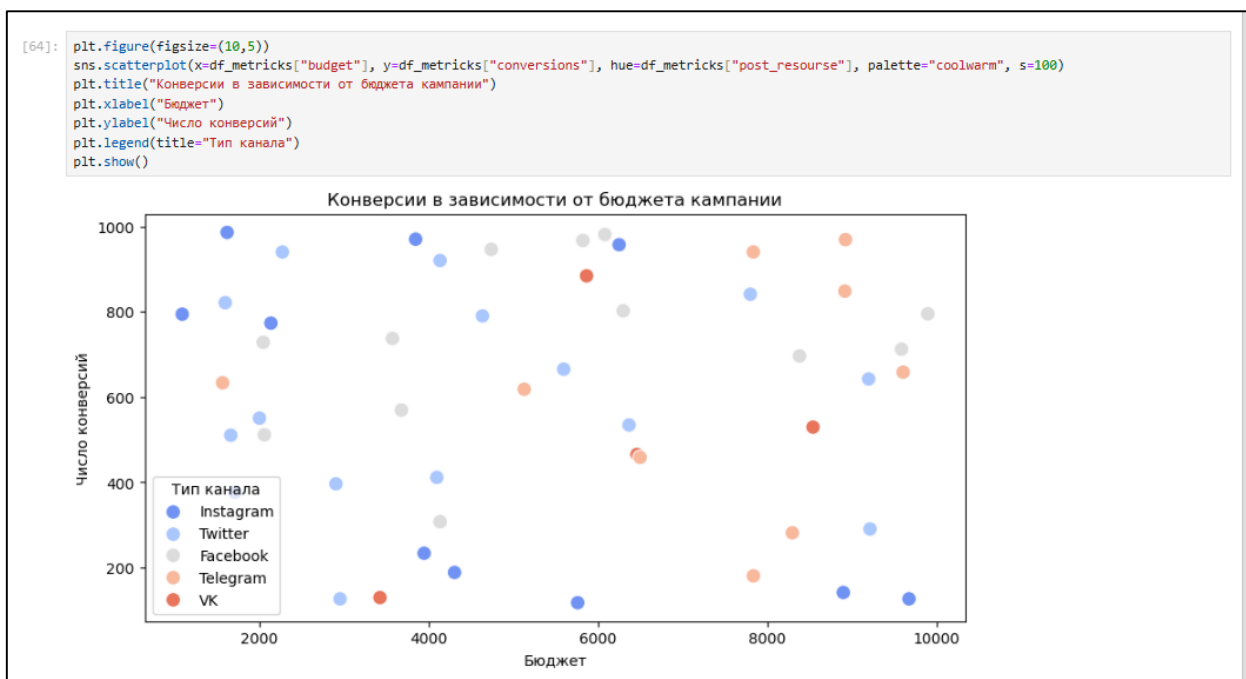


Рис. 16 – Конверсии по бюджетам

Далее проанализируем доходы в зависимости от канала (Рис. 17). Исходя из полученного графика, видно, что медиана Instagram и VK самые большие, то есть эти каналы приносят больше всего прибыли. Однако у этих каналов очень большой размах, то есть как очень удачные кампания, так и провальные. А вот доход по Telegram, Facebook самый стабильный предсказуемый (наименьший размах), но небольшой.

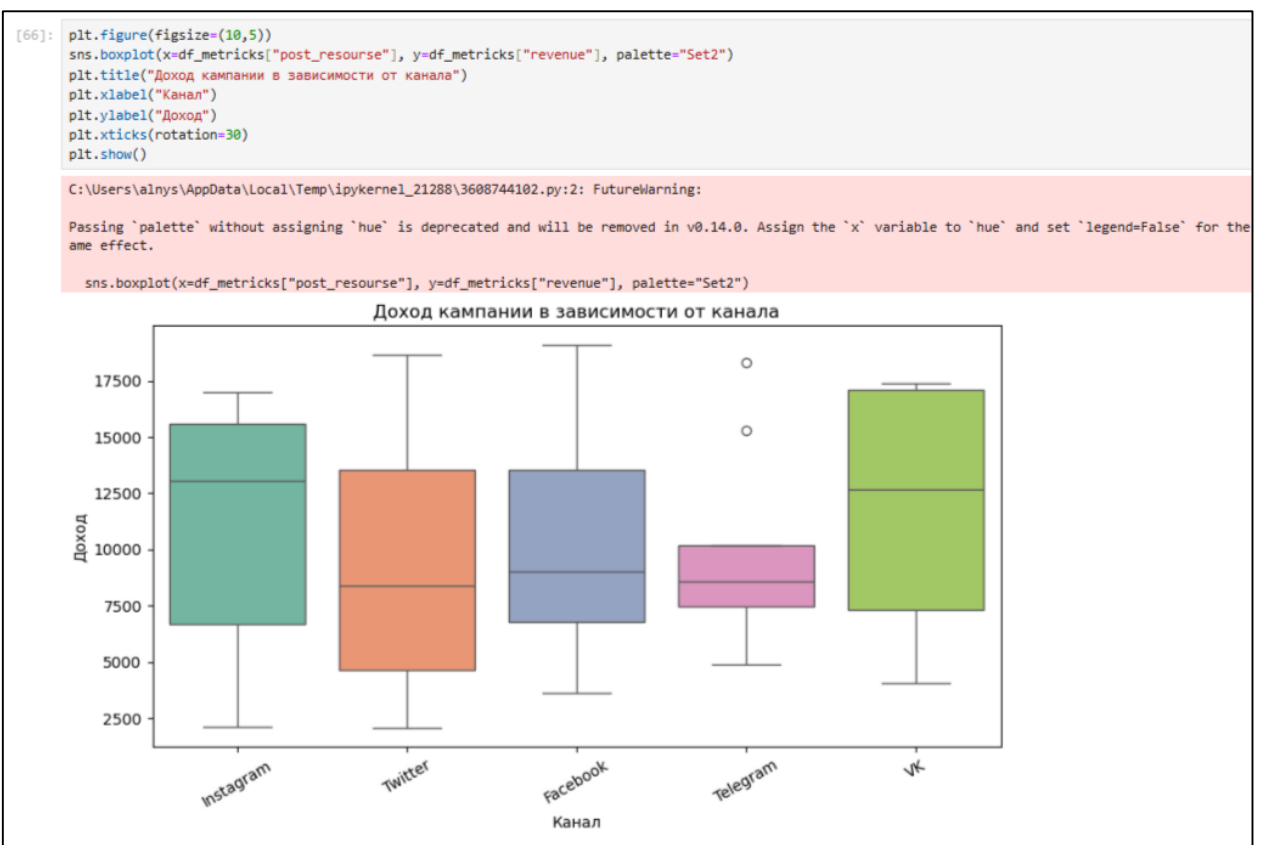
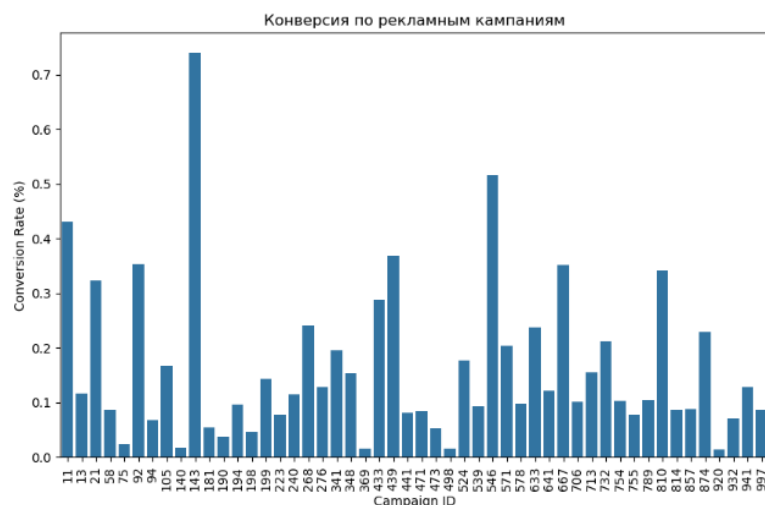


Рис. 17 – Доход по кампаниям в зависимости от канала

Далее просмотрим метрики кампаний (Рис. 18). Некоторые кампании имеют очень высокий коэффициент конверсии, но большинство находятся на низком уровне. Видно, что у некоторых кампаний CPC сильно выше, чем у других. Разные кампании требуют разного бюджета, и некоторые могут быть дорогими, но неэффективными. Большая часть кампаний имеет ROI ниже 2, что указывает на низкую эффективность и необходимость оптимизации.



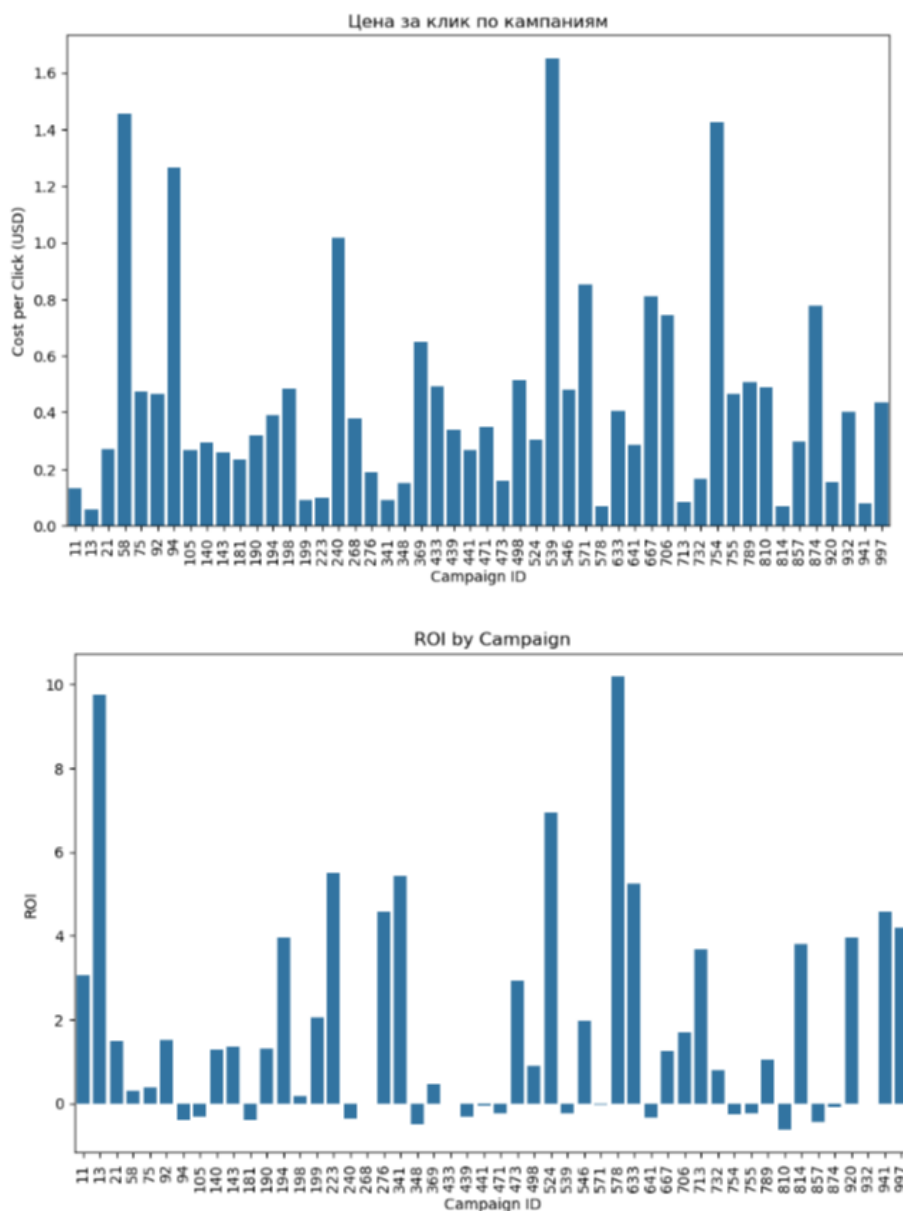


Рис. 18 – Метрики кампаний

Задание 4. Разработать рекомендации по улучшению качества данных.

Важно обрабатывать пропуски, убирать выбросы и дубликаты. Следует приводить данные к единому формату и стандартизировать категории. Также необходимо улучшить анализ данных, а именно рассмотреть выбросы и постараться снизить их количество.

Вариант 6. Интеграция маркетинговых данных: - PostgreSQL база с данными рекламных кампаний. - CSV файлы с данными из социальных сетей. - Excel файл с бюджетами на рекламу. Задача: проанализировать эффективность маркетинговых каналов.

Задания варианта были учтены при выполнении общего задания.

Контрольные вопросы:

1. Какие методы чтения данных предоставляет pandas?

Чтение данных из файлов .csv, .xlsx, .json, .xml, а также за счет подключения к БД.

2. Как обрабатывать пропущенные значения?

Сперва необходимо обнаружить пропущенные значения `df.isnull()`. Удалить строки с пропусками можно командой `df.dropna()`

3. Какие типы объединения данных существуют?

В библиотеке pandas можно использовать merge, он работает, как и join SQL (есть параметры inner, left, right, outer).

`concat()` — быстро склеить фреймы по строкам или столбцам.

4. Как проверить качество объединения данных?

Сравнить количество строк до и после объединения. Проверка на пропущенные значения до и после объединения

5. Какие методы дедупликации данных существуют?

Например, удаление полных дубликатов `df.drop_duplicates(inplace=True)`. Или же нормализация данных для приведения данных к единому виду.

Вывод: таким образом, была выполнена гармонизация данных из различных условий, проведен анализ эффективности маркетинговых кампаний. В результате анализа данных стало ясно, что компании требуется пересмотр финансирования рекламных кампаний и выборов каналов. Поставленные цели и задачи выполнены.