

Документация

1. Описание структуры БД

Основные таблицы

1. Таблица orders

- Ключ: row_id
- Основные атрибуты:
 - Идентификаторы: order_id, product_id, customer_id
 - Дата: order_date,
 - Доставка: ship_date, ship_mode
 - Клиент: customer_name, segment
 - География: country, city, state, postal_code, region
 - Товар: category, subcategory, product_name
 - Финансы: sales, quantity, discount, profit

2. Таблица returns

- Ключ: order_id.
- Содержит флаг возврата (returned = 'Yes') и номер заказа.
- Атрибуты: order_id, returned.

3. Таблица people

- Ключ: person.
- Содержит 4 менеджеров, закрепленных за регионами.
- Атрибуты: person, region.

Связи между таблицами

- orders → returns: связь по order_id (один заказ может быть возвращён).
- orders → people: связь по region (каждый регион закреплён за конкретным сотрудником).

2. Пояснения к реализации заданий

-- Задание 1: [Создать представление по возвратам]

CREATE OR REPLACE VIEW public.returns_view AS

SELECT DISTINCT -- оставляем только уникальные строки

o.order_id, -- идентификатор заказа

o.order_date, -- дата заказа

o.customer_id, -- идентификатор покупателя

o.customer_name, -- имя покупателя

o.product_id, -- идентификатор товара

o.product_name, -- название товара

o.category, -- категория товара

o.subcategory, -- подкатегория товара

o.sales, -- продажи (доход)

o.quantity, -- количество товара

o.discount, -- скидка

o.profit, -- прибыль

o.region, -- регион

r.returned -- возвраты товара

FROM public.orders o

JOIN public.returns r

ON o.order_id = r.order_id; -- соединение заказов с возвратами по идентификатору заказа

-- Задание 2: [Рассчитать прибыль по регионам]

CREATE TABLE public.profit_by_region AS

SELECT

region, -- регион, по которому считается прибыль

SUM(profit) AS total_profit -- суммарная прибыль для региона

FROM public.orders -- из таблицы заказов

GROUP BY region -- группировка по региону

ORDER BY total_profit DESC; -- сортировка результата: сначала самые прибыльные регионы

-- Задание 3: [Определить количество клиентов по сегментам]

SELECT

segment, -- сегмент клиента (Consumer, Corporate, Home Office)

COUNT(DISTINCT customer_id) AS customer_count -- количество уникальных клиентов в сегменте

FROM public.orders -- из таблицы заказов

GROUP BY segment -- группировка по сегменту

ORDER BY customer_count DESC; -- сортировка по убыванию числа клиентов

3. Результаты проверок

1) Задание 1

Query

Query History

1

SELECT COUNT(*)

2

FROM public.returns_view;

Data Output

Messages

Notifications

+

SQL

count

bigint

1

800

Query Query History

```

1 SELECT order_id, product_id, COUNT(*) AS cnt
2 FROM public.returns_view
3 GROUP BY order_id, product_id
4 HAVING COUNT(*) > 1;

```

Data Output Messages Notifications

order_id	product_id	cnt
character varying (14)	character varying (15)	bigint

Query Query History

```

1 SELECT *
2 FROM public.returns_view
3 WHERE order_id = 'CA-2016-143336';

```

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1 of 1

order_id	order_date	customer_id	customer_name	product_id	product_name	category	subcategory
CA-2016-143336	2016-08-27	ZD-21925	Zuschuss Donatelli	OFF-AR-10003056	Newell 341	Office Supplies	Art
CA-2016-143336	2016-08-27	ZD-21925	Zuschuss Donatelli	OFF-BI-10002215	Wilson Jones Hanging View Binder, White, ...	Office Supplies	Binders
CA-2016-143336	2016-08-27	ZD-21925	Zuschuss Donatelli	TEC-PH-10001949	Cisco SPA 501G IP Phone	Technology	Phones

2) Задание 2

Query Query History

```

1 SELECT COUNT(*)
2 FROM public.profit_by_region;

```

Data Output Messages Notifications

Showing rows:

count
4

Query
Query History

```

1 SELECT
2 (SELECT SUM(total_profit) FROM public.profit_by_region) as total1,
3 (SELECT SUM(profit) FROM public.orders) as total2;

```

Data Output
Messages
Notifications

+

📄

▼

📋

▼

🗑️

📊

⬇️

📈

SQL

	total1 numeric	total2 numeric
1	286397.0216999999887055	286397.0216999999887055

Query
Query History

```

1 SELECT
2 'profit_by_region' as name_table,
3 total_profit as profit
4 FROM public.profit_by_region
5 WHERE region = 'West'
6
7 UNION ALL
8
9 SELECT
10 'orders' as name_table,
11 SUM(profit) as profit
12 FROM public.orders
13 WHERE region = 'West';

```

Data Output
Messages
Notifications

+

📄

▼

📋

▼

🗑️

📊

⬇️

📈

SQL

	name_table text	profit numeric
1	profit_by_region	108418.4488999999957984
2	orders	108418.4488999999957984

3) Задание 3

Query Query History

```
1  SELECT DISTINCT segment
2  FROM public.orders;
```

Data Output Messages Notifications

Icons: List, File, Dropdown, Clipboard, Dropdown, Trash, Database, Download, Refresh, SQL

	segment character varying (11)
1	Consumer
2	Corporate
3	Home Office

```
1  SELECT customer_id, COUNT(DISTINCT segment) AS seg_count
2  FROM public.orders
3  GROUP BY customer_id
4  HAVING COUNT(DISTINCT segment) > 1;
```

Data Output Messages Notifications

Icons: List, File, Dropdown, Clipboard, Dropdown, Trash, Database, Download, Refresh, SQL

customer_id character varying (8)	seg_count bigint
--------------------------------------	---------------------