BeeCycle – SQL Mini Project

PREPARED BY MUH.HANIF FAISALUDDIN

The Topics

1. Introduction
2. The Database
3. Get to Know the Data
4. Exploring the Data

1. Introduction

DATA BACKGROUND

Bee Cycle is a cycle stuff retailer, which is selling bikes, accessories, clothing, and components.

TOOLS

Google Colaboratory for creating the SQL queries

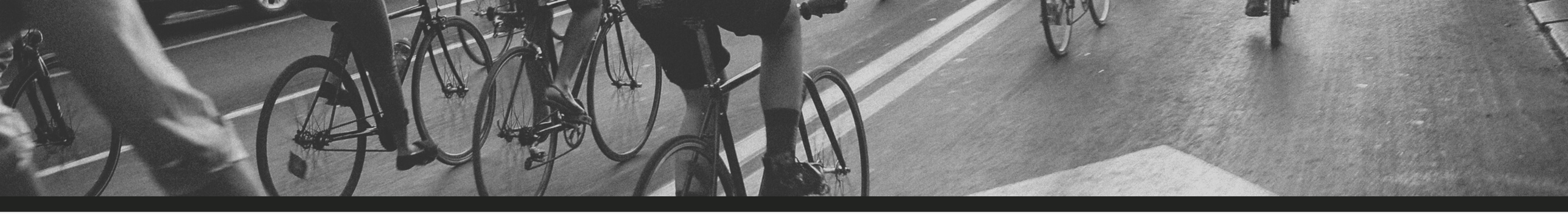
DBeaver for database management

2. The Database

Create a database manager on Heroku Postgres, based on PostgreSQL. Then create all tables on it.

THE TABLES

- dim_geography
- dim_customer
- dim_product
- dim_territory
- fact_sales



Create Table & Import Data

- CREATE TABLE, e.g. dim_product

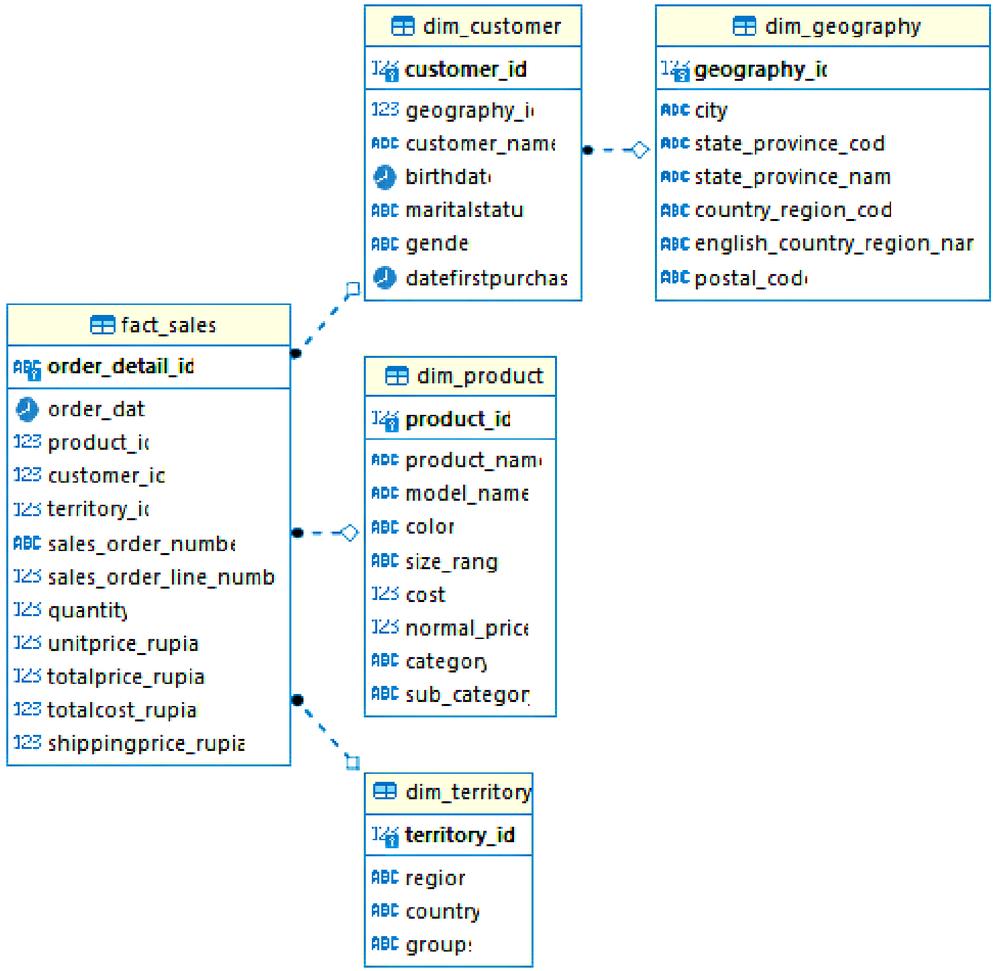
```
CREATE TABLE IF NOT EXISTS public.dim_product (
  product_id int4 primary key,
  product_name varchar(100),
  model_name varchar(100),
  color varchar(30),
  size_range varchar(30),
  "cost" numeric(10),
  normal_price numeric(100),
  sub_category varchar(100),
  category varchar(100)
);
```

IMPORT DATA dim_product via DBeaver.

- Repeat the step in another table.

3. Get to Know the Data

The Entity Relationship (ER) Diagram of the database



The Data

dim_geography

```
# dim_geography
%%sql

SELECT * FROM dim_geography LIMIT 5
```

```
 * postgresql://jnlosugsvinhfn:***@ec2-34-200-205-45.compute-1.amazonaws.com:5432/d2604bq39tv4uh
5 rows affected.
```

geography_id	city	state_province_code	state_province_name	country_region_code	english_country_region_name	postal_code
2	Coffs Harbour	NSW	New South Wales	AU	Australia	2450
3	Darlinghurst	NSW	New South Wales	AU	Australia	2010
4	Goulburn	NSW	New South Wales	AU	Australia	2580
5	Lane Cove	NSW	New South Wales	AU	Australia	1597
6	Lavender Bay	NSW	New South Wales	AU	Australia	2060

Columns:

- geography_id > identifier (Primary Key)
- city
- state_province_code
- state_province_name
- country_region_code
- english_country_region_name
- postal_code

The Data

dim_customer

```
# dim_customer
%%sql

SELECT * FROM dim_customer LIMIT 5
```

```
 * postgresql://jnlosugsvinhfn:***@ec2-34-200-205-45.compute-1.amazonaws.com:5432/d2604bq39tv4uh
5 rows affected.
```

customer_id	geography_id	customer_name	birthdate	maritalstatus	gender	datefirstpurchase
11000	26	Jon Yang	1986-04-08	M	M	2016-07-22
11001	37	Eugene Huang	1985-05-14	S	M	2016-07-18
11002	31	Ruben Torres	1985-08-12	M	M	2016-07-10
11004	19	Elizabeth Johnson	1988-08-08	S	F	2016-07-26
11005	22	Julio Ruiz	1985-08-05	S	M	2016-07-02

Columns:

- customer_id > identifier (Primary Key)
- geography_id > references dim_geography (Foreign Key)
- customer_name
- birthdate
- maritalstatus
- gender
- datefirstpurchase

The Data

dim_product

```
# dim_product
%%sql

SELECT * FROM dim_product LIMIT 5
```

```
 * postgresql://jnlosugsvinhfn:***@ec2-34-200-205-45.compute-1.amazonaws.com:5432/d2604bq39tv4uh
5 rows affected.
```

product_id	product_name	model_name	color	size_range	cost	normal_price	category	sub_category
210	HL Road Frame - Black, 58	HL Road Frame	Black	54-58 CM	11000	11000	Components	Road Frames
211	HL Road Frame - Red, 58	HL Road Frame	Red	54-58 CM	11000	11000	Components	Road Frames
480	Patch Kit/8 Patches	Patch kit	NA	NA	11991	32060	Accessories	Tires and Tubes
529	Road Tire Tube	Road Tire Tube	NA	NA	20892	55860	Accessories	Tires and Tubes
477	Water Bottle - 30 oz.	Water Bottle	NA	NA	26128	69860	Accessories	Bottles and Cages

Columns:

- product_id> identifier (Primary Key)
- product_name
- model_name
- color
- size_range
- cost
- normal_price
- category
- sub_category

The Data

dim_territory

```
# dim_territory
%%sql

SELECT * FROM dim_territory LIMIT 5
```

```
 * postgresql://jnlosugsvinhfn:***@ec2-34-200-205-45.compute-1.amazonaws.com:5432/d2604bq39tv4uh
5 rows affected.
```

territory_id	region	country	groups
1	Northwest	United States	North America
2	Northeast	United States	North America
3	Central	United States	North America
4	Southwest	United States	North America
5	Southeast	United States	North America

Columns:

- territory_id> identifier (Primary Key)
- region
- country
- groups

The Data

fact_sales

```
# fact_sales
%%sql

SELECT * FROM fact_sales LIMIT 5
```

```
 * postgresql://jnlosugsvinhfn:***@ec2-34-200-205-45.compute-1.amazonaws.com:5432/d2604bq39tv4uh
5 rows affected.
```

order_detail_id	order_date	product_id	customer_id	territory_id	sales_order_number	sales_order_line_number	quantity	unitprice_rupiah	totalprice_rupiah	totalcost_rupiah	shippingprice_rupiah
SO43698-1	2016-07-01 346	28389	7	SO43698	1	1	47599860	47599860	26770162	1189997	
SO43704-1	2016-07-02 351	11005	9	SO43704	1	1	47249860	47249860	26573322	1181247	
SO43705-1	2016-07-02 344	11011	9	SO43705	1	1	47599860	47599860	26770162	1189997	
SO43713-1	2016-07-05 310	27601	4	SO43713	1	1	50095780	50095780	30398119	1252395	
SO43714-1	2016-07-05 311	13591	10	SO43714	1	1	50095780	50095780	30398119	1252395	

Columns:

- order_detail_id > identifier (Primary Key)
- order_date
- product_id > references dim_product (Foreign Key)
- customer_id > references dim_customer (Foreign Key)
- territory_id > references dim_territory (Foreign Key)
- sales_order_number

Columns:

- sales_order_line_number
- quantity
- unitprice_rupiah
- totalprice_rupiah
- totalcost_rupiah
- shippingprice_rupiah

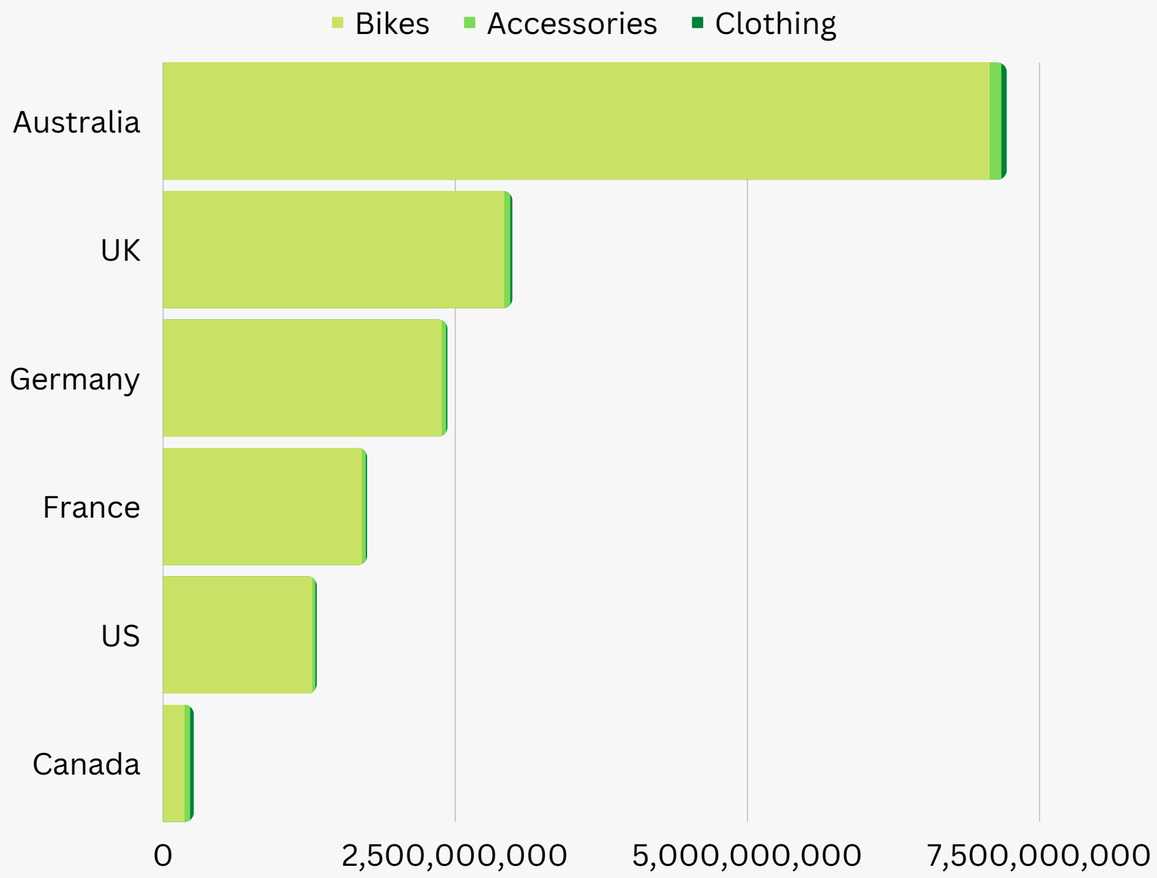
4. Exploring the Data

Creating some queries to explore & generate more information about the data.

What is the total revenue of each country branch in 2019?

```sql
SELECT
  country,
  category,
  COUNT(order_detail_id) AS sales_count,
  SUM(totalprice_rupiah) AS total_revenue
FROM
  fact_sales
    LEFT JOIN dim_territory USING(territory_id)
    LEFT JOIN dim_product USING(product_id)
WHERE DATE_PART('year', order_date) = 2019
GROUP BY 1,2
ORDER BY 1,4 DESC
```

country	category	sales_count	total_revenue
Australia	Bikes	248	7069198080
Australia	Accessories	332	101611860
Australia	Clothing	100	46214560
Canada	Bikes	6	180427520
Canada	Accessories	187	47421920
Canada	Clothing	55	30617580
France	Bikes	74	1698814040
France	Accessories	130	32497500
France	Clothing	25	10636920
Germany	Bikes	104	2381523760
Germany	Accessories	126	36639540
Germany	Clothing	30	11230940
United Kingdom	Bikes	118	2915663800
United Kingdom	Accessories	185	52359580
United Kingdom	Clothing	48	17430840
United States	Bikes	53	1272861380
United States	Accessories	104	26415900
United States	Clothing	27	11931640

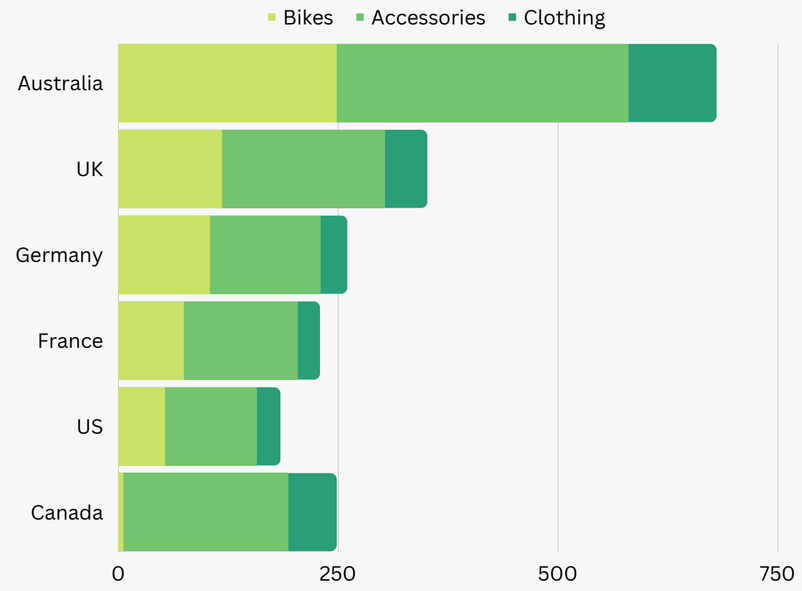


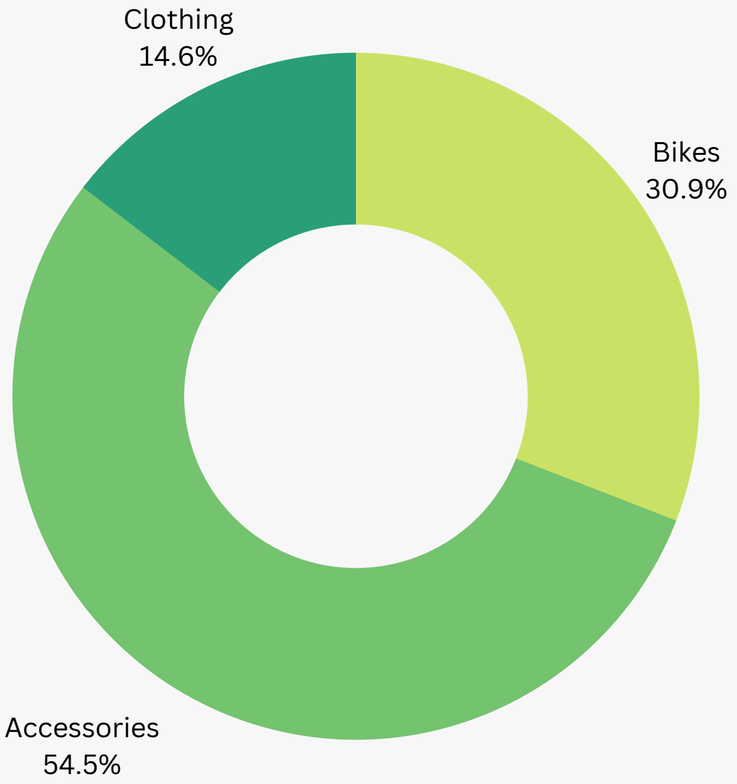
The total revenue of each country branch in 2019

The total revenue in 2019 was

15.9 Billion rupiahs

The total unit sales of each country branch in 2019





The total unit sales in 2019:

1952 pcs

Calculate profit margin for each sub category product

```sql
SELECT
  category,
  sub_category,
  ROUND(
    (totalprice_rupiah - totalcost_rupiah) * 100 / totalprice_rupiah
    ,2)AS total_margin
FROM
  fact_sales LEFT JOIN dim_product USING(product_id)
WHERE DATE_PART('year', order_date) = 2019
GROUP BY 1,2,3
ORDER BY 1,3 DESC
```

category	sub_category	total_margin
Accessories	Bike Racks	62.60
Accessories	Bike Stands	62.60
Accessories	Bottles and Cages	62.60
Accessories	Cleaners	62.60
Accessories	Fenders	62.60
Accessories	Helmets	62.60
Accessories	Hydration Packs	62.60
Accessories	Tires and Tubes	62.60
Bikes	Mountain Bikes	45.45
Bikes	Mountain Bikes	43.76
Bikes	Touring Bikes	37.84
Bikes	Road Bikes	36.36
Clothing	Gloves	62.60
Clothing	Shorts	62.60
Clothing	Socks	62.60
Clothing	Vests	62.60
Clothing	Caps	23.00
Clothing	Jerseys	23.00

Average profit margin for each category product

41%

50%

63%

What are the TOP 3 Bikes products with high turnover in each region?

```sql
WITH top3 AS(
  SELECT
    region,
    product_name,
    SUM(totalprice_rupiah) AS turnover,
    ROW_NUMBER() OVER(PARTITION BY region ORDER BY SUM(totalprice_rupiah) DESC)
      AS ranking
  FROM fact_sales
    LEFT JOIN dim_territory USING(territory_id)
    INNER JOIN dim_product USING(product_id)
  WHERE category = 'Bikes'
  GROUP BY 1, 2
)

SELECT *
FROM top3
WHERE ranking <=3
```

region	product_name	turnover	ranking
Australia	Mountain-200 Black, 42	2219255330	1
Australia	Mountain-200 Silver, 38	2192390536	2
Australia	Mountain-200 Silver, 42	1831632090	3
Canada	Road-150 Red, 48	651245140	1
Canada	Road-150 Red, 62	500957800	2
Canada	Road-150 Red, 44	400766240	3
France	Mountain-200 Black, 42	880128665	1
France	Mountain-200 Black, 46	833081370	2
France	Mountain-200 Silver, 46	744716780	3
Germany	Mountain-200 Silver, 46	869996248	1
Germany	Mountain-200 Silver, 38	545197648	2
Germany	Mountain-200 Black, 46	535880165	3
Northwest	Road-150 Red, 62	1302490280	1
Northwest	Road-150 Red, 48	1052011380	2
Northwest	Road-150 Red, 56	951819820	3
Southeast	Mountain-200 Silver, 38	32479860	1
Southwest	Road-150 Red, 48	1853543860	1
Southwest	Road-150 Red, 56	1552969180	2
Southwest	Road-150 Red, 62	1502873400	3
United Kingdom	Road-150 Red, 62	751436700	1
United Kingdom	Mountain-200 Black, 42	696529465	2
United Kingdom	Mountain-200 Silver, 42	668157116	3

What grouping age and gender have the highest transactions on Bee Cycle?

```sql
WITH tab1 AS(
  SELECT customer_id, gender, birthdate,
    CASE WHEN DATE_PART('year', current_date) - DATE_PART('year', birthdate) <= 20 THEN 'Group <=20'
         WHEN DATE_PART('year', current_date) - DATE_PART('year', birthdate) BETWEEN 21 AND 40 THEN 'Group 21-40'
         WHEN DATE_PART('year', current_date) - DATE_PART('year', birthdate) BETWEEN 41 AND 60 THEN 'Group 41-60'
    ELSE 'Group >60' END AS grouping_age,
  COUNT(DISTINCT order_detail_id) AS sales
  FROM fact_sales LEFT JOIN dim_customer USING(customer_id)
  GROUP BY 1,2,3
)

SELECT grouping_age, gender, SUM(sales) AS transaction
FROM tab1
group by 1,2
ORDER BY 3 DESC
```

grouping_age	gender	transaction
Group 21-40	F	1551
Group 41-60	F	1454
Group 21-40	M	1401
Group 41-60	M	1304
Group >60	M	161
Group >60	F	82
Group >60	None	1

Actionable Insights

- Communicate to the marketing team to create a discount coupon for bike products to get market interest & increase the revenue in any branch, due to the past revenue data in Australia.
- Discuss the previous customer behavior to get the most valuable user to apply the discount in each branch.
- Discuss with the marketing team to create a perfect bundle product that suits market and campaign needs.

Thank you

Love to discuss and learn,
feel free to connect.

EMAIL ADDRESS

muh.haniffaisaluddin@gmail.com

LINKEDIN

https://www.linkedin.com/in/mhfaisaluddin/

GITHUB

https://github.com/mhfaisaluddin