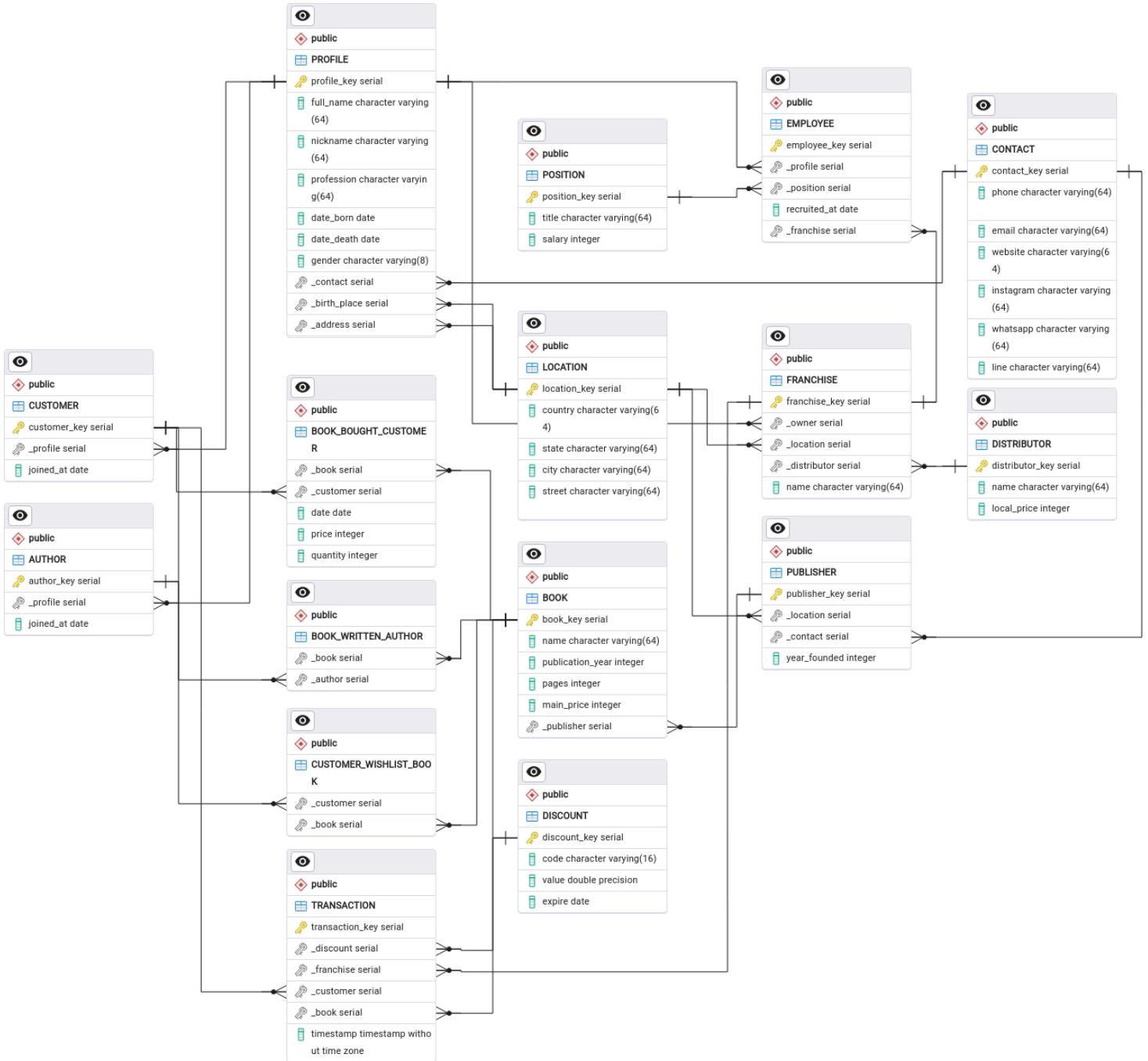


Database Technology

Final Project: Good Reading Bookstore (Part A)

Yitzhak Edmund Tio Manalu (22/499769/TK/54763)

A. Brand New ERD



The ERD above is an expansion of Good Reading Bookstore with the main consideration of data normalization. Also the data is **categorized** to three main categories, such as:

- Personal Profiles**,
- Book Details**, and
- Franchise Information**.

These categories are created to separate the information, yet could be integrated in a clean way to support the scalability of the database. Meaning that, for **Personal Profiles**, the information of roles like Customer,

Author, Employee, and owner of a franchise share the same structure of information. Also, for **Book Details**, the information related to the author, publisher, and other relations like wishlist, and transactions are managed in separate tables. Lastly, for **Franchise Information**, the ERD also managed to maintain information like the location of the franchise, the employees and their positions, also the owner of the franchise. This is useful for Good Reading Bookstore to track the expansion of the franchises across the country to enable better accessibility for customers outside Yogyakarta.

In order to explain more about the ERD, here are the brief **relationship explanations** of each table.

Personal Profiles

a. PROFILE

Containing major information about each person. Relates one-to-one relationships to roles like customer, author, employee, and contact. Yet, relates many-to-one relationships to location (for birth loc and address) as it is possible for some profiles to exist in the same location with each other.

b. CUSTOMER

Majorly contain data from the referenced profile with additional information of join date.

c. AUTHOR

Majorly contain data from the referenced profile with additional information of join date.

d. EMPLOYEE

Majorly contain data from the referenced profile with additional information of join date, the position, and franchise position with one-to-one relationship.

Book Details

a. BOOK

Contain every information about the book with reference to a many-to-one relationship to the publisher.

b. PUBLISHER

Contain every information about the publisher with relation only to books as mentioned before.

c. DISCOUNT

Performs as one-time-use code that is possible to expire. Has a one-to-one relationship to transaction.

d. TRANSACTION

Contain many integration to other tables like the customer, book, discount, and franchise location where the book will be taken from. Currently developed to count the prices from the applied discount but might be developed for distribution fee based on the customer's local location.

e. BOOK_BOUGHT_CUSTOMER

Basically the list of the books bought by the customer with many-to-many relationships.

f. CUSTOMER_WISHLIST_BOOK

Basically the relationship of many-to-many relationships to explain the books being in the wishlist of the customer.

g. BOOK_WRITTEN_AUTHOR

Basically the same as preceding tables, work as many-to-many relationships for authors contributing to the book.

Franchise Information

a. FRANCHISE

Contains information about the specific franchise and its location and distributor.

b. DISTRIBUTOR

Works as a one-to-one relationship to the franchise as one franchise has a local distributor for the incoming distribution requirements of the books that originated from the main store in Yogyakarta.

c. POSITION

This is basically to allow ease of position management that contains the salary that could be applied to certain employees.

Multifunctional Tables

a. CONTACT

Contains every possible contact such as phone number, instagram, website, whatsapp number, etc for either a person or a franchise.

b. LOCATION

Works like contact, contains every possible location aspects, such as country, state, city, and address for either a person or a franchise.

B. Brand New Database Schema (DDL)

As the ERD is created in pgAdmin, the definition of the table was also made using the features given in pgAdmin to generate the SQL. Here are the generated columns with their required constraints. In general, most keys are required but not for special conditions like discounts that might not be applied to transactions.

```

1 BEGIN;
2
3
4 CREATE TABLE IF NOT EXISTS public."AUTHOR"
5 (
6     author_key serial NOT NULL,
7     _profile serial NOT NULL,
8     joined_at date,
9     PRIMARY KEY (author_key)
10 );
11
12 CREATE TABLE IF NOT EXISTS public."CUSTOMER"
13 (
14     customer_key serial NOT NULL,
15     _profile serial NOT NULL,
16     joined_at date,
17     PRIMARY KEY (customer_key)
18 );
19
20 CREATE TABLE IF NOT EXISTS public."LOCATION"
21 (
22     location_key serial NOT NULL,
23     country character varying(64) NOT NULL DEFAULT '',
24     state character varying(64) NOT NULL DEFAULT '',
25     city character varying(64) NOT NULL DEFAULT '',
26     street character varying(64) NOT NULL DEFAULT '',
27     PRIMARY KEY (location_key)
28 );
29
30 CREATE TABLE IF NOT EXISTS public."PROFILE"
31 (
32     profile_key serial NOT NULL,
33     full_name character varying(64) NOT NULL DEFAULT '',
34     nickname character varying(64) NOT NULL DEFAULT '',
35     profession character varying(64) NOT NULL DEFAULT '',
36     date_born date NOT NULL,
37     date_death date,
38     gender character varying(8) NOT NULL DEFAULT 'Male',
39     _contact serial NOT NULL,
40     _birth_place serial NOT NULL,
41     _address serial NOT NULL,
42     PRIMARY KEY (profile_key)
43 );
44
45 CREATE TABLE IF NOT EXISTS public."PUBLISHER"
46 (
47     publisher_key serial NOT NULL,
48     _location serial NOT NULL,
49     _contact serial NOT NULL,
50     year_founded integer NOT NULL DEFAULT 0,
51     PRIMARY KEY (publisher_key)
52 );
53
54 CREATE TABLE IF NOT EXISTS public."CONTACT"
55 (
56     contact_key serial NOT NULL,
57     phone character varying(64),
58     email character varying(64),
59     website character varying(64),
60     instagram character varying(64),
61     whatsapp character varying(64),
62     line character varying(64),
63     PRIMARY KEY (contact_key)
64 );
65
66 CREATE TABLE IF NOT EXISTS public."BOOK"
67 (
68     book_key serial NOT NULL,
69     name character varying(64) NOT NULL DEFAULT '',
70     publication_year integer NOT NULL DEFAULT 0,
71     pages integer NOT NULL DEFAULT 0,
72     main_price integer NOT NULL DEFAULT 0,
73     _publisher serial NOT NULL,
74     PRIMARY KEY (book_key)
75 );
76
77 CREATE TABLE IF NOT EXISTS public."BOOK_BOUGHT_CUSTOMER"
78 (
79     _book serial NOT NULL,
80     _customer serial NOT NULL,
81     date date NOT NULL,
82     price integer NOT NULL DEFAULT 0,
83     quantity integer NOT NULL DEFAULT 0
84 );

```

```

1 CREATE TABLE IF NOT EXISTS public."BOOK_WRITTEN_AUTHOR"
2 (
3     _book serial NOT NULL,
4     _author serial NOT NULL
5 );
6
7 CREATE TABLE IF NOT EXISTS public."FRANCHISE"
8 (
9     franchise_key serial NOT NULL,
10    _owner serial NOT NULL,
11    _location serial NOT NULL,
12    _distributor serial NOT NULL,
13    name character varying(64),
14    PRIMARY KEY (franchise_key)
15 );
16
17 CREATE TABLE IF NOT EXISTS public."TRANSACTION"
18 (
19     transaction_key serial NOT NULL,
20     _discount serial,
21     _franchise serial NOT NULL,
22     _customer serial NOT NULL,
23     _book serial NOT NULL,
24     "timestamp" timestamp without time zone,
25     PRIMARY KEY (transaction_key)
26 );
27
28 CREATE TABLE IF NOT EXISTS public."DISCOUNT"
29 (
30     discount_key serial NOT NULL,
31     code character varying(16) NOT NULL DEFAULT '',
32     value double precision,
33     expire_date,
34     PRIMARY KEY (discount_key)
35 );
36
37 CREATE TABLE IF NOT EXISTS public."EMPLOYEE"
38 (
39     employee_key serial NOT NULL,
40     _profile serial NOT NULL,
41     _position serial NOT NULL,
42     recruited_at date,
43     _franchise serial NOT NULL,
44     PRIMARY KEY (employee_key)
45 );
46
47 CREATE TABLE IF NOT EXISTS public."POSITION"
48 (
49     position_key serial NOT NULL,
50     title character varying(64),
51     salary integer,
52     PRIMARY KEY (position_key)
53 );
54
55 CREATE TABLE IF NOT EXISTS public."DISTRIBUTOR"
56 (
57     distributor_key serial NOT NULL,
58     name character varying(64) NOT NULL DEFAULT '',
59     local_price integer NOT NULL DEFAULT 0,
60     PRIMARY KEY (distributor_key)
61 );
62
63 CREATE TABLE IF NOT EXISTS public."CUSTOMER_WISHLIST_BOOK"
64 (
65     _customer serial NOT NULL,
66     _book serial NOT NULL
67 );
68
69 ALTER TABLE IF EXISTS public."AUTHOR"
70     ADD FOREIGN KEY (_profile)
71     REFERENCES public."PROFILE" (profile_key) MATCH SIMPLE
72     ON UPDATE NO ACTION
73     ON DELETE NO ACTION
74     NOT VALID;
75
76
77 ALTER TABLE IF EXISTS public."CUSTOMER"
78     ADD FOREIGN KEY (_profile)
79     REFERENCES public."PROFILE" (profile_key) MATCH SIMPLE
80     ON UPDATE NO ACTION
81     ON DELETE NO ACTION
82     NOT VALID;
83

```

```

1 ALTER TABLE IF EXISTS public."PROFILE"
2   ADD FOREIGN KEY (_contact)
3     REFERENCES public."CONTACT" (contact_key) MATCH SIMPLE
4     ON UPDATE NO ACTION
5     ON DELETE NO ACTION
6     NOT VALID;
7
8
9 ALTER TABLE IF EXISTS public."PROFILE"
10  ADD FOREIGN KEY ( birth_place )
11    REFERENCES public."LOCATION" (location_key) MATCH SIMPLE
12    ON UPDATE NO ACTION
13    ON DELETE NO ACTION
14    NOT VALID;
15
16
17 ALTER TABLE IF EXISTS public."PROFILE"
18   ADD FOREIGN KEY (_address)
19     REFERENCES public."LOCATION" (location_key) MATCH SIMPLE
20     ON UPDATE NO ACTION
21     ON DELETE NO ACTION
22     NOT VALID;
23
24
25 ALTER TABLE IF EXISTS public."PUBLISHER"
26   ADD FOREIGN KEY (_location)
27     REFERENCES public."LOCATION" (location_key) MATCH SIMPLE
28     ON UPDATE NO ACTION
29     ON DELETE NO ACTION
30     NOT VALID;
31
32
33 ALTER TABLE IF EXISTS public."PUBLISHER"
34   ADD FOREIGN KEY (_contact)
35     REFERENCES public."CONTACT" (contact_key) MATCH SIMPLE
36     ON UPDATE NO ACTION
37     ON DELETE NO ACTION
38     NOT VALID;
39
40
41 ALTER TABLE IF EXISTS public."BOOK"
42   ADD FOREIGN KEY (_publisher)
43     REFERENCES public."PUBLISHER" (publisher_key) MATCH SIMPLE
44     ON UPDATE NO ACTION
45     ON DELETE NO ACTION
46     NOT VALID;
47
48
49 ALTER TABLE IF EXISTS public."BOOK_BOUGHT_CUSTOMER"
50   ADD FOREIGN KEY (_book)
51     REFERENCES public."BOOK" (book_key) MATCH SIMPLE
52     ON UPDATE NO ACTION
53     ON DELETE NO ACTION
54     NOT VALID;
55
56
57 ALTER TABLE IF EXISTS public."BOOK_BOUGHT_CUSTOMER"
58   ADD FOREIGN KEY (_customer)
59     REFERENCES public."CUSTOMER" (customer_key) MATCH SIMPLE
60     ON UPDATE NO ACTION
61     ON DELETE NO ACTION
62     NOT VALID;
63
64
65 ALTER TABLE IF EXISTS public."BOOK_WRITTEN_AUTHOR"
66   ADD FOREIGN KEY (_book)
67     REFERENCES public."BOOK" (book_key) MATCH SIMPLE
68     ON UPDATE NO ACTION
69     ON DELETE NO ACTION
70     NOT VALID;
71
72
73 ALTER TABLE IF EXISTS public."BOOK_WRITTEN_AUTHOR"
74   ADD FOREIGN KEY (_author)
75     REFERENCES public."AUTHOR" (author_key) MATCH SIMPLE
76     ON UPDATE NO ACTION
77     ON DELETE NO ACTION
78     NOT VALID;

```

```

1 ALTER TABLE IF EXISTS public."FRANCHISE"
2   ADD FOREIGN KEY (_owner)
3     REFERENCES public."PROFILE" (profile_key) MATCH SIMPLE
4     ON UPDATE NO ACTION
5     ON DELETE NO ACTION
6     NOT VALID;
7
8
9 ALTER TABLE IF EXISTS public."FRANCHISE"
10  ADD FOREIGN KEY (_location)
11    REFERENCES public."LOCATION" (location_key) MATCH SIMPLE
12    ON UPDATE NO ACTION
13    ON DELETE NO ACTION
14    NOT VALID;
15
16
17 ALTER TABLE IF EXISTS public."FRANCHISE"
18   ADD FOREIGN KEY (_distributor)
19     REFERENCES public."DISTRIBUTOR" (distributor_key) MATCH SIMPLE
20     ON UPDATE NO ACTION
21     ON DELETE NO ACTION
22     NOT VALID;
23
24
25 ALTER TABLE IF EXISTS public."TRANSACTION"
26   ADD FOREIGN KEY (_discount)
27     REFERENCES public."DISCOUNT" (discount_key) MATCH SIMPLE
28     ON UPDATE NO ACTION
29     ON DELETE NO ACTION
30     NOT VALID;
31
32
33 ALTER TABLE IF EXISTS public."TRANSACTION"
34   ADD FOREIGN KEY (_franchise)
35     REFERENCES public."FRANCHISE" (franchise_key) MATCH SIMPLE
36     ON UPDATE NO ACTION
37     ON DELETE NO ACTION
38     NOT VALID;
39
40
41 ALTER TABLE IF EXISTS public."TRANSACTION"
42   ADD FOREIGN KEY (_customer)
43     REFERENCES public."CUSTOMER" (customer_key) MATCH SIMPLE
44     ON UPDATE NO ACTION
45     ON DELETE NO ACTION
46     NOT VALID;
47
48
49 ALTER TABLE IF EXISTS public."TRANSACTION"
50   ADD FOREIGN KEY ( book )
51     REFERENCES public."BOOK" (book_key) MATCH SIMPLE
52     ON UPDATE NO ACTION
53     ON DELETE NO ACTION
54     NOT VALID;
55
56
57 ALTER TABLE IF EXISTS public."EMPLOYEE"
58   ADD FOREIGN KEY ( profile )
59     REFERENCES public."PROFILE" (profile_key) MATCH SIMPLE
60     ON UPDATE NO ACTION
61     ON DELETE NO ACTION
62     NOT VALID;
63
64
65 ALTER TABLE IF EXISTS public."EMPLOYEE"
66   ADD FOREIGN KEY ( position )
67     REFERENCES public."POSITION" (position_key) MATCH SIMPLE
68     ON UPDATE NO ACTION
69     ON DELETE NO ACTION
70     NOT VALID;
71
72
73 ALTER TABLE IF EXISTS public."EMPLOYEE"
74   ADD FOREIGN KEY ( franchise )
75     REFERENCES public."FRANCHISE" (franchise_key) MATCH SIMPLE
76     ON UPDATE NO ACTION
77     ON DELETE NO ACTION
78     NOT VALID;
79
80
81 ALTER TABLE IF EXISTS public."CUSTOMER_WISHLIST_BOOK"
82   ADD FOREIGN KEY (_customer)
83     REFERENCES public."CUSTOMER" (customer_key) MATCH SIMPLE
84     ON UPDATE NO ACTION
85     ON DELETE NO ACTION
86     NOT VALID;
87
88
89 ALTER TABLE IF EXISTS public."CUSTOMER_WISHLIST_BOOK"
90   ADD FOREIGN KEY ( book )
91     REFERENCES public."BOOK" (book_key) MATCH SIMPLE
92     ON UPDATE NO ACTION
93     ON DELETE NO ACTION
94     NOT VALID;
95
96 END;

```

C. SQL Population and Views

For the record, every data given here was generated using ChatGPT with predefined context to support localized names and consistency between the other tables.

Populations

1. Location

```
● ● ●
1 -- @block
2 -- LOCATION
3 BEGIN;
4
5 INSERT INTO public."LOCATION" (country, state, city, street)
6 VALUES
7 ('Indonesia', 'Yogyakarta', 'Yogyakarta', 'Jalan Malioboro No. 123'),
8 ('Indonesia', 'Jakarta', 'Jakarta', 'Jalan Sudirman No. 234'),
9 ('Indonesia', 'Bali', 'Denpasar', 'Jalan Sunset No. 345'),
10 ('Indonesia', 'Java Barat', 'Bandung', 'Jalan Braga No. 456'),
11 ('Indonesia', 'Java Tengah', 'Semarang', 'Jalan Pahlawan No. 567'),
12 ('Indonesia', 'Java Timur', 'Surabaya', 'Jalan Tunjungan No. 678'),
13 ('Indonesia', 'Sumatera Utara', 'Medan', 'Jalan Thamrin No. 901'),
14 ('Indonesia', 'Sumatera Selatan', 'Palembang', 'Jalan Demang Lebar Daun No. 112'),
15 ('Indonesia', 'Kalimantan Timur', 'Balikpapan', 'Jalan Jendral Sudirman No. 223'),
16 ('Indonesia', 'Sulawesi Utara', 'Manado', 'Jalan Sam Ratulangi No. 334'),
17 ('Indonesia', 'Papua Barat', 'Sorong', 'Jalan Basuki Rahmat No. 445'),
18 ('Indonesia', 'Nusa Tenggara Timur', 'Kupang', 'Jalan El Tari No. 556'),
19 ('Indonesia', 'Aceh', 'Banda Aceh', 'Jalan Teuku Umar No. 667'),
20 ('Indonesia', 'Aceh', 'Lhokseumawe', 'Jalan Tengku Agam No. 778'),
21 ('Indonesia', 'Aceh', 'Langsa', 'Jalan Ahmad Yani No. 889'),
22 ('Indonesia', 'Maluku', 'Amboin', 'Jalan Raya Pattimura No. 990'),
23 ('Indonesia', 'Gorontalo', 'Gorontalo', 'Jalan Sultan Hasanuddin No. 1101'),
24 ('Indonesia', 'Lampung', 'Bandar Lampung', 'Jalan Zainal Abidin No. 1212'),
25 ('Indonesia', 'Sumatera Barat', 'Padang', 'Jalan Hayam Wuruk No. 2323'),
26 ('Indonesia', 'Banten', 'Serang', 'Jalan KH Abdul Hamid No. 3434'),
27 ('Indonesia', 'Banten', 'Tangerang', 'Jalan Gatot Subroto No. 4545'),
28 ('Indonesia', 'Yogyakarta', 'Sleman', 'Jalan Magelang No. 5656'),
29 ('Indonesia', 'Yogyakarta', 'Bantul', 'Jalan Parangtritis No. 6767'),
30 ('Indonesia', 'Yogyakarta', 'Gunung Kidul', 'Jalan Wonosari No. 7878'),
31 ('Indonesia', 'Yogyakarta', 'Kulon Progo', 'Jalan Wates No. 8989');
32
33 END;
```

The location is mainly informed to be in Indonesia. Names are based on Indonesian addresses that contain the only country, Indonesia, the state, the city, and also the street.

This table is used for **both** person and franchise.

2. Contact

```
● ● ●
1 -- @block
2 -- CONTACT
3 BEGIN;
4
5 -- For Users
6 INSERT INTO public."CONTACT" (phone, email, website, instagram, whatsapp, line)
7 VALUES
8 ('081234567890', 'yitzhak@gmail.com', '', '@bukuindonesia', '+6281234567890', 'yitzhak_line'),
9 ('082345678901', 'edmund@gmail.com', '', '@penerbitnusantara', '+6282345678901', 'edmund_line'),
10 ('081112233445', 'aline@gmail.com', '', '@tokobukujogja', '+6281112233445', 'aline_line'),
11 ('087654321098', 'surya@mail.com', '', '@gramedia', '+6287654321098', 'surya_line'),
12 ('085678901234', 'amalia@gmail.com', '', '@gramediaselatan', '+6285678901234', 'amalia_line'),
13 ('083456789012', 'aditya@gmail.com', '', '@gramediatimur', '+6283456789012', 'aditya_line'),
14 ('089012345678', 'trisna@gmail.com', '', '@tokobukusby', '+6289012345678', 'trisna_line'),
15 ('081231231231', 'ratih@gmail.com', '', '@bukuonline', '+6281231231231', 'ratih_line'),
16 ('082123123132', 'danang@gmail.com', '', '@ustakajaya', '+6282123123132', 'danang_line'),
17 ('0842123123214', 'nadia@gmail.com', '', '@bukuremaja', '+62842123123214', 'nadia_line'),
18 ('087987654321', 'jojo@gmail.com', '', '@gramediamks', '+6287987654321', 'jojo_line'),
19 ('089987654321', 'nina@gmail.com', '', '@tokobukuaceh', '+6289987654321', 'nina_line'),
20 ('081098765432', 'merry@gmail.com', '', '@tokobukubdg', '+6281098765432', 'merry_line'),
21 ('083987654321', 'daniel@gmail.com', '', '@bukuonline', '+6283987654321', 'daniel_line'),
22 ('081234567891', 'linda@gmail.com', '', '@bukubagus', '+6281234567891', 'linda_line'),
23 ('081234567890', 'putri.utami@gmail.com', '', 'putriutami', '+6281234567890', 'putriutami_line'),
24 ('081234567891', 'budi.santoso@gmail.com', '', 'budisantoso', '+6281234567891', 'budisantoso_line'),
25 ('081234567892', 'sri.wijaya@gmail.com', '', 'sriwijaya', '+6281234567892', 'sriwijaya_line'),
26 ('081234567893', 'adi.nugroho@gmail.com', '', 'adinugroho', '+6281234567893', 'adinugroho_line'),
27 ('081234567894', 'dewi.angraini@gmail.com', '', 'dewi.angraini', '+6281234567894', 'dewi.angraini_line'),
28
29 -- For Franchises
30 INSERT INTO public."CONTACT" (phone, email, website, instagram, whatsapp, line)
31 VALUES
32 ('085098765432', 'penerbitsejahtera@penerbitsejahtera.co.id', 'www.penerbitsejahtera.co.id', '@penerbitsejahtera', '+6285098765432', 'ani_line'),
33 ('087987654321', 'gramediasmg@gramediasmg.co.id', 'www.gramediasmg.co.id', '@gramediasmg', '+6287987654321', 'dika_line'),
34 ('0899098765432', 'tokobukuuny@tokobukuuny.co.id', 'www.tokobukuuny.co.id', '@tokobukuuny', '+6289098765432', 'andi_line'),
35 ('081198765432', 'penerbitgembira@penerbitgembira.co.id', 'www.penerbitgembira.co.id', '@penerbitgembira', '+6281198765432', 'tina_line'),
36 ('083198765432', 'bukubagus@bukubagus.co.id', 'www.bukubagus.co.id', '@bukubagus', '+6283198765432', 'rizki_line');
37
38 END;
```

Working the same way as Location, the Contact was designed to contain data for **both** users (person) and franchises. Here, the normal users are considered having no website for the contact.

3. Position

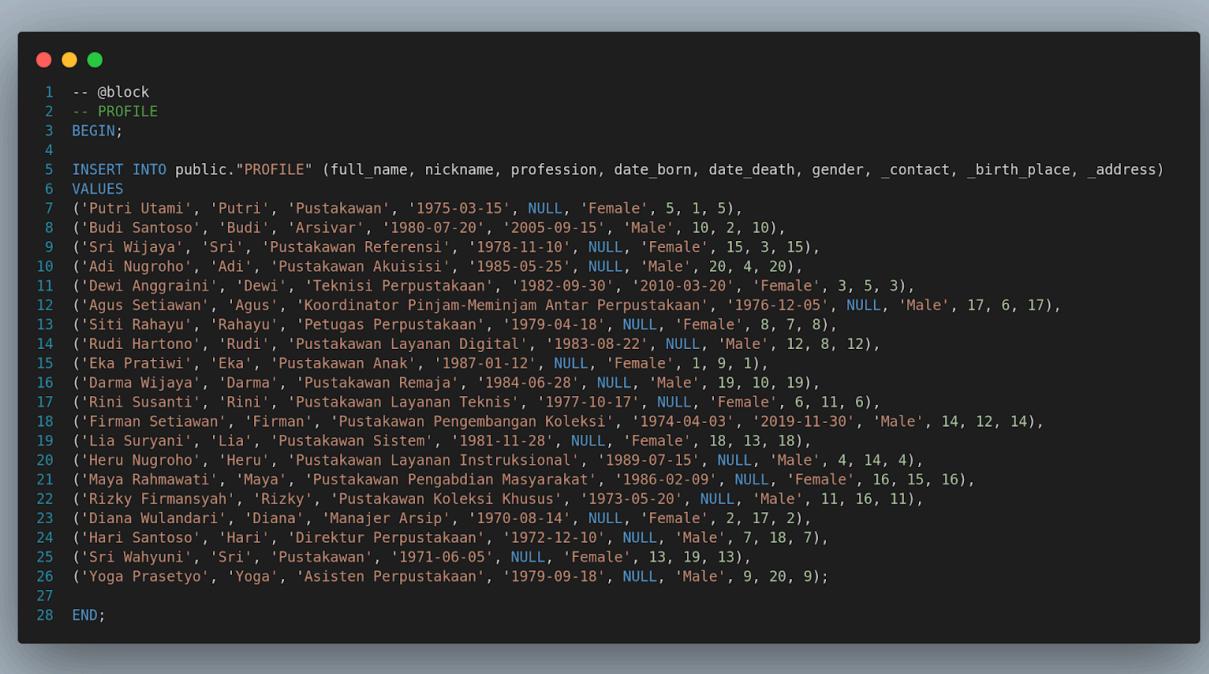


```
1 -- @block
2 -- POSITION
3 INSERT INTO public."POSITION" (title, salary)
4 VALUES
5 ('Pustakawan', 5500000),
6 ('Asisten Perpustakaan', 5200000),
7 ('Spesialis Katalogisasi', 5800000),
8 ('Arsivar', 5400000),
9 ('Pustakawan Referensi', 5900000),
10 ('Pustakawan Akuisisi', 5600000),
11 ('Teknisi Perpustakaan', 5100000),
12 ('Koordinator Pinjam-Meminjam Antar Perpustakaan', 5700000),
13 ('Petugas Perpustakaan', 4800000),
14 ('Pustakawan Layanan Digital', 5600000),
15 ('Pustakawan Anak', 5400000),
16 ('Pustakawan Remaja', 5400000),
17 ('Pustakawan Layanan Teknis', 5800000),
18 ('Pustakawan Pengembangan Koleksi', 5800000),
19 ('Pustakawan Sistem', 6200000),
20 ('Pustakawan Layanan Instruksional', 5800000),
21 ('Pustakawan Pengabdian Masyarakat', 5300000),
22 ('Pustakawan Koleksi Khusus', 6100000),
23 ('Manajer Arsip', 6300000),
24 ('Direktur Perpustakaan', 7200000);
```

The position here was also localized by using Indonesian roles. Also the salary given is based on Indonesian salary starting from 4 million Rupiah.

This data might be used to count the expense required for the business to pay their HR monthly.

4. Profile



```
1 -- @block
2 -- PROFILE
3 BEGIN;
4
5 INSERT INTO public."PROFILE" (full_name, nickname, profession, date_born, date_death, gender, _contact, _birth_place, _address)
6 VALUES
7 ('Putri Utami', 'Putri', 'Pustakawan', '1975-03-15', NULL, 'Female', 5, 1, 5),
8 ('Budi Santoso', 'Budi', 'Arsivar', '1980-07-20', '2005-09-15', 'Male', 10, 2, 10),
9 ('Sri Wijaya', 'Sri', 'Pustakawan Referensi', '1978-11-10', NULL, 'Female', 15, 3, 15),
10 ('Adi Nugroho', 'Adi', 'Pustakawan Akuisisi', '1985-05-25', NULL, 'Male', 20, 4, 20),
11 ('Dewi Anggraini', 'Dewi', 'Teknisi Perpustakaan', '1982-09-30', '2010-03-20', 'Female', 3, 5, 3),
12 ('Agus Setiawan', 'Agus', 'Koordinator Pinjam-Meminjam Antar Perpustakaan', '1976-12-05', NULL, 'Male', 17, 6, 17),
13 ('Siti Rahayu', 'Rahayu', 'Petugas Perpustakaan', '1979-04-18', NULL, 'Female', 8, 7, 8),
14 ('Rudi Hartono', 'Rudi', 'Pustakawan Layanan Digital', '1983-08-22', NULL, 'Male', 12, 8, 12),
15 ('Eka Pratiwi', 'Eka', 'Pustakawan Anak', '1987-01-12', NULL, 'Female', 1, 9, 1),
16 ('Darma Wijaya', 'Darma', 'Pustakawan Remaja', '1984-06-28', NULL, 'Male', 19, 10, 19),
17 ('Rini Susanti', 'Rini', 'Pustakawan Layanan Teknis', '1977-10-17', NULL, 'Female', 6, 11, 6),
18 ('Firman Setiawan', 'Firman', 'Pustakawan Pengembangan Koleksi', '1974-04-03', '2019-11-30', 'Male', 14, 12, 14),
19 ('Lia Suryani', 'Lia', 'Pustakawan Sistem', '1981-11-28', NULL, 'Female', 18, 13, 18),
20 ('Heru Nugroho', 'Heru', 'Pustakawan Layanan Instruksional', '1989-07-15', NULL, 'Male', 4, 14, 4),
21 ('Maya Rahmawati', 'Maya', 'Pustakawan Pengabdian Masyarakat', '1986-02-09', NULL, 'Female', 16, 15, 16),
22 ('Rizky Firmansyah', 'Rizky', 'Pustakawan Koleksi Khusus', '1973-05-20', NULL, 'Male', 11, 16, 11),
23 ('Diana Wulandari', 'Diana', 'Manajer Arsip', '1970-08-14', NULL, 'Female', 2, 17, 2),
24 ('Hari Santoso', 'Hari', 'Direktur Perpustakaan', '1972-12-10', NULL, 'Male', 7, 18, 7),
25 ('Sri Wahyuni', 'Sri', 'Pustakawan', '1971-06-05', NULL, 'Female', 13, 19, 13),
26 ('Yoga Prasetyo', 'Yoga', 'Asisten Perpustakaan', '1979-09-18', NULL, 'Male', 9, 20, 9);
27
28 END;
```

As mentioned before, the profile is described to have personal information.

5. Author

```
1 -- @block
2 BEGIN;
3
4 -- AUTHOR
5 INSERT INTO public."AUTHOR" (_profile, joined_at)
6 VALUES
7 (3, '2021-02-14'),
8 (17, '2020-05-23'),
9 (6, '2019-11-30'),
10 (11, '2020-07-15'),
11 (8, '2021-08-20'),
12 (14, '2020-12-01'),
13 (2, '2021-03-10'),
14 (19, '2019-09-25'),
15 (5, '2021-01-05'),
16 (12, '2020-06-18'),
17 (1, '2021-04-22'),
18 (15, '2020-08-09'),
19 (4, '2021-07-14'),
20 (10, '2019-10-03'),
21 (7, '2020-03-19'),
22 (18, '2021-09-27'),
23 (9, '2020-11-11'),
24 (20, '2021-05-31'),
25 (13, '2019-12-21'),
26 (16, '2020-02-06');
27
28 END;
```

The query here is quite simple as it simply integrates an author with a predefined profile to a joining date information.

6. Customer

```
1 -- @block
2 -- CUSTOMER
3 BEGIN;
4
5 INSERT INTO public."CUSTOMER" (_profile, joined_at)
6 VALUES
7 (15, '2020-11-14'),
8 (3, '2021-02-25'),
9 (12, '2020-06-10'),
10 (8, '2021-09-03'),
11 (18, '2020-12-18'),
12 (5, '2021-03-11'),
13 (9, '2020-01-22'),
14 (14, '2021-04-27'),
15 (1, '2020-05-19'),
16 (17, '2021-07-30'),
17 (10, '2020-08-08'),
18 (4, '2021-11-21'),
19 (20, '2020-03-14'),
20 (7, '2021-10-07'),
21 (2, '2020-02-11'),
22 (13, '2021-06-16'),
23 (6, '2020-04-29'),
24 (19, '2021-05-05'),
25 (16, '2020-09-12'),
26 (11, '2021-01-01');
27
28 END;
```

Works with the same logic as Location but for another person role, Customer.

7. Distributor

```
1 -- @block
2 BEGIN;
3
4 INSERT INTO public."DISTRIBUTOR" (name, local_price)
5 VALUES
6 ('Maju Jaya', 15000),
7 ('Mitra Sejahtera', 16000),
8 ('Berkah Abadi', 15500),
9 ('Bintang Makmur', 17000),
10 ('Agung Jaya', 16500),
11 ('Sinar Makmur', 15500),
12 ('Maju Bersama', 16000),
13 ('Sentosa Jaya', 14500),
14 ('Makmur Jaya', 17000),
15 ('Indah Sejahtera', 15500),
16 ('Jaya Abadi', 16500),
17 ('Maju Sejahtera', 15000),
18 ('Makmur Bersama', 17500),
19 ('Berkah Maju', 16000),
20 ('Harapan Jaya', 15500),
21 ('Berkah Sejahtera', 16500),
22 ('Maju Makmur', 16000),
23 ('Berkah Bersama', 15000),
24 ('Jaya Makmur', 17000),
25 ('Maju Sentosa', 15500);
26
27 END;
```

Contains information about the names of specific distributors with the local price of distribution fee for the customers around the franchise location.

8. Franchise

```
1 -- @block
2 -- FRANCHISE
3 BEGIN;
4
5 INSERT INTO public."FRANCHISE" (_owner, _location, _distributor, name)
6 VALUES
7 (5, 18, 3, 'Warung Buku Ceria'),
8 (10, 7, 14, 'Toko Buku Mutiara'),
9 (15, 12, 1, 'Buku Impian'),
10 (20, 19, 10, 'Pustaka Abadi'),
11 (3, 5, 16, 'Surga Buku'),
12 (8, 3, 20, 'Tokobuku Bahagia'),
13 (13, 9, 7, 'Buku Ajaib'),
14 (18, 17, 4, 'Warung Buku Sejahtera'),
15 (1, 2, 15, 'Pustaka Jaya'),
16 (6, 16, 9, 'Buku Inspirasi'),
17 (11, 11, 6, 'Toko Buku Serba Ada'),
18 (16, 13, 11, 'Buku Terang'),
19 (4, 6, 18, 'Toko Buku Kreatif'),
20 (9, 4, 8, 'Buku Berkah'),
21 (14, 1, 17, 'Toko Buku Cerdas'),
22 (19, 20, 5, 'Pustaka Bintang'),
23 (2, 8, 12, 'Buku Pintar'),
24 (7, 14, 19, 'Toko Buku Modern'),
25 (12, 10, 2, 'Buku Ilmiah'),
26 (17, 15, 13, 'Pustaka Cemerlang');
27
28 END;
```

Includes information of the name of a franchise and the corresponding location, distributor, and owner.

This is to track the franchises of the book store in various locations across the country.

9. Employee

```
● ● ●
1 -- @block
2 -- EMPLOYEE
3 BEGIN;
4
5 INSERT INTO public."EMPLOYEE" (_profile, _position, recruited_at, _franchise)
6 VALUES
7 (3, 7, '2021-05-10', 5),
8 (8, 4, '2020-08-15', 10),
9 (12, 9, '2021-03-25', 15),
10 (5, 2, '2020-10-01', 3),
11 (18, 6, '2021-07-20', 17),
12 (9, 8, '2020-11-30', 1),
13 (14, 3, '2021-04-05', 12),
14 (1, 10, '2020-01-15', 7),
15 (17, 1, '2021-09-10', 19),
16 (10, 5, '2020-02-20', 14),
17 (4, 11, '2021-06-30', 16),
18 (20, 7, '2020-03-12', 11),
19 (7, 4, '2021-11-05', 9),
20 (2, 9, '2020-04-18', 4),
21 (13, 2, '2021-08-22', 1),
22 (6, 6, '2020-12-07', 15),
23 (19, 8, '2021-01-25', 5),
24 (16, 3, '2020-05-28', 17),
25 (11, 10, '2021-10-15', 3),
26 (15, 5, '2020-07-01', 16);
27
28 END;
```

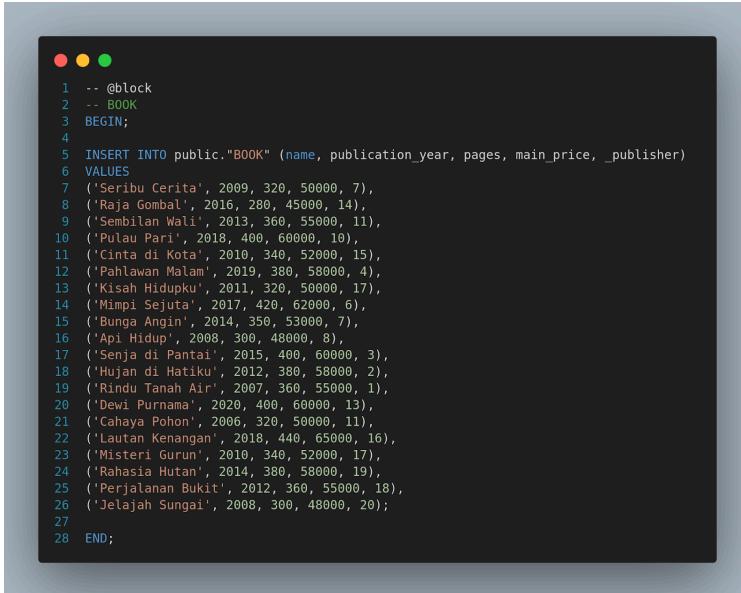
Basically performs the same as the rest profiles, yet includes the specific information about the work location and job position.

10. Publisher

```
● ● ●
1 -- @block
2 -- PUBLISHER
3 BEGIN;
4
5 INSERT INTO public."PUBLISHER" (_location, _contact, year_founded)
6 VALUES
7 (6, 16, 1995),
8 (8, 10, 1988),
9 (12, 18, 2000),
10 (4, 14, 1979),
11 (9, 3, 1998),
12 (2, 7, 1985),
13 (15, 11, 2005),
14 (10, 17, 1973),
15 (7, 19, 1980),
16 (13, 5, 1990),
17 (11, 9, 2003),
18 (3, 1, 1975),
19 (16, 20, 1983),
20 (5, 13, 1993),
21 (20, 2, 1988),
22 (17, 12, 1978),
23 (1, 15, 1997),
24 (18, 8, 2001),
25 (19, 4, 1971),
26 (14, 6, 1987);
27
28 END;
```

Simply to define the publishers that exist to publish the books and the supporting information such as contact and year founded.

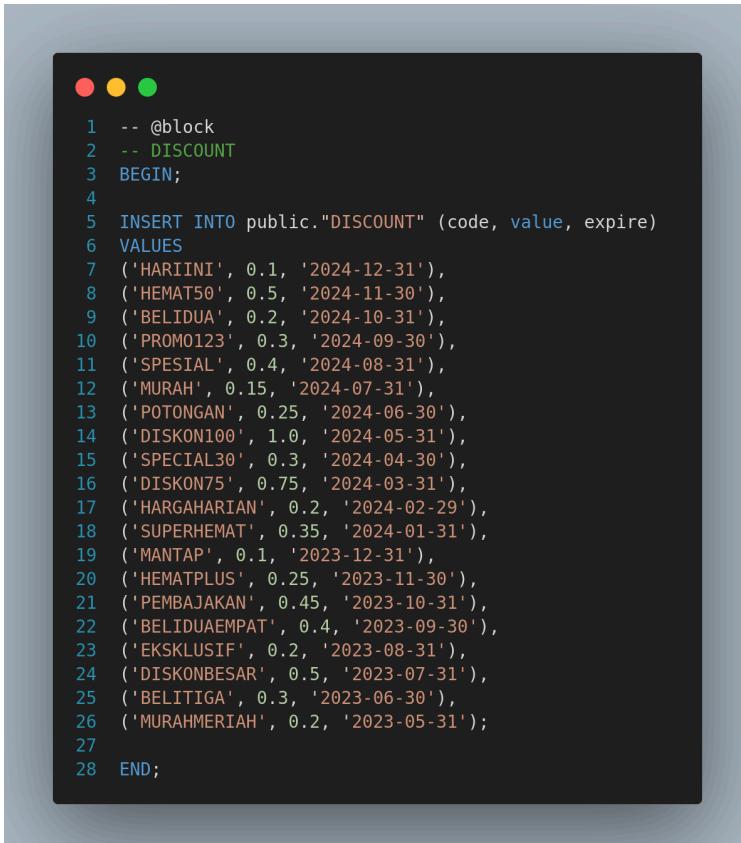
11. Book



```
1 -- @block
2 -- BOOK
3 BEGIN;
4
5 INSERT INTO public."BOOK" (name, publication_year, pages, main_price, _publisher)
6 VALUES
7 ('Seriibu Cerita', 2009, 320, 50000, 7),
8 ('Raja Gombal', 2016, 280, 45000, 14),
9 ('Sembilan Wali', 2013, 360, 55000, 11),
10 ('Pulau Pari', 2018, 400, 60000, 10),
11 ('Cinta di Kota', 2018, 340, 52000, 15),
12 ('Pahlawan Malam', 2019, 380, 58000, 4),
13 ('Kisah Hidupku', 2011, 320, 50000, 17),
14 ('Mimpi Sejuta', 2017, 420, 62000, 6),
15 ('Bunga Angin', 2014, 350, 53000, 7),
16 ('Api Hidup', 2008, 300, 48000, 8),
17 ('Senja di Pantai', 2015, 400, 60000, 3),
18 ('Hujan di Hatiku', 2012, 380, 58000, 2),
19 ('Rindu Tanah Air', 2007, 360, 55000, 1),
20 ('Dewi Purnama', 2020, 400, 60000, 13),
21 ('Cahaya Pohon', 2006, 320, 50000, 11),
22 ('Lautan Kenangan', 2018, 440, 65000, 16),
23 ('Misteri Gurun', 2010, 340, 52000, 17),
24 ('Rahasia Hutan', 2014, 380, 58000, 19),
25 ('Perjalanan Bukit', 2012, 360, 55000, 18),
26 ('Jelajah Sungai', 2008, 300, 48000, 20);
27
28 END;
```

The core information of books, the details, and most importantly the price that will be the main information required for transactions.

12. Discount



```
1 -- @block
2 -- DISCOUNT
3 BEGIN;
4
5 INSERT INTO public."DISCOUNT" (code, value, expire)
6 VALUES
7 ('HARIINI', 0.1, '2024-12-31'),
8 ('HEMAT50', 0.5, '2024-11-30'),
9 ('BELIDUA', 0.2, '2024-10-31'),
10 ('PROMO123', 0.3, '2024-09-30'),
11 ('SPESIAL', 0.4, '2024-08-31'),
12 ('MURAH', 0.15, '2024-07-31'),
13 ('POTONGAN', 0.25, '2024-06-30'),
14 ('DISKON100', 1.0, '2024-05-31'),
15 ('SPECIAL30', 0.3, '2024-04-30'),
16 ('DISKON75', 0.75, '2024-03-31'),
17 ('HARGAHARIAN', 0.2, '2024-02-29'),
18 ('SUPERHEMAT', 0.35, '2024-01-31'),
19 ('MANTAP', 0.1, '2023-12-31'),
20 ('HEMATPLUS', 0.25, '2023-11-30'),
21 ('PEMBAJAKAN', 0.45, '2023-10-31'),
22 ('BELIDUAEMPAT', 0.4, '2023-09-30'),
23 ('EKSKLUSIF', 0.2, '2023-08-31'),
24 ('DISKONBESAR', 0.5, '2023-07-31'),
25 ('BELITIGA', 0.3, '2023-06-30'),
26 ('MURAHMERIAH', 0.2, '2023-05-31');
27
28 END;
```

Works as single-use codes that could expire. This is to allow the bookstore to provide more promo by inserting more limited codes here. The codes are also localized to Indonesian.

13. Transaction

```
1 -- @block
2 -- TRANSACTION
3 BEGIN;
4
5 INSERT INTO public."TRANSACTION" (_discount, _franchise, _customer, _book, timestamp)
6 VALUES
7 (4, 10, 8, 14, '2024-01-15 08:30:00'),
8 (NULL, 5, 15, 20, '2024-02-20 10:45:00'),
9 (1, 17, 3, 3, '2024-03-25 14:20:00'),
10 (9, 12, 11, 19, '2024-04-10 16:55:00'),
11 (NULL, 20, 19, 10, '2024-05-05 12:10:00'),
12 (NULL, 8, 1, 1, '2024-06-10 09:25:00'),
13 (NULL, 13, 14, 8, '2024-07-20 11:30:00'),
14 (8, 2, 6, 2, '2024-08-28 13:40:00'),
15 (6, 18, 12, 12, '2024-09-05 15:15:00'),
16 (10, 7, 16, 4, '2024-10-10 17:20:00'),
17 (11, 15, 4, 17, '2024-11-12 18:45:00'),
18 (NULL, 4, 9, 9, '2024-12-17 19:50:00'),
19 (NULL, 19, 18, 18, '2025-01-22 20:30:00'),
20 (14, 6, 2, 6, '2025-02-25 21:55:00'),
21 (15, 1, 17, 13, '2025-03-30 22:40:00'),
22 (16, 16, 13, 5, '2025-04-05 23:10:00'),
23 (NULL, 3, 7, 7, '2025-05-10 08:30:00'),
24 (18, 11, 10, 11, '2025-06-15 09:45:00'),
25 (19, 14, 5, 16, '2025-07-20 10:20:00'),
26 (NULL, 9, 20, 15, '2025-08-25 11:35:00');
27
28 END;
```

The table will be the most used as it is used to track all the interaction between the customers and the existing books in the book store.

14. Book Written By

```
1 -- @block
2 -- BOOK_WRITTEN_AUTHOR
3 BEGIN;
4
5 INSERT INTO public."BOOK_WRITTEN_AUTHOR" (_book, _author)
6 VALUES
7 (9, 15),
8 (16, 2),
9 (4, 8),
10 (13, 18),
11 (2, 7),
12 (5, 20),
13 (3, 12),
14 (1, 4),
15 (12, 9),
16 (6, 1),
17 (17, 13),
18 (15, 6),
19 (18, 5),
20 (10, 3),
21 (7, 11),
22 (14, 16),
23 (11, 19),
24 (8, 10),
25 (20, 14),
26 (19, 15),
27 (3, 18),
28 (14, 6),
29 (1, 13),
30 (17, 8),
31 (5, 11);
32
33 END;
```

This is an intermediary relationship between books and the author of the book.

15. Books Bought

```
1 -- @block
2 -- BOOK_BOUGHT_CUSTOMER
3 BEGIN;
4
5 INSERT INTO public."BOOK_BOUGHT_CUSTOMER" (_book, _customer, date, price, quantity)
6 VALUES
7 (1, 10, '2024-01-15', 100000, 2),
8 (2, 2, '2024-02-20', 150000, 1),
9 (3, 6, '2024-03-25', 80000, 3),
10 (4, 1, '2024-04-10', 120000, 1),
11 (5, 18, '2024-05-05', 90000, 2),
12 (6, 13, '2024-06-15', 110000, 1),
13 (7, 5, '2024-07-20', 130000, 2),
14 (8, 16, '2024-08-28', 95000, 1),
15 (9, 19, '2024-09-05', 70000, 3),
16 (10, 3, '2024-10-10', 135000, 1),
17 (11, 9, '2024-11-12', 105000, 2),
18 (12, 15, '2024-12-17', 125000, 1),
19 (13, 14, '2025-01-22', 100000, 2),
20 (14, 20, '2025-02-25', 145000, 1),
21 (15, 7, '2025-03-30', 80000, 3),
22 (16, 8, '2025-04-05', 115000, 1),
23 (17, 11, '2025-05-10', 95000, 2),
24 (18, 4, '2025-06-15', 120000, 1),
25 (19, 17, '2025-07-20', 110000, 2),
26 (20, 12, '2025-08-25', 130000, 1);
27
28 END;
```

16. Customer-Book (Wishlist)

```
1 -- @block
2 -- CUSTOMER_WISHLIST_BOOK
3 BEGIN;
4
5 INSERT INTO public."CUSTOMER_WISHLIST_BOOK" (_customer, _book)
6 VALUES
7 (1, 5),
8 (2, 12),
9 (3, 8),
10 (4, 17),
11 (5, 3),
12 (6, 20),
13 (7, 9),
14 (8, 14),
15 (9, 6),
16 (10, 11),
17 (11, 16),
18 (12, 1),
19 (13, 18),
20 (14, 4),
21 (15, 10),
22 (16, 15),
23 (17, 7),
24 (18, 2),
25 (19, 19),
26 (20, 13);
27
28 END;
```

This is an intermediary relationship between books and the customers who bought the book with more details regarding the price and the quantity of the book.

This is an intermediary relationship between books and the customer to store the information of a wishlist for each customer.

Views

1. Transaction View

```
1 -- @block
2 -- See Transaction Details
3 CREATE VIEW Lihat_Transaksi AS
4 SELECT
5     public."TRANSACTION".timestamp AS "Waktu Transaksi",
6     public."PROFILE".full_name AS "Nama Pelanggan",
7     public."BOOK".name AS "Judul Buku",
8     CASE
9         WHEN
10             public."DISCOUNT".expire >= public."TRANSACTION".timestamp
11             OR
12                 public."DISCOUNT".value IS NULL
13             THEN public."BOOK".main_price
14             ELSE (1 - public."DISCOUNT".value) * public."BOOK".main_price
15         END AS "Nilai Bayar",
16         public."BOOK".main_price AS "Harga Buku",
17     CASE
18         WHEN
19             public."DISCOUNT".expire >= public."TRANSACTION".timestamp
20             OR
21                 public."DISCOUNT".value IS NULL
22             THEN 0
23             ELSE public."DISCOUNT".value * 100
24         END AS "Diskon (%)",
25     CASE
26         WHEN
27             public."DISCOUNT".expire >= public."TRANSACTION".timestamp
28             THEN 'Expired'
29             WHEN
30                 public."DISCOUNT".value IS NULL
31             THEN 'Tidak Pakai'
32             ELSE 'Valid'
33         END AS "Status Diskon"
34
35     FROM public."TRANSACTION"
36     INNER JOIN public."FRANCHISE" ON
37     public."TRANSACTION".franchise = public."FRANCHISE".franchise_key
38     INNER JOIN public."CUSTOMER" ON
39     public."TRANSACTION".customer = public."CUSTOMER".customer_key
40     INNER JOIN public."PROFILE" ON
41     public."PROFILE".profile_key = public."CUSTOMER".profile_key
42     LEFT JOIN public."DISCOUNT" ON
43     public."TRANSACTION".discount = public."DISCOUNT".discount_key
44     INNER JOIN public."BOOK" ON
45     public."TRANSACTION".book = public."BOOK".book_key;
```

Output:

Waktu Transaksi	Nama Pelanggan	Judul Buku	Nilai Bayar	Harga Buku	Diskon (%)	Status Diskon
2024-01-15 08:30:00	Heru Nugroho	Dewi Purnama	60000	60000	0	Expired
2024-02-20 10:45:00	Budi Santoso	Jelajah Sungai	48000	48000	0	Tidak Pakai
2024-03-25 14:20:00	Firman Setiawan	Sembilan Wali	55000	55000	0	Expired
2024-04-10 16:55:00	Darma Wijaya	Perjalanan Bukit	55000	55000	0	Expired
2024-05-05 12:10:00	Rizky Firmansyah	Api Hidup	48000	48000	0	Tidak Pakai
2024-06-15 09:25:00	Maya Rahmawati	Seribu Cerita	50000	50000	0	Tidak Pakai
2024-07-20 11:30:00	Siti Rahayu	Mimpi Sejuta	62000	62000	0	Tidak Pakai
2024-08-28 13:40:00	Dewi Anggraini	Raja Gombal	0	45000	100	Valid
2024-09-05 15:15:00	Adi Nugroho	Hujan di Hatiku	49300	58000	15	Valid
2024-10-10 17:20:00	Lia Suryani	Pulau Pari	15000	60000	75	Valid
2024-11-12 18:45:00	Rudi Hartono	Misteri Gurun	41600	52000	20	Valid
2024-12-17 19:50:00	Putri Utami	Bunga Angin	53000	53000	0	Tidak Pakai
2025-01-22 20:30:00	Sri Wahyuni	Rahasia Hutan	58000	58000	0	Tidak Pakai
2025-02-25 21:55:00	Sri Wijaya	Pahlawan Malam	43500	58000	25	Valid
2025-03-30 22:40:00	Agus Setiawan	Rindu Tanah Air	30250.00000000004	55000	45	Valid
2025-04-05 23:10:00	Yoga Prasetyo	Cinta di Kota	31200	52000	40	Valid
2025-05-10 08:30:00	Eka Pratiwi	Kisah Hidupku	50000	50000	0	Tidak Pakai
2025-06-15 09:45:00	Diana Wulandari	Senja di Pantai	30000	60000	50	Valid
2025-07-20 10:20:00	Hari Santoso	Lautan Kenangan	45500	65000	30	Valid
2025-08-25 11:35:00	Rini Susanti	Cahaya Pohon	50000	50000	0	Tidak Pakai

The view given was made to allow the owner of the bookstore to track the transaction process of every franchise existing across the country.

The owner will be able to track the time of the transaction, know the name of the customers and the books bought.

The most important part of the view is to count the final value of the transaction based on the functional discount being applied to the transaction.

2. Staff View

```
1 -- @block
2 -- See Employee Details
3 CREATE VIEW Lihat_Staff AS
4 SELECT
5     public."PROFILE".full_name AS "Nama",
6     public."POSITION".title AS "Jabatan",
7     public."FRANCHISE".name AS "Franchise",
8     public."EMPLOYEE".recruited_at AS "Tanggal Rekrut",
9     public."POSITION".salary AS "Gaji"
10
11    FROM public."EMPLOYEE"
12    INNER JOIN public."PROFILE" ON
13        public."EMPLOYEE".profile = public."PROFILE".profile_key
14    INNER JOIN public."POSITION" ON
15        public."EMPLOYEE".position = public."POSITION".position_key
16    INNER JOIN public."FRANCHISE" ON
17        public."EMPLOYEE".franchise = public."FRANCHISE".franchise_key;
```

Output:

Nama	Jabatan	Franchise	Tanggal Rekrut	Gaji
Sri Wijaya	Teknisi Perpustakaan	Surga Buku	2021-05-10	5100000
Rudi Hartono	Arsivar	Buku Inspirasi	2020-08-15	5400000
Firman Setiawan	Petugas Perpustakaan	Toko Buku Cerdas	2021-03-25	4800000
Dewi Anggraini	Asisten Perpustakaan	Buku Impian	2020-10-01	5200000
Hari Santoso	Pustakawan Akuisisi	Buku Pintar	2021-07-20	5600000
Eka Pratiwi	Koordinator Pinjam-...	Warung Buku Ceria	2020-11-30	5700000
Heru Nugroho	Spesialis Katalogisasi	Buku Terang	2021-04-05	5800000
Putri Utami	Pustakawan Layanan ...	Buku Ajaib	2020-01-15	5600000
Diana Wulandari	Pustakawan	Buku Ilmiah	2021-09-10	5500000
Darma Wijaya	Pustakawan Referensi	Buku Berkah	2020-02-20	5900000
Adi Nugroho	Pustakawan Anak	Pustaka Bintang	2021-06-30	5400000
Yoga Prasetyo	Teknisi Perpustakaan	Toko Buku Serba Ada	2020-03-12	5100000
Siti Rahayu	Arsivar	Pustaka Jaya	2021-11-05	5400000
Budi Santoso	Petugas Perpustakaan	Pustaka Abadi	2020-04-18	4800000
Lia Suryani	Asisten Perpustakaan	Warung Buku Ceria	2021-08-22	5200000
Agus Setiawan	Pustakawan Akuisisi	Toko Buku Cerdas	2020-12-07	5600000
Sri Wahyuni	Koordinator Pinjam-...	Surga Buku	2021-01-25	5700000
Rizky Firmansyah	Spesialis Katalogisasi	Buku Pintar	2020-05-28	5800000
Rini Susanti	Pustakawan Layanan ...	Buku Impian	2021-10-15	5600000
Maya Rahmawati	Pustakawan Referensi	Pustaka Bintang	2020-07-01	5900000

This view is basically more generalized to allow the owner to know the expense from the HR in each franchise.

As the position and the salary of each staff member is shown here, the owner is allowed to count the monthly expense to sustain the business.