

ANALISIS DATABASE MUSIC STORE (CHINOOK)

“Makalah ini disusun untuk memenuhi Tugas Akhir Mata Kuliah Basis Data”

Dosen Pengampu : Muhaza Liebenlito, M.Si



Universitas Islam Negeri
SYARIF HIDAYATULLAH JAKARTA

Disusun Oleh :

Kelompok 7

Alamsyah Nur M	(11180940000021)
Savira Fika Amanda	(11190940000001)
Sharfinna Zaldy	(11190940000011)
Ruly Erhandi	(11190940000012)
Vina Wulandari R	(11190940000026)
Salsabila Farah Harissaputri	(11190940000029)

PROGRAM STUDI MATEMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH JAKARTA

2022 M/1443 H

KATA PENGANTAR

Assalammu'alaikum Warahmatullahi Wabarakatuh

Puji syukur kami haturkan kepada Allah SWT, karena atas rahmat dan karuniaNya kami dapat menyelesaikan penyusunan Tugas Akhir dengan judul “***Analisis Database Music Store (Chinook)***” dapat terselesaikan dengan baik dan tepat waktu.

Kami menyadari penuh bahwa laporan ini dapat diselesaikan karena adanya dukungan dan bantuan dari berbagai pihak. Untuk itu, pada kesempatan kali ini kami sangat berterima kasih kepada semua yang berperan penting dalam penyelesaian Tugas Besar ini, yang bukan lain adalah :

1. Ibu Dr. Suma'inna, M.Si., selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta.
2. Bapak Muhaza Liebenlito, M.Si selaku Dosen Pengampu dalam Mata Kuliah Basis Data di Program Studi Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
3. Kepada teman seperjuangan Tugas Besar di Mata Kuliah Basis Data yang telah membantu menyelesaikan tugas akhir ini dan merasakan suka-duka nya bersama.
4. Kepada seluruh pihak yang telah membantu kami dalam menyusun serta dalam pembuatan Tugas Besar ini yang tidak dapat kami sebut satu-persatu tetapi tidak mengurangi rasa terima kasih dan rasa hormat kami kepada yang bersangkutan.

Kami menyadari penuh bahwa dalam penyusunan ini masih terdapat banyak kekurangan. Sehingga mengharapkan kritik dan saran yang membangun dari pembaca agar kedepannya mendapatkan hasil yang lebih baik. Terakhir, kami berharap semoga laporan ini dapat bermanfaat dan dapat tercapai sesuai dengan yang diharapkan.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Jakarta, 26 Juni 2022

Penulis

DAFTAR ISI

DAFTAR ISI	3
BAB I	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	2
BAB II	3
METODOLOGI PENELITIAN	3
2.1 Langkah-Langkah Penyelesaian Masalah	3
2.2 Entity Relationship Diagram (ERD)	3
2.3 Data Definition Language (DDL)	5
2.4 Query	13
BAB III	15
HASIL DAN PEMBAHASAN	15
3.1 Visualisasi dan Analisis Data	15
KESIMPULAN	23
4.1 Kesimpulan	23
REFERENSI	24

BAB I

PENDAHULUAN

1.1 Latar Belakang

Basis data (*database*) merupakan suatu kumpulan data informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh suatu informasi dari database tersebut. Dapat pula diartikan sebagai sekumpulan data yang tersusun dalam beberapa tabel sedemikian sehingga saling memiliki sebuah relasi atau koneksi antara tabel satu dengan lainnya maupun berdiri sendiri. Terdapat beberapa model dalam basis data, salah satunya yang sering digunakan saat ini adalah model basis data relasional.

Setiap perusahaan maupun toko-toko biasa memerlukan sebuah basis data. Basis data sangat penting karena selain menyimpan informasi tetapi juga dapat memberikan informasi yang dapat meningkatkan suatu daya saing atau penjualan terhadap perusahaan maupun toko-toko biasa lainnya. Dalam pengelolaan basis data tersebut, memungkinkan penyimpanan data yang lebih mudah diproses dan dapat disimpan jangka panjang, sehingga dapat memudahkan para perusahaan maupun toko-toko biasa dalam menggunakan data-data tersebut serta dapat menghasilkan informasi yang baik. Kehadiran basis data pun dapat mempercepat upaya pelayanan kepada pelanggan.

Pada kasus ini *database* menyimpan suatu informasi mengenai toko musik stars yaitu music store (Chinook). Perkembangan teknologi yang pesat ini sangat memberikan suatu kemajuan dalam penjualan media digital. Hal ini membuat music store harus mengetahui suatu informasi yang penting untuk meningkatkan suatu daya saing penjualan mereka. Oleh karena itu, kami ingin memberikan suatu informasi yang dapat membantu music store (Chinook) dalam memahami penjualan media digital di toko mereka agar para pelanggan mendapatkan kepuasan kepada toko mereka. Selain itu, kami juga memberikan suatu beberapa informasi penting mengenai pelanggan, karyawan dan infrastruktur yang baik untuk music store (Chinook).

Berdasarkan uraian penjelasan diatas, untuk itu kami merancang suatu sistem basis data dengan tema “*Analisis Database Music Store (Chinook)*”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, permasalahan yang akan dibahas adalah sebagai berikut :

1. Bagaimana proses perancangan implementasi basis data dalam musik store (Chinook)?
2. Bagaimana hasil ERD ternormalisasi, DDL, dan query dengan SQL pada basis data dalam musik store (chinook)?

1.3 Tujuan Penelitian

Tujuan penulisan penelitian ini antara lain :

1. Menerapkan sistem basis data dalam mengatasi masalah pada musik store (Chinook)
2. Memberikan solusi yang efektif dan efisien untuk masalah mengimplementasi basis data dalam musik store (Chinook)

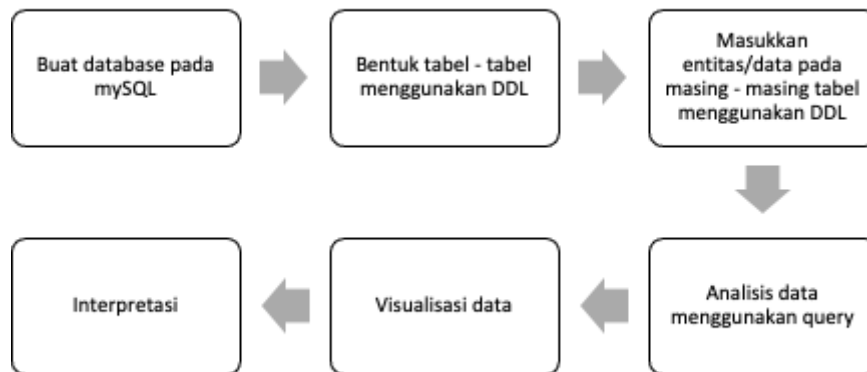
1.4 Manfaat Penelitian

Manfaat penelitian ini adalah untuk memberikan informasi dalam memahami penjualan media digital pada music store (chinook) dengan menggunakan software SQL.

BAB II

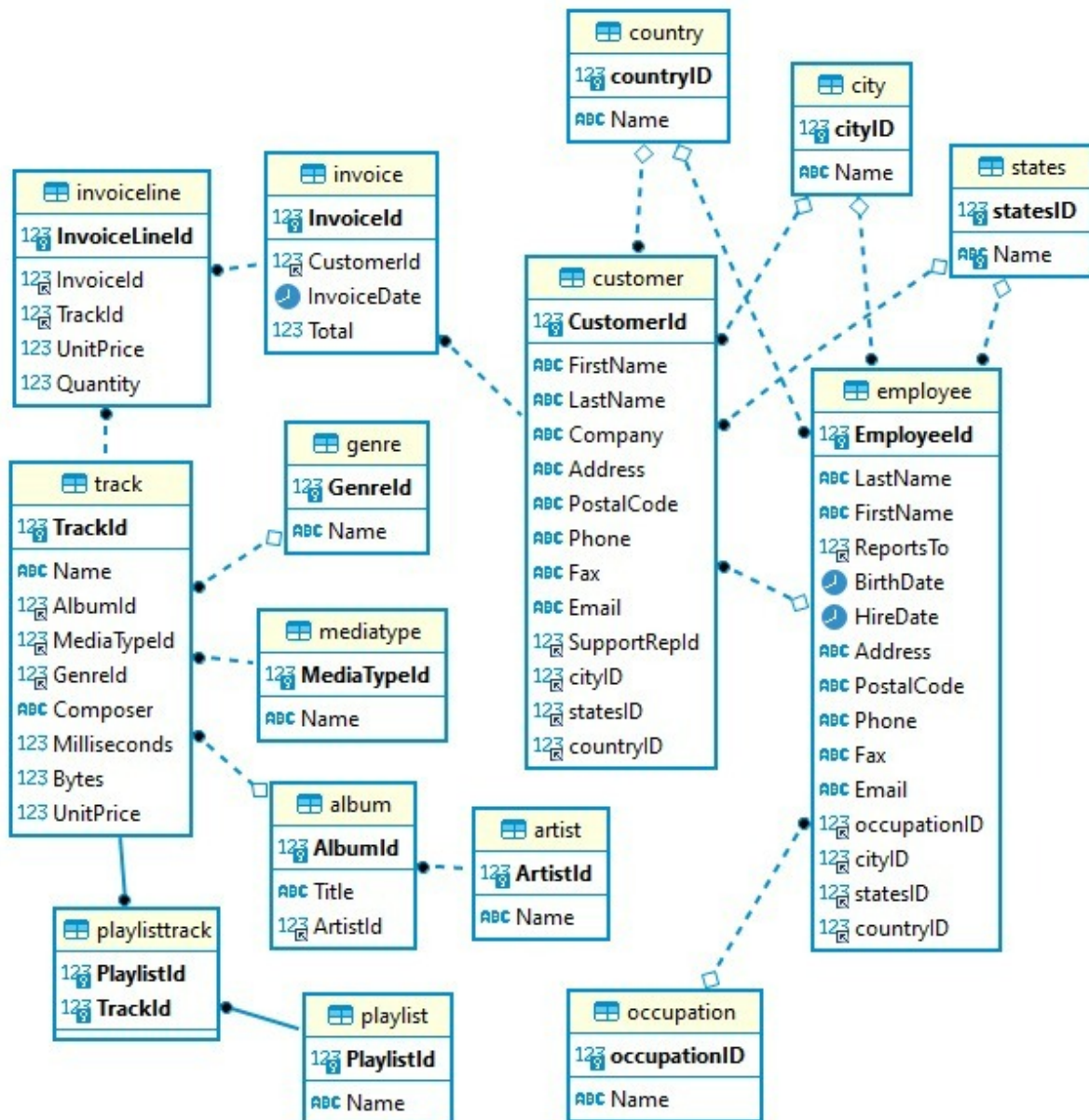
METODOLOGI PENELITIAN

2.1 Langkah-Langkah Penyelesaian Masalah



2.2 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan model yang digunakan untuk membuat database dengan hasil output yang rapi dan terstruktur, output yang dimaksud yaitu berbentuk tabel dan memiliki hubungan dengan basis data yang akan dipakai nantinya [2].



ERD diatas merupakan ERD dari *database Chinook*, dimana terdiri atas 14 tabel yang telah dinormalisasi. Tabel *customer* merupakan *main table* yang menghubungkan banyak tabel lainnya. Tabel *customer* memiliki *primary key* yaitu *CustomerId*, dan memiliki *foreign key* dari tabel *country*, *city*, *state*, *employee*, dan *occupation*. Tabel *Customer* menjadi tabel yang menghubungkan tabel-tabel yang berisi data penjualan, yaitu tabel *invoice*, dimana tabel ini merupakan hubungan dari tabel *invoiceLine*. Tabel *invoiceLine* terhubung pada tabel *track* yang memiliki *primary key* *TrackId*, tabel ini menghubungkan ke beberapa tabel lainnya seperti *genre*, *mediatype*, *album*, *artist*, *playlist*, dan *playlisttrack*.

2.3 Data Definition Language (DDL)

Data Definition Language (DDL) merupakan sebuah kelompok yang memerintahkan untuk mendefinisikan atribut-atribut pada basis data, atribut (kolom), batasan-batasan terhadap suatu atribut, serta hubungan antar tabel. Berikut ini perintah-perintah yang terdapat dalam kelompok Data Definition Language (DDL):

1. CREATE TABLE

Membuat tabel basis data baru yang berisi tabel untuk menyimpan suatu entitas yang terdiri dari atribut (kolom) dan record (basis data). Tipe data yang digunakan untuk mendefinisikan tipe dari atribut (kolom) di tabel. Beberapa tipe data yang sering digunakan sebagai berikut:

Tipe Data	Keterangan
INT	Menyimpan nilai integer
FLOAT	Menyimpan nilai float
VARCHAR	Menyimpan nilai string
CHAR	Menyimpan nilai satu karakter
DATE	Menyimpan nilai waktu
TEXT	Menyimpan nilai teks

Terdapat pula primary key yang digunakan sebagai identifier unik pada setiap record dan tidak boleh bernilai NULL.

2. ALTER TABLE

Mengubah tabel basis data atau struktur tabel seperti menambah, merubah/modifikasi dan menghapus kolom. Perintah ini sangat berharga saat mendefinisi ulang tabel, namun perubahan struktur tabel yang telah atau sedang digunakan operasional penuh dengan resiko bencana. *View* yang diturunkan dari tabel tersebut dapat tidak berfungsi dengan baik atau bahkan tidak berjalan sama sekali. Sehingga mengharuskan rancangan basis data ini menjadi pilihan terakhir yang dapat digunakan [1].

3. DROP TABLE

Digunakan untuk menghapus tabel atau basis data.

Pada kasus ini berikut query Data Definition Language (DDL) -nya:

```
DROP TABLE IF EXISTS `album`;  
/*!40101 SET @saved_cs_client      = @@character_set_client */;  
/*!40101 SET character_set_client = utf8 */;  
CREATE TABLE `album` (  
  `AlbumId` int NOT NULL,
```



```

    `Title` varchar(160) CHARACTER SET utf8 COLLATE utf8_general_ci
NOT NULL,
    `ArtistId` int NOT NULL,
    PRIMARY KEY (`AlbumId`),
    KEY `IFK_AlbumArtistId` (`ArtistId`),
    CONSTRAINT `FK_AlbumArtistId` FOREIGN KEY (`ArtistId`) REFERENCES
`artist` (`ArtistId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `artist`
--

DROP TABLE IF EXISTS `artist`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `artist` (
    `ArtistId` int NOT NULL,
    `Name` varchar(120) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
    PRIMARY KEY (`ArtistId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `city`
--

DROP TABLE IF EXISTS `city`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `city` (
    `cityID` int NOT NULL AUTO_INCREMENT,
    `Name` varchar(100) DEFAULT NULL,
    PRIMARY KEY (`cityID`)
) ENGINE=InnoDB AUTO_INCREMENT=67 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `country`
--

```

```

DROP TABLE IF EXISTS `country`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `country` (
  `countryID` int NOT NULL AUTO_INCREMENT,
  `Name` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`countryID`)
) ENGINE=InnoDB AUTO_INCREMENT=32 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `customer`
--

DROP TABLE IF EXISTS `customer`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `customer` (
  `CustomerId` int NOT NULL,
  `FirstName` varchar(40) CHARACTER SET utf8 COLLATE utf8_general_ci
NOT NULL,
  `LastName` varchar(20) CHARACTER SET utf8 COLLATE utf8_general_ci
NOT NULL,
  `Company` varchar(80) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Address` varchar(70) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `PostalCode` varchar(10) CHARACTER SET utf8 COLLATE
utf8_general_ci DEFAULT NULL,
  `Phone` varchar(24) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Fax` varchar(24) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Email` varchar(60) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
NULL,
  `SupportRepId` int DEFAULT NULL,
  `cityID` int DEFAULT NULL,
  `statesID` int DEFAULT NULL,
  `countryID` int DEFAULT NULL,
  PRIMARY KEY (`CustomerId`),
  KEY `IFK_CustomerSupportRepId` (`SupportRepId`),

```

```

KEY `customer_FK` (`cityID`),
KEY `customer_FK_1` (`countryID`),
KEY `customer_FK_2` (`statesID`),
  CONSTRAINT `customer_FK` FOREIGN KEY (`cityID`) REFERENCES `city`
(`cityID`),
  CONSTRAINT `customer_FK_1` FOREIGN KEY (`countryID`) REFERENCES
`country` (`countryID`),
  CONSTRAINT `customer_FK_2` FOREIGN KEY (`statesID`) REFERENCES
`states` (`statesID`),
  CONSTRAINT `FK_CustomerSupportRepId` FOREIGN KEY (`SupportRepId`)
REFERENCES `employee` (`EmployeeId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `employee`
--

DROP TABLE IF EXISTS `employee`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `employee` (
  `EmployeeId` int NOT NULL,
  `LastName` varchar(20) CHARACTER SET utf8 COLLATE utf8_general_ci
NOT NULL,
  `FirstName` varchar(20) CHARACTER SET utf8 COLLATE utf8_general_ci
NOT NULL,
  `ReportsTo` int DEFAULT NULL,
  `BirthDate` datetime DEFAULT NULL,
  `HireDate` datetime DEFAULT NULL,
  `Address` varchar(70) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `PostalCode` varchar(10) CHARACTER SET utf8 COLLATE
utf8_general_ci DEFAULT NULL,
  `Phone` varchar(24) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Fax` varchar(24) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Email` varchar(60) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `occupationID` int DEFAULT NULL,
  `cityID` int DEFAULT NULL,
  `statesID` int DEFAULT NULL,

```

```

    `countryID` int DEFAULT NULL,
    PRIMARY KEY (`EmployeeId`),
    KEY `IFK_EmployeeReportsTo` (`ReportsTo`),
    KEY `employee_FK_1` (`cityID`),
    KEY `employee_FK_2` (`countryID`),
    KEY `employee_FK_3` (`occupationID`),
    KEY `employee_FK_4` (`statesID`),
    CONSTRAINT `employee_FK_1` FOREIGN KEY (`cityID`) REFERENCES
`city` (`cityID`),
    CONSTRAINT `employee_FK_2` FOREIGN KEY (`countryID`) REFERENCES
`country` (`countryID`),
    CONSTRAINT `employee_FK_3` FOREIGN KEY (`occupationID`) REFERENCES
`occupation` (`occupationID`),
    CONSTRAINT `employee_FK_4` FOREIGN KEY (`statesID`) REFERENCES
`states` (`statesID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `genre`
--

DROP TABLE IF EXISTS `genre`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `genre` (
  `GenreId` int NOT NULL,
  `Name` varchar(120) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  PRIMARY KEY (`GenreId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `invoice`
--

DROP TABLE IF EXISTS `invoice`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `invoice` (
  `InvoiceId` int NOT NULL,
  `CustomerId` int NOT NULL,

```

```

    `InvoiceDate` datetime NOT NULL,
    `Total` decimal(10,2) NOT NULL,
    PRIMARY KEY (`InvoiceId`),
    KEY `IFK_InvoiceCustomerId` (`CustomerId`),
    CONSTRAINT `FK_InvoiceCustomerId` FOREIGN KEY (`CustomerId`)
REFERENCES `customer` (`CustomerId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `invoiceline`
--

DROP TABLE IF EXISTS `invoiceline`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `invoiceline` (
    `InvoiceLineId` int NOT NULL,
    `InvoiceId` int NOT NULL,
    `TrackId` int NOT NULL,
    `UnitPrice` decimal(10,2) NOT NULL,
    `Quantity` int NOT NULL,
    PRIMARY KEY (`InvoiceLineId`),
    KEY `IFK_InvoiceLineInvoiceId` (`InvoiceId`),
    KEY `IFK_InvoiceLineTrackId` (`TrackId`),
    CONSTRAINT `FK_InvoiceLineInvoiceId` FOREIGN KEY (`InvoiceId`)
REFERENCES `invoice` (`InvoiceId`),
    CONSTRAINT `FK_InvoiceLineTrackId` FOREIGN KEY (`TrackId`)
REFERENCES `track` (`TrackId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `mediatype`
--

DROP TABLE IF EXISTS `mediatype`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `mediatype` (
    `MediaTypeId` int NOT NULL,
    `Name` varchar(120) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,

```

```

        PRIMARY KEY (`MediaTypeId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `occupation`
--

DROP TABLE IF EXISTS `occupation`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `occupation` (
  `occupationID` int NOT NULL AUTO_INCREMENT,
  `Name` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`occupationID`)
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `playlist`
--

DROP TABLE IF EXISTS `playlist`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `playlist` (
  `PlaylistId` int NOT NULL,
  `Name` varchar(120) CHARACTER SET utf8 COLLATE utf8_general_ci
  DEFAULT NULL,
  PRIMARY KEY (`PlaylistId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `playlisttrack`
--

DROP TABLE IF EXISTS `playlisttrack`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `playlisttrack` (
  `PlaylistId` int NOT NULL,

```

```

    `TrackId` int NOT NULL,
    PRIMARY KEY (`PlaylistId`, `TrackId`),
    KEY `IFK_PlaylistTrackTrackId` (`TrackId`),
    CONSTRAINT `FK_PlaylistTrackPlaylistId` FOREIGN KEY (`PlaylistId`)
REFERENCES `playlist` (`PlaylistId`),
    CONSTRAINT `FK_PlaylistTrackTrackId` FOREIGN KEY (`TrackId`)
REFERENCES `track` (`TrackId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `states`
--

DROP TABLE IF EXISTS `states`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `states` (
  `statesID` int NOT NULL AUTO_INCREMENT,
  `Name` varchar(100) NOT NULL,
  PRIMARY KEY (`statesID`),
  UNIQUE KEY `states_un` (`Name`)
) ENGINE=InnoDB AUTO_INCREMENT=33 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `track`
--

DROP TABLE IF EXISTS `track`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `track` (
  `TrackId` int NOT NULL,
  `Name` varchar(200) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
NULL,
  `AlbumId` int DEFAULT NULL,
  `MediaTypeId` int NOT NULL,
  `GenreId` int DEFAULT NULL,
  `Composer` varchar(220) CHARACTER SET utf8 COLLATE utf8_general_ci
DEFAULT NULL,
  `Milliseconds` int NOT NULL,

```

```

`Bytes` int DEFAULT NULL,
`UnitPrice` decimal(10,2) NOT NULL,
PRIMARY KEY (`TrackId`),
KEY `IFK_TrackAlbumId` (`AlbumId`),
KEY `IFK_TrackGenreId` (`GenreId`),
KEY `IFK_TrackMediaTypeId` (`MediaTypeId`),
CONSTRAINT `FK_TrackAlbumId` FOREIGN KEY (`AlbumId`) REFERENCES
`album` (`AlbumId`),
CONSTRAINT `FK_TrackGenreId` FOREIGN KEY (`GenreId`) REFERENCES
`genre` (`GenreId`),
CONSTRAINT `FK_TrackMediaTypeId` FOREIGN KEY (`MediaTypeId`)
REFERENCES `mediatype` (`MediaTypeId`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

2.4 Query

- a. Analisis *Playlist* Terbanyak

```

-- Playlist Terbanyak
select p2.Name, count(p.TrackId) as Track from playlisttrack p
join playlist p2 on p.PlaylistId = p2.PlaylistId
group by p2.Name
order by Track desc

```

- b. Analisis Pegawai dengan Jabatan *Sales Support Agent*

```

-- Employee dengan jabatan Sales Support Agent
select e.FirstName, e.LastName, o.Name
from employee e
join occupation o on e.occupationID = o.occupationID
where o.Name = "Sales Support Agent"
-- done

```

- c. Analisis 3 *States* dengan Penjualan Terbanyak

```

-- top 3 states
select c.statesID, s.Name,
count(i.InvoiceId) as total invoice per state
from states s
join customer c on s.statesID = c.statesID
join invoice i on c.CustomerId = i.CustomerId
group by statesID
order by total_invoice_per_state desc
-- done

```

- d. Analisis 5 Negara Teratas dengan Pembelian Album Terbanyak


```
-- top 5 countries
select c.countryID, c2.Name,
count(i.InvoiceId) as total_invoice_per_country
from country c2
join customer c on c2.countryID = c.countryID
join invoice i on c.CustomerId = i.CustomerId
group by countryID
order by total_invoice_per_country desc
```

- e. Analisis *Artist* dengan Album Terbanyak

```
-- Artist yang memiliki album terbanyak
select a2.Name, count(a.AlbumId) as Jumlah_Album from album a
join artist a2 on a.ArtistId = a2.ArtistId
group by a2.ArtistId
order by Jumlah_Album desc
```

- f. Analisis *Artist* dengan Penjualan Album Terbanyak

```
-- artist dengan penjualan album terbanyak
select a3.Name, count(i2.InvoiceId) as Invoice_Total
from artist a
join album a2 on a.ArtistId = a2.ArtistId
join track t on a2.AlbumId = t.AlbumId
join invoiceline i on t.TrackId = i.TrackId
join invoice i2 on i.InvoiceId = i2.InvoiceId
join artist a3 on a2.ArtistId = a3.ArtistId
group by a2.ArtistId
order by Invoice_Total desc
```

- g. Analisis Genre Musik yang Paling diminati Berdasarkan Banyaknya Penjualan

```
-- Genre terbanyak terjual
select g.Name, t.GenreId, count(i.InvoiceId) as total_invoice
from genre g
join track t on g.GenreId = t.GenreId
join invoiceline i2 on t.TrackId = i2.TrackId
join invoice i on i2.InvoiceId = i.InvoiceId
group by GenreId
order by total_invoice desc
```

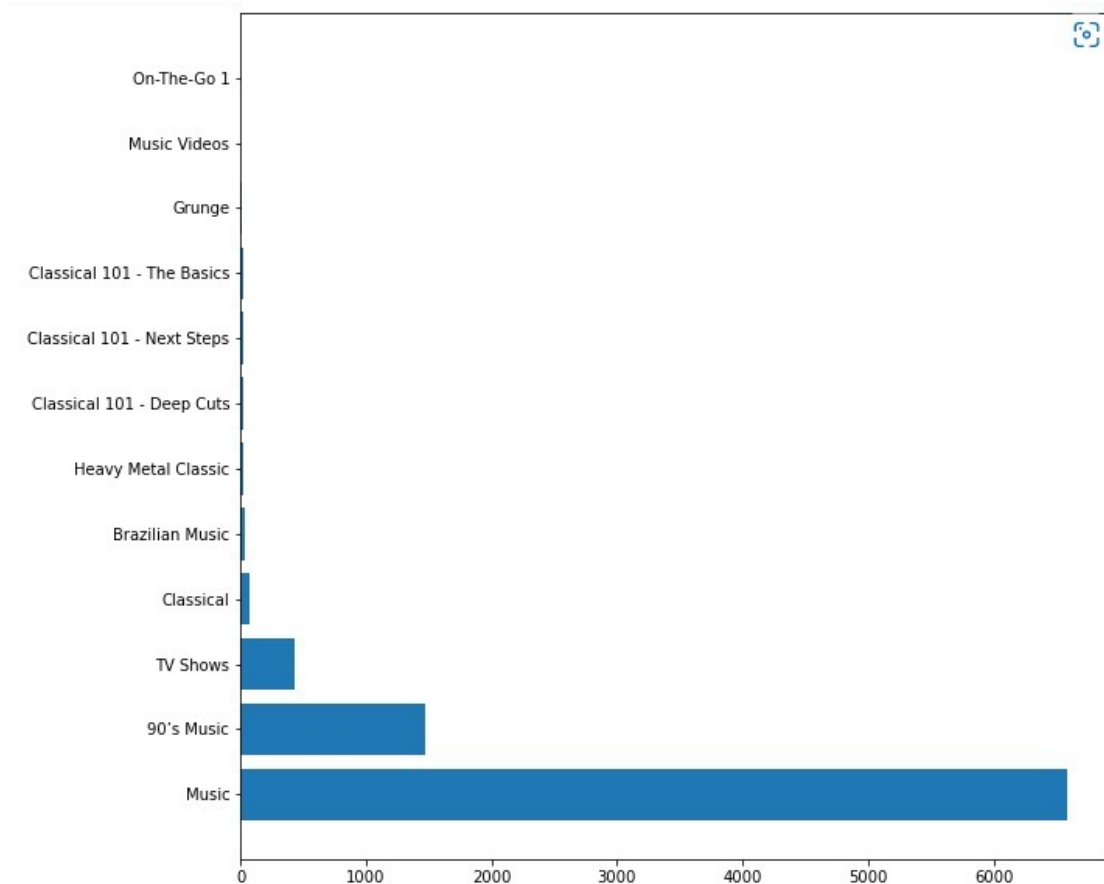
BAB III

HASIL DAN PEMBAHASAN

3.1 Visualisasi dan Analisis Data

a. Playlist Terbanyak

	Name	Track
0	Music	6580
1	90's Music	1477
2	TV Shows	426
3	Classical	75
4	Brazilian Music	39
5	Heavy Metal Classic	26
6	Classical 101 - Deep Cuts	25
7	Classical 101 - Next Steps	25
8	Classical 101 - The Basics	25
9	Grunge	15
10	Music Videos	1
11	On-The-Go 1	1



Interpretasi : Berdasarkan hasil analisis dan visualisasi diagram diatas, dapat diketahui bahwa *playlist* yang sering diputar yaitu *Music* dengan *track* 6580, yang kemudian diikuti oleh *90's Music*, dan *TV shows*. Maka, dapat disimpulkan bahwa *Music* banyak digemari oleh masyarakat.

b. Analisis Pegawai dengan Jabatan Sales Agent

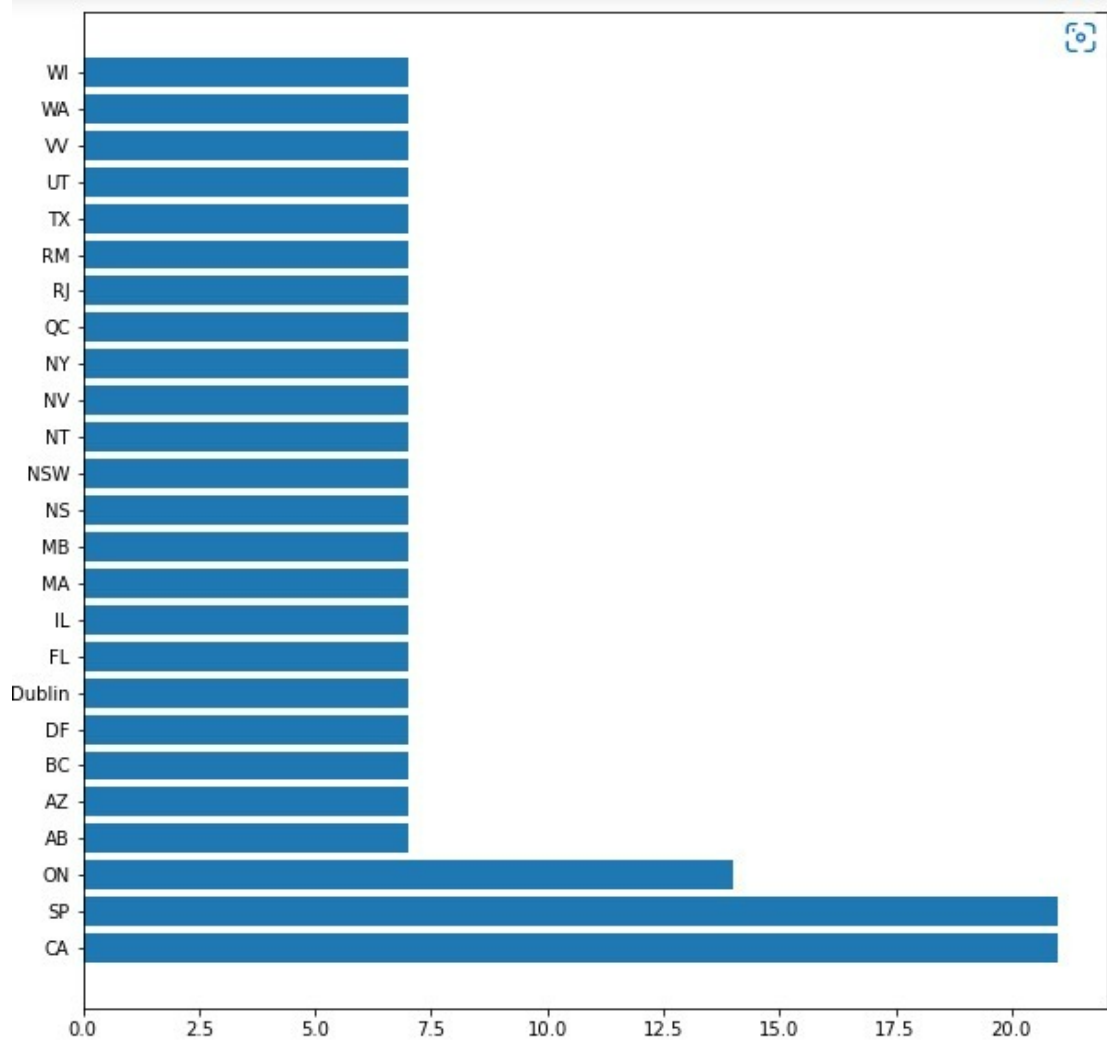
	FirstName	LastName	Name
0	Jane	Peacock	Sales Support Agent
1	Margaret	Park	Sales Support Agent
2	Steve	Johnson	Sales Support Agent

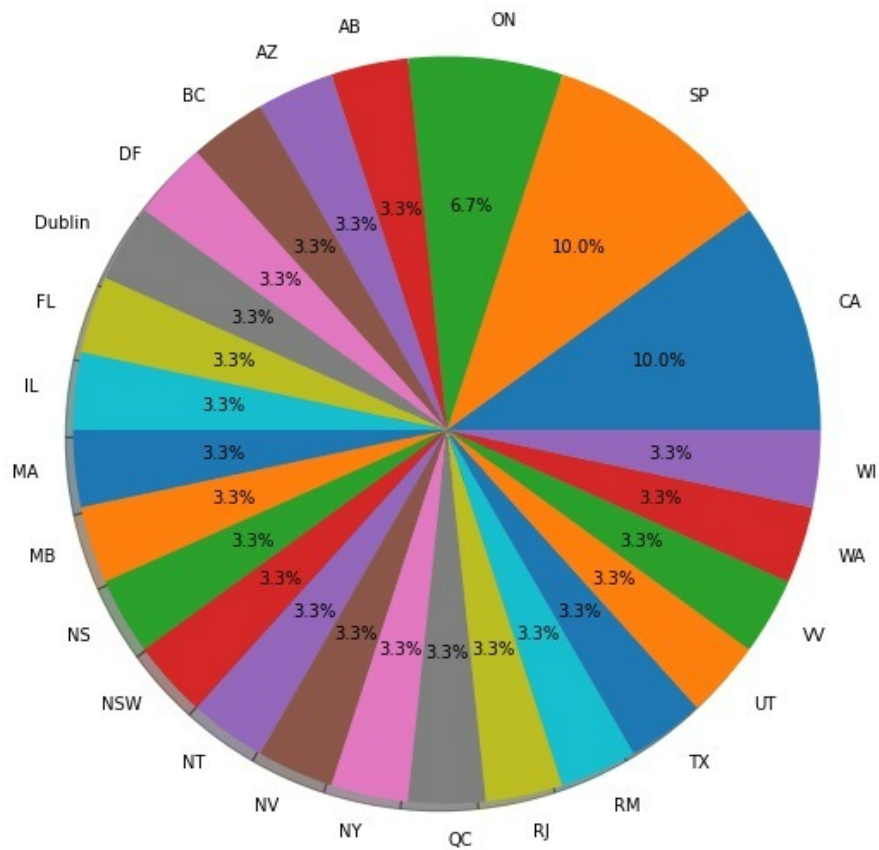
Interpretasi :

Pegawai dengan jabatan Sales Support Agent ada 3 orang, yaitu Jane Peacock, Margaret Park, Steve Johnson. Sedangkan untuk pegawai lainnya memiliki jabatan yang masing-masing berbeda.

c. Analisis *States* yang Melakukan Pembelian Terbanyak

	statesID	Name	total_invoice_per_state
0	7	CA	21
1	1	SP	21
2	18	ON	14

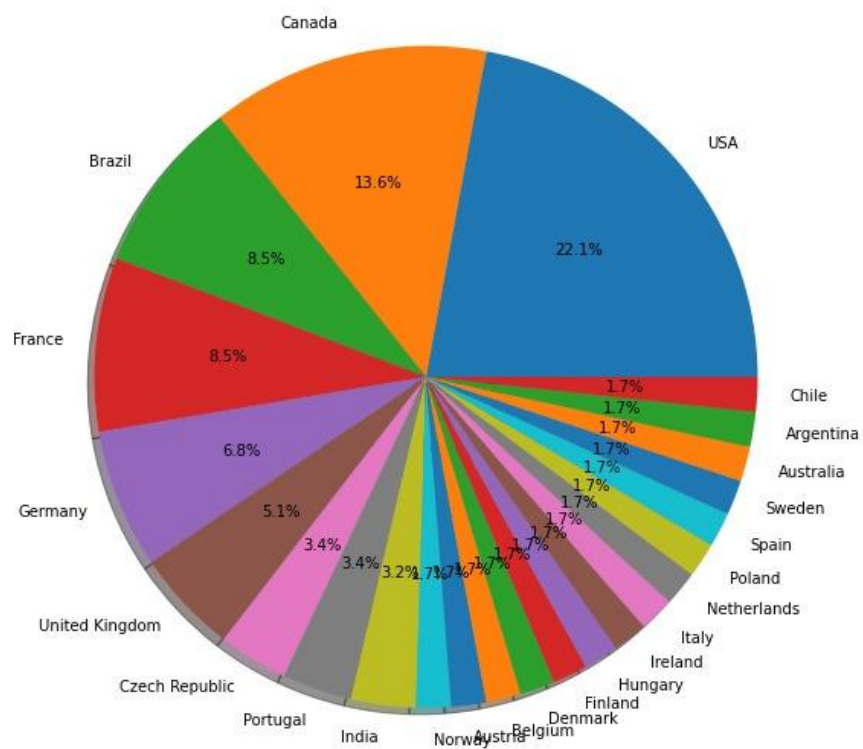
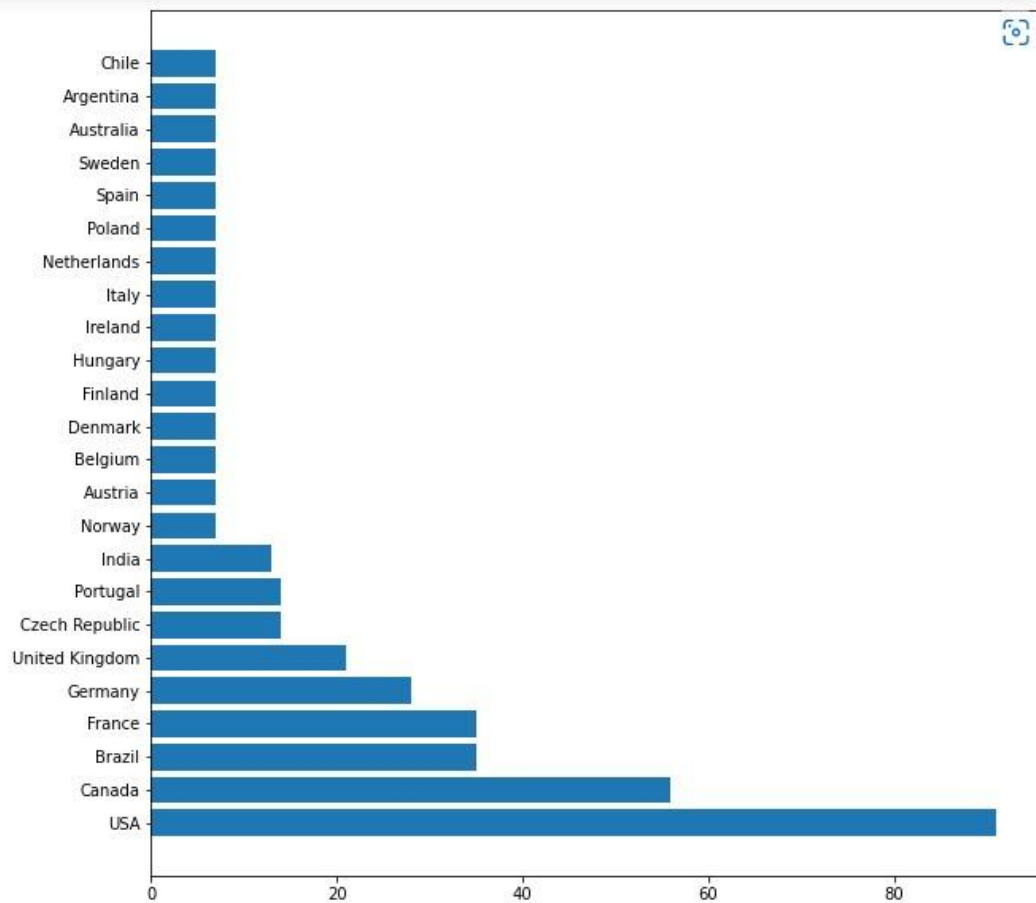




Interpretasi : Berdasarkan visualisasi diatas, dari 25 states, 3 states yang paling banyak melakukan pembelian album yaitu CA (California) sebesar 10%, SP (Sao Paulo) sebesar 10%, dan ON (Ontario) sebesar 6.7% dari keseluruhan *states*.

d. Analisis 5 Negara Teratas dalam Pembelian Album

	countryID	Name	total_invoice_per_country
0	9	USA	91
1	3	Canada	56
2	1	Brazil	35
3	11	France	35
4	2	Germany	28

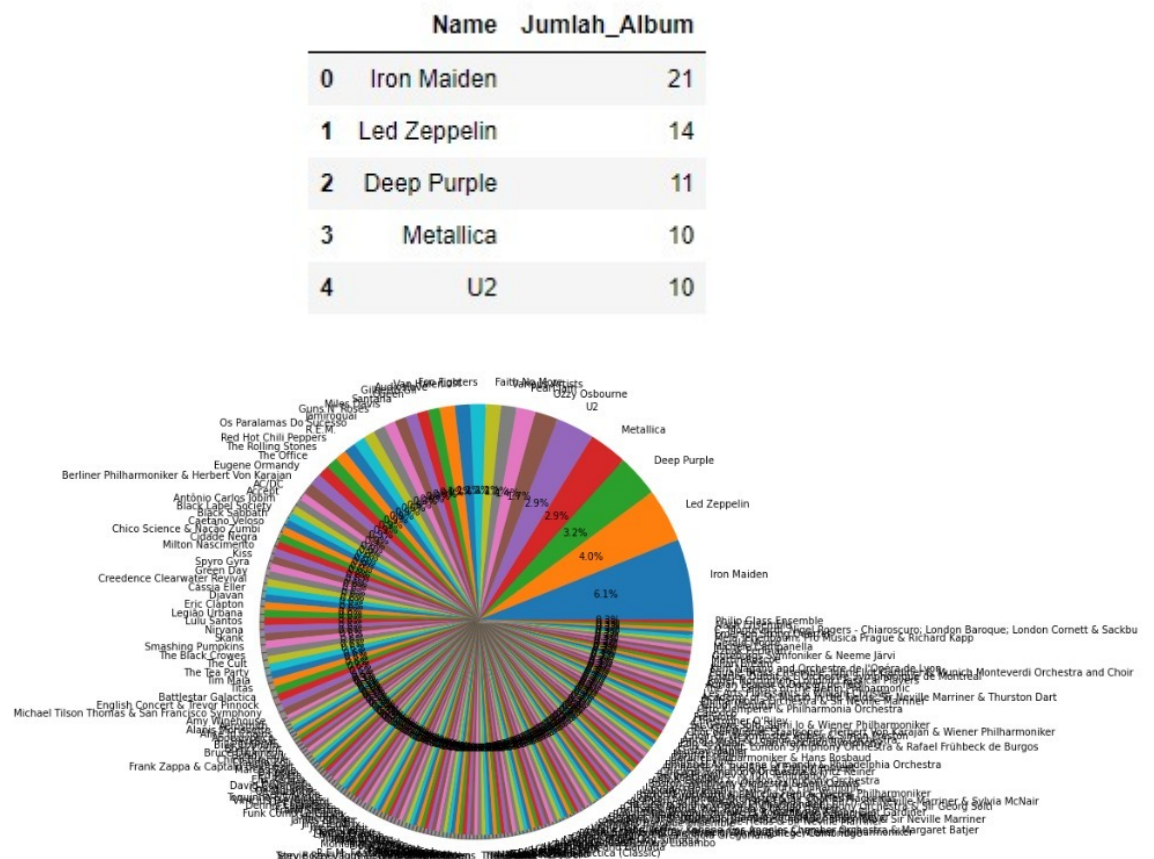


Interpretasi : Berdasarkan hasil analisis dan visualisasi diatas, 5 negara dengan pembelian album terbanyak yaitu :

1. USA sebesar 22.1% atau sekitar 91,
2. Canada sebesar 13.6% atau sekitar 56,
3. Brazil sebesar 8.5% atau sekitar 35,
4. France sebesar 8.5% atau sekitar 35, dan
5. Germany sebesar 6.8% atau sekitar 28

Untuk negara-negara lainnya berada pada rentang persentase antara 1%-5%.

e. Analisis *Artist* dengan Album Terbanyak



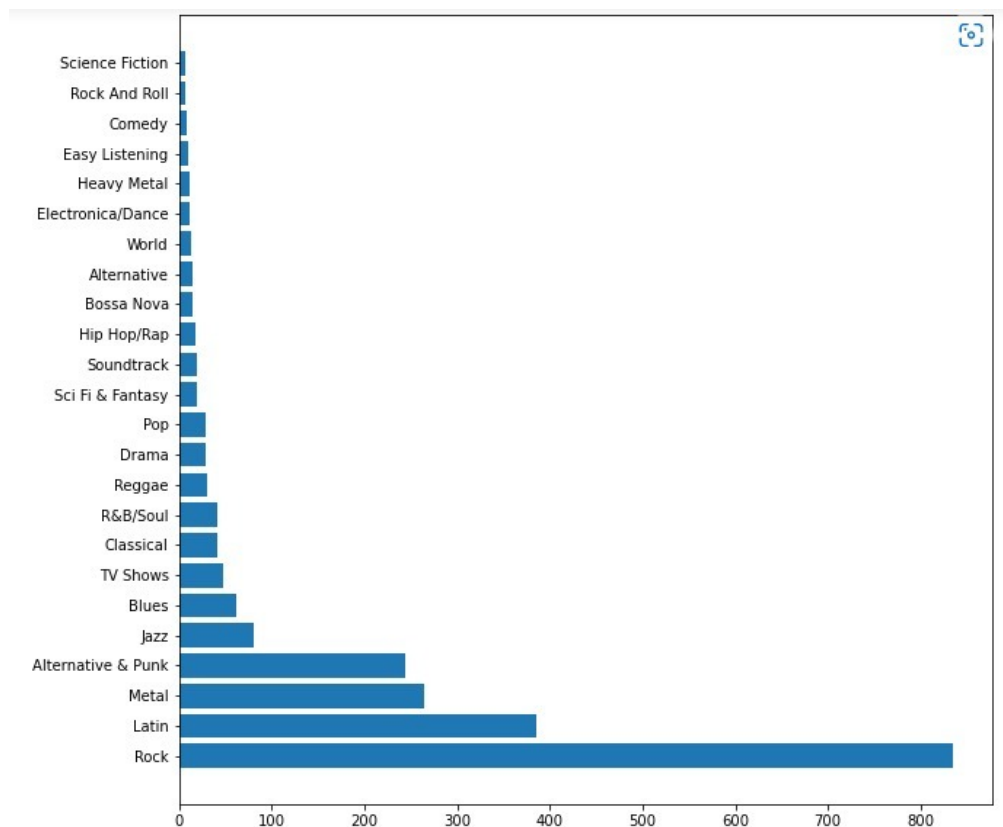
Interpretasi : Berdasarkan hasil analisis di atas, 5 *artist* dengan total album yang dikeluarkan terbanyak adalah Iron Maiden dengan total 21 album, Led Zeppelin dengan 14 album, Deep Purple dengan 11 album, Metallica dan U2 dengan masing-masing 10 album.

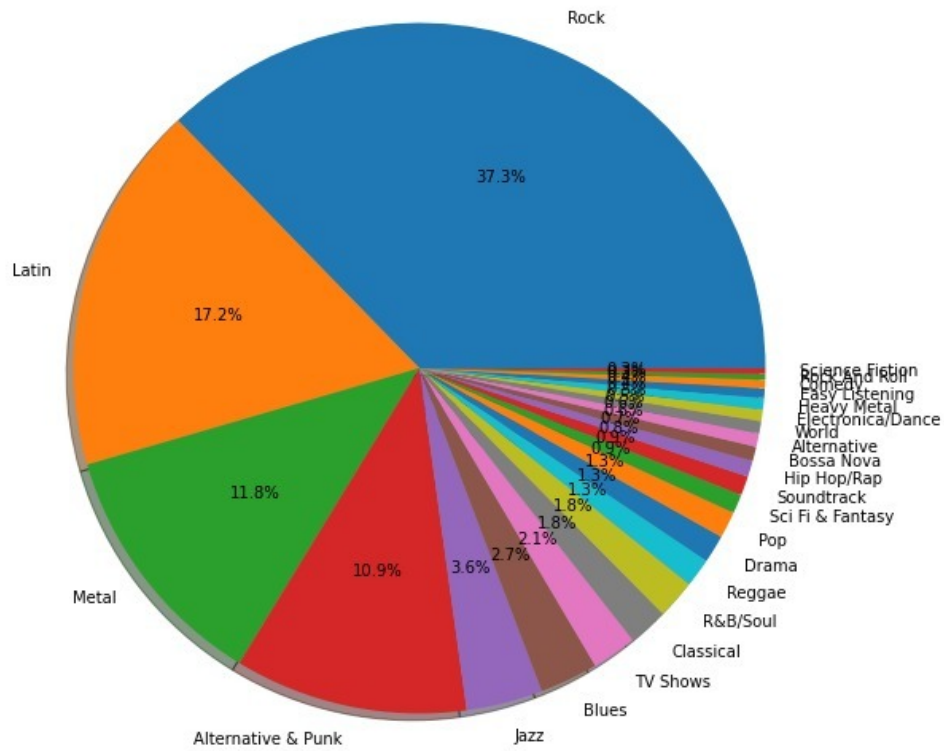
f. Analisis *Artist* dengan Penjualan Album Terbanyak

	Name	Invoice_Total
0	Iron Maiden	140
1	U2	107
2	Metallica	91
3	Led Zeppelin	87
4	Os Paralamas Do Sucesso	45

Interpretasi : Berdasarkan analisis data diatas, dari keseluruhan *artists* dapat diketahui terdapat 5 *artists* teratas dengan penjualan album terbanyak yaitu Iron Maiden dengan total 140 album, U2 dengan 107 album, Metallica dengan penjualan sebesar 91 album, Led Zeppelin dengan penjualan 87 album, dan Os Paralamas Do Sucesso dengan penjualan 45 album.

g. Analisis Genre Musik yang Paling diminati Berdasarkan Banyaknya Penjualan





Interpretasi : Berdasarkan hasil analisis data diatas dapat diketahui bahwa dari seluruh genre musik dan berbagai album yang ada, satu genre yang memiliki penjualan tertinggi dan peminatan yang tinggi adalah genre *Rock*. Hal ini dapat dilihat pada *bar chart* diatas dimana ditunjukkan penjualan musik dengan genre *Rock* berada pada angka di atas 800 atau sekitar 37.3%.

BAB IV

KESIMPULAN

4.1 Kesimpulan

Kesimpulan yang diperoleh dari hasil dan analisis diatas adalah bahwa mayoritas masyarakat pada *database* memiliki kecenderungan untuk memilih mendengarkan musik dibandingkan dengan *playlist* lainnya. Pada *digital store* yang dianalisis, terdapat banyak pegawai yang mengurus segala hal jual beli, untuk bagian *sales support agent* terdiri atas tiga orang, yaitu Jane Peacock, Margaret Park, Steve Johnson. Sedangkan untuk pegawai lainnya memiliki jabatan yang masing-masing berbeda.

Penjualan yang tersebar di berbagai negara membuat *database* ini dapat dianalisis lebih jauh dengan mencari tahu *states* yang melakukan pembelian terbanyak yaitu California sebesar 10%, Sao Paulo sebesar 10%, dan Ontario sebesar 67% dari keseluruhan *states*. Sedangkan untuk negara dengan pembelian album terbanyak, yaitu USA sebesar 22.1% atau sekitar 91, Canada sebesar 13.6% atau sekitar 56, Brazil sebesar 8.5% atau sekitar 35, France sebesar 8.5% atau sekitar 35, dan Germany sebesar 6.8% atau sekitar 28.

Selain hal tersebut, dari *database* ini juga dapat dianalisis lima *artist* dengan penerbitan album terbanyak yaitu Iron Maiden dengan total 21 album, Led Zeppelin dengan 14 album, Deep Purple dengan 11 album, Metallica dan U2 dengan masing-masing 10 album.

Namun, untuk *artist* dengan penjualan album terbanyak sedikit berbeda, yaitu Iron Maiden dengan total 140 album, U2 dengan 107 album, Metallica dengan penjualan sebesar 91 album, Led Zeppelin dengan penjualan 87 album, dan Os Paralamas Do Sucesso dengan penjualan 45 album. Untuk genre yang paling banyak diminati oleh masyarakat pada *database* yang dianalisis yaitu *genre Rock* dengan persentase 37.3% dari keseluruhan data.

REFERENSI

- [1] "Entity Relationship Diagram: Pengertian, Fungsi, dan Cara Membuatnya." *Accurate Online*, 14 October 2021,
<https://accurate.id/marketing-manajemen/entity-relationship-diagram/>. Accessed 27 June 2022.
- [2] "Sistem manajemen basisdata / Bambang Hariyanto | Perpustakaan UIN Sultan Syarif Kasim Riau." *Portal Aplikasi Inlis Lite*,
<http://inlislite.uin-suska.ac.id/opac/detail-opac?id=14641>. Accessed 27 June 2022.