

--This work has been done with active collaboration from all members so there was no reason to slip the work into parts

```
CREATE TABLE ARTIST(  
    Artistic_Name VARCHAR(20)  
    constraint PK_C_ArtistName primary key clustered,  
    First_Name VARCHAR(32) not null,  
    Second_Name VARCHAR(32) not null,  
    Nationality VARCHAR(32) not null,  
    Year_Birthday INT,  
    Age INT DEFAULT 18  
    CONSTRAINT CK_ARTIST_YEARSOFEXPAG CHECK(Age >= 0),  
    Years_Of_Experience INT DEFAULT 0  
    CONSTRAINT CK_ARTIST_YEARSOFEXPY CHECK(Years_Of_Experience >= 0),  
    Debut_Date DATE,  
    Total_Sales INT DEFAULT 0  
    constraint CHECK_ARTISTSALLES CHECK(Total_Sales >= 0),  
    Royalties INT DEFAULT 0  
    constraint CHECK_ARTISTROYAL CHECK(Royalties >= 0),  
    Awards VARCHAR(32)  
    CONSTRAINT CK_ARTIST_YEARSOFEXPA CHECK(Awards >= 0) ,  
    Is_Alive BIT DEFAULT 'SI'  
    CONSTRAINT CK_ARTIST_ISALIVE CHECK(Is_Alive LIKE '[SI, NO]') ,  
  
    )
```

```
CREATE TABLE COUPONS(  
    Code VARCHAR(32)  
    constraint CHECK_COUPONSC CHECK(Code LIKE '[0-9][0-9][0-9][0-9][0-9]  
[a-z,A-Z]') )  
    constraint PK_C_Code primary key clustered,  
    Expiration DATE,  
    Discount INT not null,  
    Coupon_Format VARCHAR(32),  
    Start_Date DATE not null  
    )
```

```
CREATE TABLE PAYMENT(  
    Payment_ID INT  
    constraint PK_C_PaymentID primary key clustered,  
    Users_full_name VARCHAR(32),  
    "Year" INT,  
    "Month" INT,  
    "Day" INT,  
    "Hour" INT  
    constraint UQ_C_Multiple unique  
nonclustered(Users_full_name,"Year","Month","Day","Hour")  
    )
```

```
CREATE TABLE PURCHASE_TRACKS(  
    Cart_ID INT  
    constraint PK_C_CartID primary key clustered,  
    Initial_Price INT not null  
    constraint CHECK_PURCHASE_TRACKSI CHECK(Initial_Price >= 0) ,  
    Number_Tracks INT not null  
    constraint CHECK_PURCHASE_TRACKSN CHECK(Number_Tracks >= 0) ,  
    Final_Price INT not null  
    constraint CHECK_PURCHASE_TRACKSF CHECK(Final_Price >= 0) ,  
    Is_Bought_By_Ads BIT,  
    Coupon_Code VARCHAR(32)  
    constraint FK_C_CouponCode foreign key references COUPONS(Code),  
    Payment_ID INT not null
```

```
constraint FK_C_PaymentID foreign key references PAYMENT(Payment_ID)
)
```

```
CREATE TABLE "USER"(
    Username VARCHAR(20)
    constraint PK_C_Username primary key clustered,
    First_Name VARCHAR(32),
    Second_Name VARCHAR(32),
    Profile_Photo IMAGE not null,
    City VARCHAR(32),
    Street VARCHAR (32),
    ZIP INT,
    Favourite_Track VARCHAR(32),
    Cart_ID INT
    constraint FK_C_FTrack foreign key references
PURCHASE_TRACKS(Cart_ID),
    Favourite_Artist VARCHAR(32),
    Favourite_Gentre VARCHAR(32),
    Number_Of_Purchase INT not null
    constraint CHECK_USERNUM CHECK(Number_Of_Purchase >= 0) ,
    Money_Balance INT not null
)
```

```
CREATE TABLE RECORD_LABEL(
    CIF VARCHAR(32)
    constraint PK_C_CIF primary key clustered,
    Brand_Name VARCHAR(32)
    constraint UQ_NC_Brand_Name unique nonclustered,
    Number_Of_Employees INT DEFAULT 0
    constraint C_RECORD_LABELN CHECK(Number_Of_Employees >= 0),
    Stock_Market_Value INT
)
```

```
CREATE TABLE TRACKS(
    Reference_Number INT
    constraint PK_C_ReferenceNumber primary key clustered,
    BPM INT
    constraint CK_TRACKSBPM CHECK(BPM>=0),
)
```

```
CREATE TABLE USER_AUDIO_PREFERENCES(
    Profile_Preferences VARCHAR(200),
    UserName VARCHAR(20)
    constraint FK_C_UserName foreign key references "USER"(Username)
    constraint PK_C_PPreferences primary key
clustered(Profile_Preferences,UserName)
)
```

```
CREATE TABLE USER_TRACK(
    Username VARCHAR(20)
    constraint FK_C_User_Name foreign key references "USER"(Username),
    Track_Reference_N INT
    constraint FK_C_TrackReference foreign key references
TRACKS(Reference_Number)
    constraint PK_C_Mult primary key clustered(Username
,Track_Reference_N )
)
```

```

CREATE TABLE BILLING_INFORMATION(
    Type_Account VARCHAR(32),
    Number_Account INT,
    User_UserName VARCHAR(20)
    constraint FK_C_User_UserName foreign key references
"USER"(Username)
    constraint PK_C_Multiple primary key
clustered(Type_Account,Number_Account,User_UserName)
)

CREATE TABLE USER_EMAIL(
    Email_Account VARCHAR(32),
    User_UserName VARCHAR(20)
    constraint FK_C_UserUserName foreign key references "USER"(Username)
    constraint PK_C_Mltpl primary key
clustered(Email_Account,User_UserName)
)

CREATE TABLE PLAYLIST(
    Playlist_ID INT
    constraint PK_C_Playlist primary key clustered,
    Name VARCHAR(32),
    User_UserName VARCHAR(20)
    constraint FK_C_UserUser_Name foreign key references
"USER"(Username)
    constraint UQ_NC_Playlist unique nonclustered(Name,User_UserName)
)

CREATE TABLE LABELS_WORKED_WITH(
    Artistic_Name VARCHAR(20)
    constraint FK_C_ArtisticName foreign key references
ARTIST(Artistic_Name),
    Labels_Name VARCHAR(32)
    constraint PK_C_Labels primary key
clustered(Artistic_Name,Labels_Name)
)

CREATE TABLE ArtRec(
    CIFRecordLabel VARCHAR(32)
    constraint FK_C_CIFRecordLabel foreign key references
RECORD_LABEL(CIF),
    Artistic_Name VARCHAR(20)
    constraint FK_C_Artistic_Name foreign key references
ARTIST(Artistic_Name)
    constraint PK_C_Art primary key
clustered(CIFRecordLabel,Artistic_Name)
)

CREATE TABLE PLAYTRACK(
    TRACK_N_REFERENCE INT
    constraint FK_C_TrackNumberReference foreign key references
TRACKS(Reference_Number),
    Playlist_ID INT
    constraint FK_C_PlaylistID foreign key references
PLAYLIST(Playlist_ID),
    constraint PK_C_PlayTrack primary key
clustered(TRACK_N_REFERENCE,Playlist_ID)
)

```

```

CREATE TABLE PAYROLL(
    ID INT
    constraint PK_C_ID primary key clustered,
    Bonus INT DEFAULT 0
)

CREATE TABLE EMPLOYEES(
    SSN INT
    constraint PK_C_SSN primary key clustered,
    Working_Years_Fullfiled INT DEFAULT 0,
    Hiring_Date DATE,
    Phone INT,
    Allowance INT NOT NULL,
    Civil_Status VARCHAR(40) DEFAULT 'SINGLE',
    Blood_Group CHAR(2),
    Payroll_ID INT not null constraint FK_EMPLOYEES_Payroll_ID foreign
key references PAYROLL(ID),
    Category VARCHAR(30) NOT NULL)

CREATE TABLE DEVELOPERS(
    SSN INT
    constraint PK_C_DEVSSN primary key clustered
    constraint FK_C_DEVSSN foreign key
    references EMPLOYEES(SSN),
    Programming_Language VARCHAR(40))
CREATE TABLE EMPADRESS(
    SSN INT
    constraint PK_C_EMPSSN primary key clustered
    constraint FK_C_EMPSSN foreign key
    references EMPLOYEES(SSN),
    City VARCHAR(60),
    Street VARCHAR(60))

CREATE TABLE MANAGERS(
    SSN INT
    constraint PK_C_MANSSN primary key clustered
    constraint FK_C_MANSSN foreign key
    references EMPLOYEES(SSN))
CREATE TABLE ADMINS(
    SSN INT
    constraint PK_C_ADMSSN primary key clustered
    constraint FK_C_ADMSSN foreign key
    references EMPLOYEES(SSN))
CREATE TABLE ALBUM(
    Number_Of_Reference INT
    constraint PK_C_ALBUMNUMBEROFREF primary key clustered,
    Genre VARCHAR(40) not null,
    Available_In_Physical_Format BIT DEFAULT 'NO'
    CONSTRAINT CK_AvailablePhysical CHECK(Available_In_Physical_Format
LIKE '[SI, NO]'),
    "Year" int
    CONSTRAINT CK_Year CHECK("Year" >= 0),
    "Month" int
    CONSTRAINT CK_Month CHECK("Month" >= 0)
)

CREATE TABLE ALBUM_AUTHORS(
    Number_Of_Reference INT
    constraint FK_C_ALBUMAUTHORSNUMOFREF foreign key references
    ALBUM(Number_Of_Reference),
    First_Name VARCHAR(100),
    Second_Name VARCHAR(130),
    constraint PK_C_ALBUMAUTHORS primary key(Number_Of_Reference,

```

First_Name, Second_Name))

```
CREATE TABLE COUPONSNAME(  
    Code VARCHAR(32)  
    constraint FK_C_COUPON_CODE foreign key references COUPONS(Code),  
    Name VARCHAR(80),  
    constraint PK_C_COUPONSNAME primary key clustered(Code,  
        Name))
```

```
CREATE TABLE DISTRIBUTOR(  
    CIF VARCHAR(60)  
    constraint PK_C_DISTRIBUTOR primary key clustered,  
    Companys_Name VARCHAR(80)  
    constraint UK_C_DISTRIBUTOR unique nonclustered,  
    Central_Officies VARCHAR(80),  
    Nationality VARCHAR(30) NOT NULL)
```

```
CREATE TABLE RecordDistr(  
    CIFDistributor VARCHAR(60)  
    constraint FK_C_CIFDistributor foreign key references  
DISTRIBUTOR(CIF),  
    CIFRecordLabel VARCHAR(32)  
    constraint FK_C_CIF_RecordLabel foreign key references  
RECORD_LABEL(CIF)  
    constraint PK_C_RecordDistr primary key  
clustered(CIFDistributor,CIFRecordLabel)  
)
```

```
CREATE TABLE DistrAdmin(  
    CIFDistributor VARCHAR(60)  
    constraint FK_C_CIF_Distributors foreign key references  
DISTRIBUTOR(CIF),  
    SSNAdmin INT  
    constraint FK_C_SSNAdmin foreign key references ADMINS(SSN)  
    constraint PK_C_DistrAdmin primary key  
clustered(CIFDistributor,SSNAdmin)  
)
```

```
CREATE TABLE ALBUMTRACKART(  
    ArtisticName VARCHAR(20)  
    constraint FK_U_ALBUMTRACKARTAR foreign key references  
ARTIST(Artistic_Name),  
    Album_Number_Of_Reference INT  
    constraint FK_U_ALBUMTRACKARTAL foreign key references  
ALBUM(Number_Of_Reference),  
    Track_Number_Of_Reference INT  
    constraint FK_U_ALBUMTRACKARTTR foreign key references  
TRACKS(Reference_Number),  
    constraint PK_C_ALBUMTRACKART primary key clustered  
(Track_Number_Of_Reference,Album_Number_Of_Reference,ArtisticName))
```

```
CREATE TABLE MANAGDEV(  
    Personal_Relation VARCHAR(120),  
    SSNDev INT  
    constraint FK_U_MANAGDEV foreign key references DEVELOPERS(SSN),  
    SSNMan INT  
    constraint FK_U_MANAGDEVM foreign key references MANAGERS(SSN),  
    constraint PK_C_MANAGDDEV primary key clustered (SSNMan,
```

```
SSNDev))
```

```
CREATE TABLE MANAGADMIN(  
    Personal_Relation VARCHAR(120),  
    SSNMan INT  
    constraint FK_U_MANAGADMINM foreign key references MANAGERS(SSN),  
    SSNAdm INT  
    constraint FK_U_MANAGADMINA foreign key references ADMINS(SSN),  
    constraint PK_C_MANAGADEV primary key clustered (SSNMan,  
        SSNAdm))
```

```
GO
```

```
CREATE TRIGGER TR_ARTIST ON ARTIST  
FOR INSERT  
AS  
INSERT ARTIST(Year_Birthday) VALUES(year(getDate())) - (select Age from  
ARTIST))  
GO  
GO
```

```
CREATE TRIGGER TR_PURCHASE_TRACKS on PURCHASE_TRACKS  
FOR INSERT  
AS  
  
INSERT PURCHASE_TRACKS(Final_Price) VALUES((SELECT Initial_Price FROM  
PURCHASE_TRACKS) - (SELECT Discount FROM COUPONS))  
GO
```