**Vyšší odborná škola a Střední průmyslová škola elektrotechnická,  
Plzeň, Koterovská 85**

***Databáze*  
Semestrální práce  
Návrh a realizace hudební databáze**

**Školní rok 2024/2025 Daniel Hajžman, IF2**

1. Téma práce

Tématem práce je návrh a implementace pro správu hudebních dat, zahrnující informace o interpretech, albech, vydavatelích a skladbách.

1. Cíl projektu

Cílem projektu je vytvořit databázi, která umožní efektivně spravovat a dotazovat se na informace o hudebních interpretech, jejich albech, skladbách a vydavatelích. Této databáze lze využít v systémech pro správu hudebních knihoven nebo streamingové platformy.

1. Návrh databáze
   1. Konceptuální model

V konceptuálním modelu jsou identifikovány hlavní entity:

* Interpret (Artist)
* Album
* Vydavatel (Publisher)
* Skladba (Song)
  1. Logický model

Entity a jejich vztahy jsou definovány následujícím způsobem:

* Interpret může mít více alb
* Album obsahuje skladby
* Album může mít jednoho vydavatele
* Každá skladba náleží jednomu albu
  1. Fyzický model

Fyzický model je implementován ve formátu tabulek a jejich vztahů v databázi.

1. Implementace databáze
   1. Vytvoření schématu

USE IF2\_hajzmand;  
GO

CREATE SCHEMA sp;  
GO

* 1. Vytvoření tabulek

Tabulky byly vytvořeny následujícími příkazy:

**Tabulka tbArtist**

CREATE TABLE sp.tbArtist(  
 ArtistID INT PRIMARY KEY IDENTITY(1,1),  
 [Name] VARCHAR(50) NOT NULL,  
 Country VARCHAR(50) NOT NULL,  
 Founded DATETIME NOT NULL CHECK(Founded < GETDATE()),  
 Tags VARCHAR(50),  
 [Description] TEXT  
);

Důvodem vytvoření tabulky tbArtist je uložení klíčových informací o interpretech.

**Tabulka tbAlbum**

CREATE TABLE sp.tbAlbum(  
 AlbumID INT PRIMARY KEY IDENTITY(1,1),  
 Title VARCHAR(100) NOT NULL,  
 Genre VARCHAR(50) NOT NULL,  
 Duration VARCHAR(50),  
 Price MONEY,  
 [Description] TEXT,  
 TrackAmount INT,  
 ISRC VARCHAR(20) NOT NULL UNIQUE  
);

Tabulka tbAlbum uchovává informace o albech, včetně žánru a ceny.

**Tabulka tbArtistAlbum**

CREATE TABLE sp.tbArtistAlbum(  
 ArtistID INT NOT NULL FOREIGN KEY REFERENCES sp.tbArtist(ArtistID),  
 AlbumID INT NOT NULL FOREIGN KEY REFERENCES sp.tbAlbum(AlbumID)  
);

Spojovací tabulka tbArtistAlbum definuje vztah mezi interprety a jejich alby.

**Tabulka tbPublisher**

CREATE TABLE sp.tbPublisher(  
 PublisherID INT PRIMARY KEY IDENTITY(1,1),  
 AlbumID INT NOT NULL FOREIGN KEY REFERENCES sp.tbAlbum(AlbumID),  
 [Name] VARCHAR(50) NOT NULL,  
 [Location] VARCHAR(50)  
);

Tabulka tbPublisher obsahuje informace o vydavatelích alb.

**Tabulka tbSong**

CREATE TABLE sp.tbSong(  
 SongID INT PRIMARY KEY IDENTITY(1,1),  
 AlbumID INT FOREIGN KEY REFERENCES sp.tbAlbum(AlbumID),  
 [Title] VARCHAR(100) NOT NULL,  
 Duration VARCHAR(50)  
);

Tabulka tbSong ukládá informace o jednotlivých skladbách.

* 1. Indexy

Pro optimalizaci dotazů byly vytvořeny indexy:

CREATE INDEX ixtbArtist ON sp.tbArtist(Country);  
CREATE INDEX ixtbAlbum ON sp.tbAlbum(Genre);

Konkrétně index země interpreta a index žánru alba.

* 1. Přidání primárního klíče do tbArtistAlbum

Jelikož nemůžeme přidat 2 primární klíče to jedné tabulky, musíme je nastavit na klíče cizí a poté tabulku upravit přidáním „constraintu“, který nastaví již zmíněné cizí klíče na klíče primární.

ALTER TABLE tbArtistAlbum ADD CONSTRAINT pk\_tbArtistAlbum PRIMARY KEY(ArtistID, AlbumID);

* 1. Naplnění tabulek daty

Pro naplnění tabulek byly použity následující příkazy:

INSERT INTO sp.tbArtist([Name], Country, Founded, Tags, [Description])  
VALUES  
('The Rolling Stones', 'UK', '1962-07-12', 'Rock, Blues', 'Legendary British rock band formed  
in London.'),  
('Daft Punk', 'France', '1993-01-01', 'Electronic, House', 'French electronic music duo known  
for their innovative sound.'),

...

1. Závěr

Tato databáze poskytuje komplexní strukturu pro správu hudebních dat. Je navržena tak, aby umožnila efektivní dotazování na interprety, alba, vydavatele a skladby. Sktruktura je snadno rozšiřitelná o další entity, jakou jsou žánry nebo playlisty či EP nebo kompilace.