

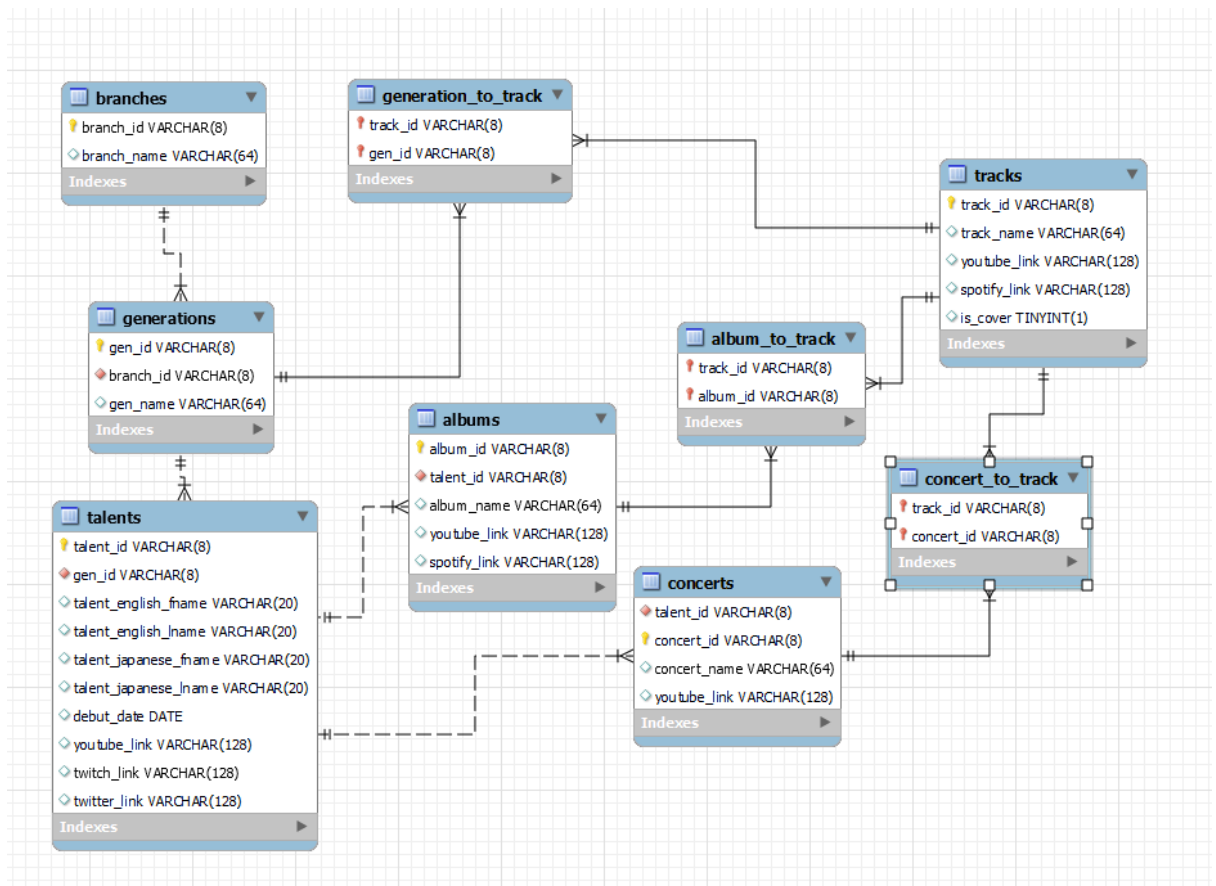
Database Project

Faculty no. —————

1. Introduction

The idea of this project is to create database for a talent agency. It should include branches of agency, generations of talents, talents themselves, their albums and their songs.

2. Database structure



EER Diagram view

3. Creation of database

Following SQL query was used to create database:

```
drop database if exists holobase;  
create database holobase;  
use holobase;
```

```
create table branches (  
    branch_id VARCHAR(8) NOT NULL,  
    branch_name VARCHAR(64),
```

```
    primary key (branch_id)  
);
```

```
create table generations (  
    gen_id VARCHAR(8) NOT NULL,  
    branch_id VARCHAR (8) NOT NULL,  
    gen_name VARCHAR(64),
```

```
    primary key (gen_id),  
    foreign key (branch_id) references branches(branch_id)  
on delete restrict on update cascade  
);
```

```
create table talents (  
    talent_id VARCHAR(8) NOT NULL,  
    gen_id VARCHAR(8) NOT NULL,  
    talent_english_fname VARCHAR(20),  
    talent_english_lname VARCHAR(20),  
    talent_japanese_name VARCHAR(20),  
    debut_date DATE,  
    youtube_link VARCHAR(128),
```

```
twitter_link VARCHAR(128),  
  
    primary key (talent_id),  
    foreign key (gen_id) references generations(gen_id) on  
delete restrict on update cascade  
);
```

```
create table albums (  
    album_id VARCHAR(8) NOT NULL,  
    talent_id VARCHAR(8) NOT NULL,  
    album_name VARCHAR(64),  
    youtube_link VARCHAR(128),  
  
    primary key (album_id),  
    foreign key (talent_id) references talents(talent_id) on  
delete restrict on update cascade  
);
```

```
create table tracks (  
    track_id VARCHAR(8) NOT NULL,  
    track_name VARCHAR(64),  
    youtube_link VARCHAR(128),  
    is_cover BOOLEAN,  
  
    primary key (track_id)  
);
```

```
create table concerts (  
    talent_id VARCHAR(8) NOT NULL,  
    concert_id VARCHAR(8) NOT NULL,  
    concert_name VARCHAR(64),  
    youtube_link VARCHAR(128),
```

```
    primary key (concert_id),
    foreign key (talent_id) references talents(talent_id) on
delete restrict on update cascade
);
```

```
create table generation_to_track (
    track_id VARCHAR(8) NOT NULL,
    gen_id VARCHAR(8) NOT NULL,
```

```
    primary key (track_id, gen_id),
    foreign key (track_id) references tracks(track_id) on
delete restrict on update cascade,
    foreign key (gen_id) references generations(gen_id) on
delete restrict on update cascade
);
```

```
create table album_to_track (
    track_id VARCHAR(8) NOT NULL,
    album_id VARCHAR(8) NOT NULL,
```

```
    primary key (track_id, album_id),
    foreign key (track_id) references tracks(track_id) on
delete restrict on update cascade,
    foreign key (album_id) references albums(album_id) on
delete restrict on update cascade
);
```

4. Populating database

Since there`s no centralized resource with agencies data, the population of database was done by hand using open source resources. Whole SQL script with data can be

found on <https://github.com/ThNeutral/holobase> in database-data.sql file.

5. Queries

1. Get generation song for group “ReGLOSS”:

```
select tracks.track_name
from generations, generation_to_track, tracks
where generations.gen_id = generation_to_track.gen_id
and generation_to_track.track_id = tracks.track_id and
generations.gen_name = "ReGLOSS";
```

track_name
瞬間ハートビート

2. Get a debut date of all talents in the holoMyth group:

```
select talents.talent_english_fname,
talents.talent_english_lname, talents.debut_date
from talents, generations
where talents.gen_id = generations.gen_id and
generations.gen_name = "holoMyth";
```

talent_english_fname	talent_english_lname	debut_date
Kiara	Takanashi	2020-09-12
Amelia	Watson	2020-09-13

3. Get all track names of all songs in “Specter” album:

```
select albums.album_name, tracks.track_name
from albums, album_to_track, tracks
where albums.album_id = album_to_track.album_id and
album_to_track.track_id = tracks.track_id and
albums.album_name = "Specter";
```

album_name	track_name
Specter	7days
Specter	デビューントボール

4. Get all talent names in “HololiveJP” branch:

```

select talents.talent_english_fname,
talents.talent_english_lname,
talents.talent_japanese_name
from branches, generations, talents
where branches.branch_id = generations.branch_id and
generations.gen_id = talents.gen_id and
branches.branch_name = "HololiveJP";

```

	talent_english_fname	talent_english_lname	talent_japanese_name
▶	Suisei	Hoshimachi	星街すいせい
	AZKi	NULL	NULL
	Sora	Tokino	ときのそら
	Botan	Shishiro	獅白ぼたん
	Polka	Omaru	尾丸ポルカ

5. Get all concerts of talents in “HololiveJP” branch:

```

select branches.branch_name,
talents.talent_english_fname,
talents.talent_english_lname, concerts.concert_name
from branches, generations, talents, concerts
where branches.branch_id = generations.branch_id and
generations.gen_id = talents.gen_id and concerts.talent_id
= talents.talent_id and branches.branch_name =
"HololiveJP";

```

	branch_name	talent_english_fname	talent_english_lname	concert_name
▶	HololiveJP	Suisei	Hoshimachi	SheenderelaDay
	HololiveJP	AZKi	NULL	ゲスト:ときのそら / 星街すいせい
	HololiveJP	Sora	Tokino	想いをこめて届ける3D配信
	HololiveJP	Botan	Shishiro	ミッションをクリアせよ
	HololiveJP	Polka	Omaru	ポルカおるよ

6. Closing thoughts

By doing this project, I found out on practice how to design, create and use databases. Creating database schema and searching information in web to fill in DB honed my SQL and data search skills. Furthermore, topic I chose is interesting to me, so I enjoyed not only writing

database itself but also looking for information for project and had great motivation to finish it.