OUTLINE

Why this database. Let's say you're interested in classical music. Perhaps you heard a piece by JS Bach you liked and want to discover other pieces he has written. Perhaps you are looking for piano concertos from the romantic period. Maybe you want to find every piece Schubert wrote for trumpet or search for all the composers from Russia. Here's a database that allows you to add such data and build searches for such things.

DATABASE OUTLINE IN WORDS

In the classical music world there are a number of relationships between composers, countries composers are from, pieces of music, instruments, and classical period.

Each composer has an id, a first name, and a last name. Composers are from a country. Each country has an id and a name. Composers belong to a classical period. Each period has an id and a name.

There are pieces of music. Each piece of music has an id, a name, and a type. Pieces of music are composed by a composer. Composers have written one or more pieces of music. Pieces of music contain instruments. They can have one or more instruments.

Instruments have an id and a name. The same instrument can be in many different pieces.

ER DIAGRAM

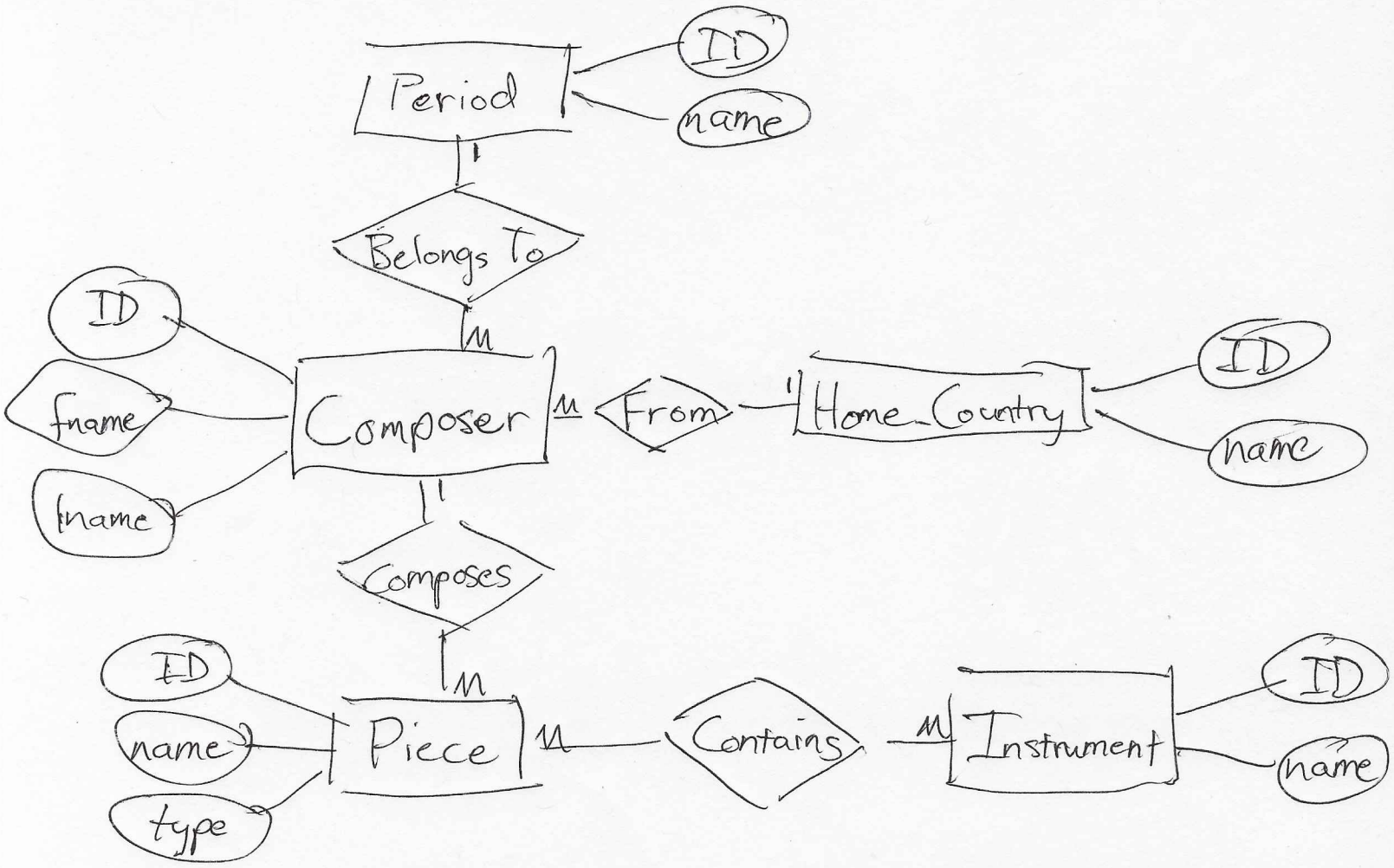
See next Pages

DATABASE SCHEMA

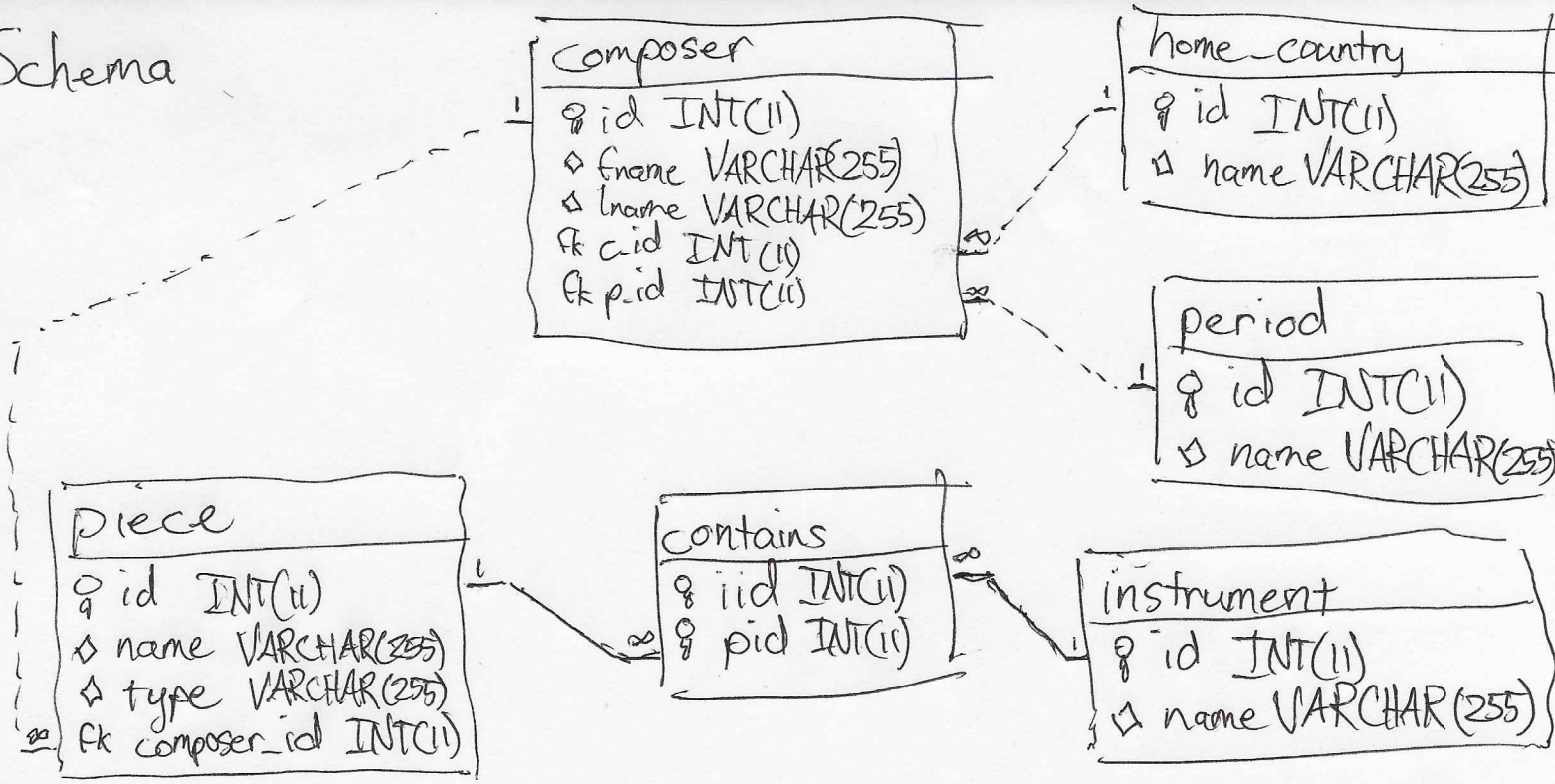
See next Pages

TABLE CREATION QUERIES

```
-- REICHMAT -- Final ASSIGNMENT -- Definition -- 3/18/17

DROP TABLE IF EXISTS `home_country`;
DROP TABLE IF EXISTS `composer`;
DROP TABLE IF EXISTS `period`;
DROP TABLE IF EXISTS `instrument`;
DROP TABLE IF EXISTS `piece`;
DROP TABLE IF EXISTS `contains`;
DROP TABLE IF EXISTS `composes_for`;

-- Create the tables

CREATE TABLE home_country
(
	id INT AUTO_INCREMENT PRIMARY KEY,
	name VARCHAR(255) NOT NULL,
	CONSTRAINT unique_name UNIQUE (name)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE period
(
	id INT AUTO_INCREMENT PRIMARY KEY,
	name VARCHAR(255) NOT NULL,
	CONSTRAINT unique_name UNIQUE (name)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE composer
(
	id INT AUTO_INCREMENT PRIMARY KEY,
	c_id INT,
	p_id INT,
	fname VARCHAR(255) NOT NULL,
	lname VARCHAR(255) NOT NULL,
	FOREIGN KEY (c_id) REFERENCES home_country(id),
	FOREIGN KEY (p_id) REFERENCES period(id)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;


CREATE TABLE instrument
(
	id INT AUTO_INCREMENT PRIMARY KEY,
	name VARCHAR(255) NOT NULL,
	CONSTRAINT unique_name UNIQUE (name)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE piece
(
```


Period
ID
name
Belongs To
1
M
ID
fname
lname
Composer
M
From
1
Home_Country
ID
name
1
M
Composes
ID
name
type
Piece
M
Contains
M
Instrument
ID
name

Schema


composer
id INT(11)
fname VARCHAR(255)
lname VARCHAR(255)
fk c_id INT(11)
fk p_id INT(11)
home_country
id INT(11)
name VARCHAR(255)
period
id INT(11)
name VARCHAR(255)
piece
id INT(11)
name VARCHAR(255)
type VARCHAR(255)
fk composer_id INT(11)
contains
iid INT(11)
pid INT(11)
instrument
id INT(11)
name VARCHAR(255)

```
	id INT AUTO_INCREMENT PRIMARY KEY,
	composer_id INT,
	name VARCHAR(255) NOT NULL,
	type VARCHAR(255) NOT NULL,
	CONSTRAINT unique_name UNIQUE (name),
	FOREIGN KEY (composer_id) REFERENCES composer(id)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE contains
(
	iid INT,
	pid INT,
	PRIMARY KEY many_to_many (iid, pid),
	FOREIGN KEY (iid) REFERENCES instrument(id),
	FOREIGN KEY (pid) REFERENCES piece(id)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- Fill the tables with values

-- country fill
INSERT INTO home_country(name) values ('[country]');

-- period fill
INSERT INTO period(name) values ('[period]');

-- instrument fill
INSERT INTO instrument(name) values ('[instrument]');

-- composer fill
INSERT INTO composer(fname, lname, c_id, p_id) values ('[first name]', '[last name]', (SELECT
id from home_country WHERE name='[country]'), (SELECT id from period WHERE
name='[period]'));

-- piece fill
INSERT INTO piece(composer_id, name, type) values ((SELECT id from composer WHERE
fname='[first name]' && lname='[last name]'),'[piece name]','[type of piece]');

-- contains fill
INSERT INTO contains(iid, pid) values ((SELECT id FROM instrument WHERE
name='[instrument name]'),
	(SELECT id FROM piece WHERE name='[piece name]'));
```

GENERAL USE QUERIES

View on home page

"SELECT piece.name, piece.type, composer.lname FROM piece INNER JOIN composer ON piece.composer_id = composer.id ORDER BY piece.name ASC”

Add functions

"INSERT INTO instrument(name) VALUES ([instrument name])”
"INSERT INTO home_country(name) VALUES ([country name])”
"INSERT INTO period(name) VALUES ([period name])”
"INSERT INTO piece(name, type) VALUES ([piece name],[piece type])”
"INSERT INTO composer(fname,lname) VALUES ([first name],[last name])”

Find all songs by a composer

"SELECT composer.lname, piece.name, piece.type FROM piece INNER JOIN composer ON piece.composer_id = composer.id WHERE composer.lname = [last name]”

Find all songs containing an instrument

"SELECT instrument.name, piece.name FROM piece INNER JOIN contains ON piece.id = contains.pid INNER JOIN instrument ON contains.iid = instrument.id WHERE instrument.name = [instrument name]”

See all composers from a specified country and see which period they are from

"SELECT home_country.name, composer.lname, period.name FROM composer INNER JOIN home_country ON composer.c_id = home_country.id INNER JOIN period ON composer.p_id = period.id WHERE home_country.name = [country name]“

Delete a composer

"DELETE FROM composer WHERE fname = [first name]“