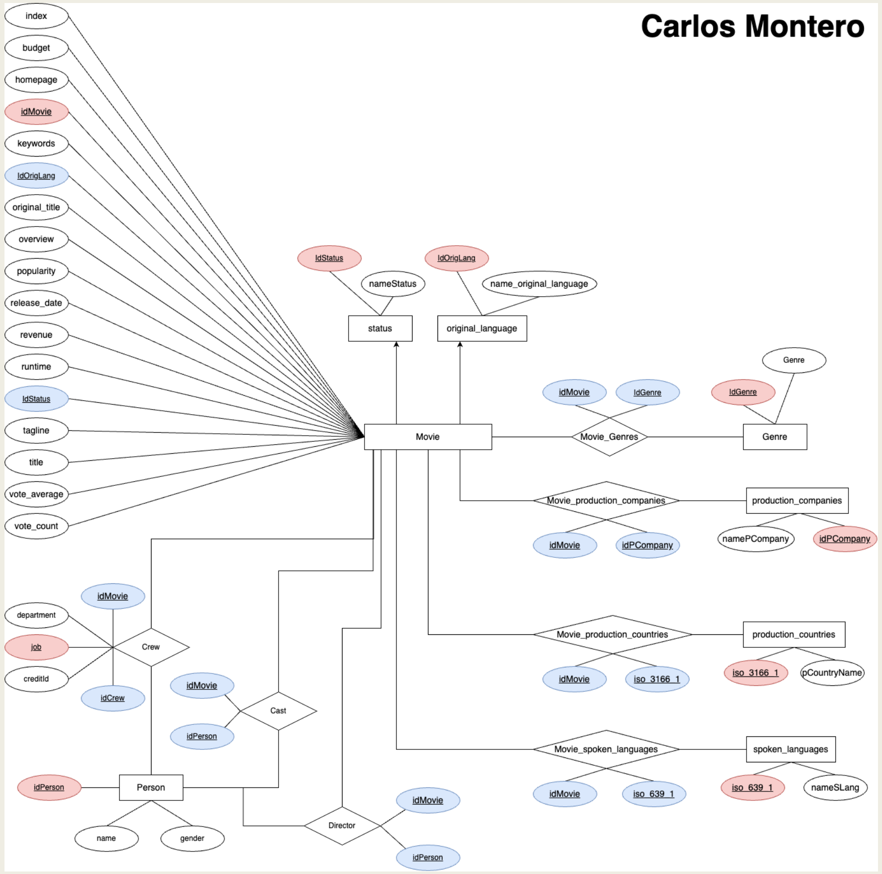
PROYECTO FINAL

Fundamentos de Bases de Datos

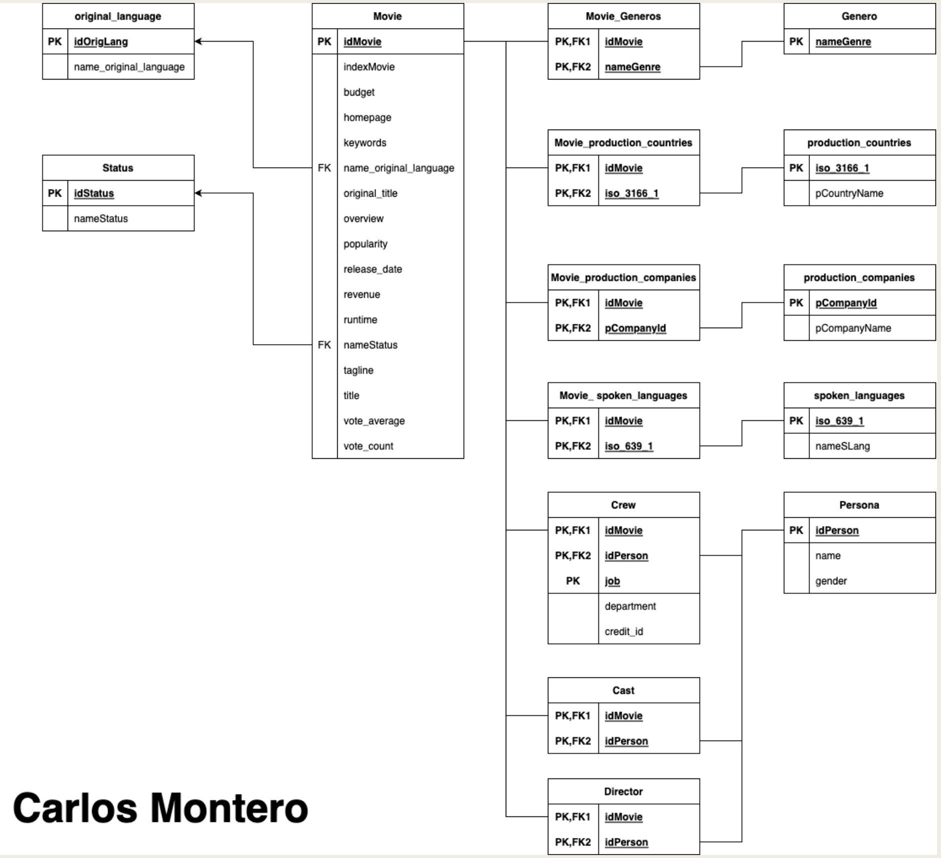
Carlos Iván Montero Torres

MODELOS

MODELO CONCEPTUAL



MODELO LOGICO



original_language	
PK	idOrigLang
	name_original_language

Movie	
PK	idMovie
	indexMovie
	budget
	homepage
	keywords
FK	name_original_language
	original_title
	overview
	popularity
	release_date
	revenue
	runtime
FK	nameStatus
	tagline
	title
	vote_average
	vote_count

Status	
PK	idStatus
	nameStatus

Movie_Generos	
PK,FK1	idMovie
PK,FK2	nameGenre

Genero	
PK	nameGenre

Movie_production_countries	
PK,FK1	idMovie
PK,FK2	iso_3166_1

production_countries	
PK	iso_3166_1
	pCountryName

Movie_production_companies	
PK,FK1	idMovie
PK,FK2	pCompanyId

production_companies	
PK	pCompanyId
	pCompanyName

Movie_ spoken_languages	
PK,FK1	idMovie
PK,FK2	iso_639_1

spoken_languages	
PK	iso_639_1
	nameSLang

Crew	
PK,FK1	idMovie
PK,FK2	idPerson
PK	job
	department
	credit_id

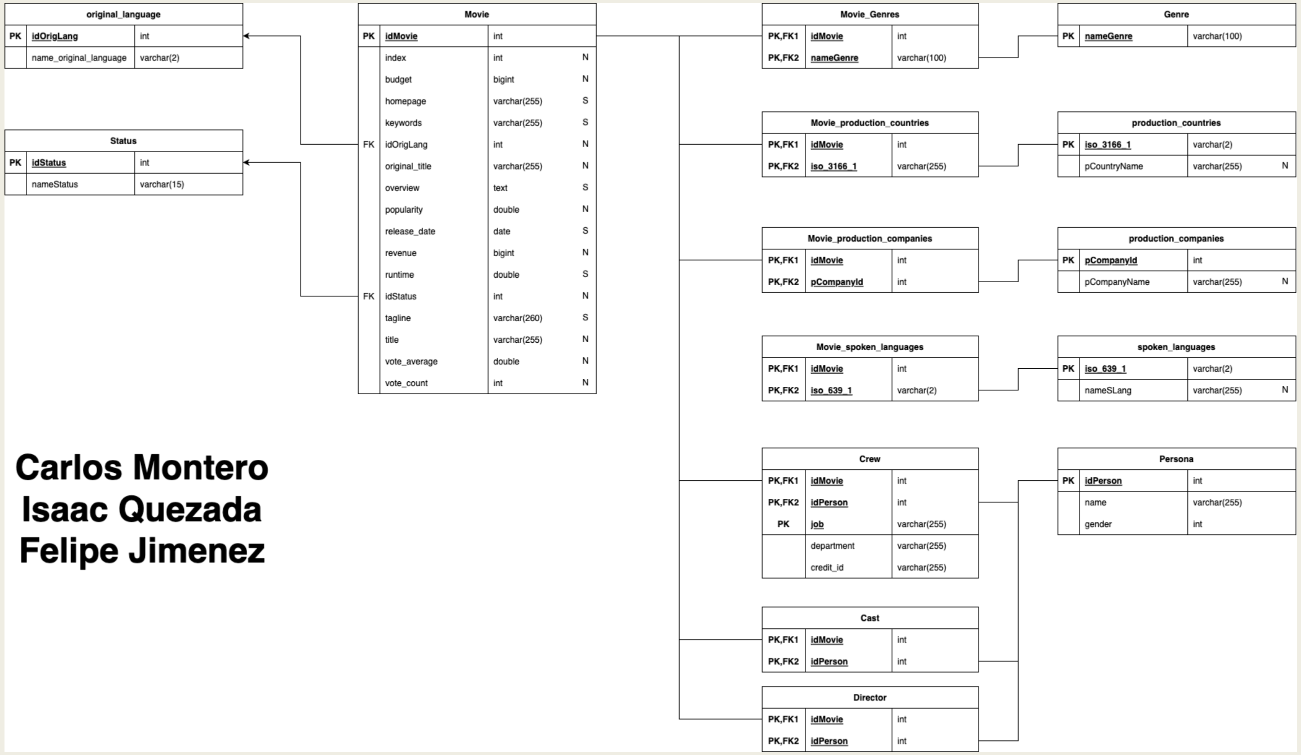
Persona	
PK	idPerson
	name
	gender

Cast	
PK,FK1	idMovie
PK,FK2	idPerson

Director	
PK,FK1	idMovie
PK,FK2	idPerson

Carlos Montero

MODELO FISICO



original_language

PK	idOrigLang	int
	name_original_language	varchar(2)

Status

PK	idStatus	int
	nameStatus	varchar(15)

Movie

PK	idMovie	int	
	index	int	N
	budget	bigint	N
	homepage	varchar(255)	S
	keywords	varchar(255)	S
FK	idOrigLang	int	N
	original_title	varchar(255)	N
	overview	text	S
	popularity	double	N
	release_date	date	S
	revenue	bigint	N
	runtime	double	S
FK	idStatus	int	N
	tagline	varchar(260)	S
	title	varchar(255)	N
	vote_average	double	N
	vote_count	int	N

Movie_Genres

PK,FK1	idMovie	int
PK,FK2	nameGenre	varchar(100)

Genre

PK	nameGenre	varchar(100)

Movie_production_countries

PK,FK1	idMovie	int
PK,FK2	iso_3166_1	varchar(255)

production_countries

PK	iso_3166_1	varchar(2)	
	pCountryName	varchar(255)	N

Movie_production_companies

PK,FK1	idMovie	int
PK,FK2	pCompanyId	int

production_companies

PK	pCompanyId	int	
	pCompanyName	varchar(255)	N

Movie_spoken_languages

PK,FK1	idMovie	int
PK,FK2	iso_639_1	varchar(2)

spoken_languages

PK	iso_639_1	varchar(2)	
	nameSLang	varchar(255)	N

Crew

PK,FK1	idMovie	int
PK,FK2	idPerson	int
PK	job	varchar(255)
	department	varchar(255)
	credit_id	varchar(255)

Persona

PK	idPerson	int
	name	varchar(255)
	gender	int

Cast

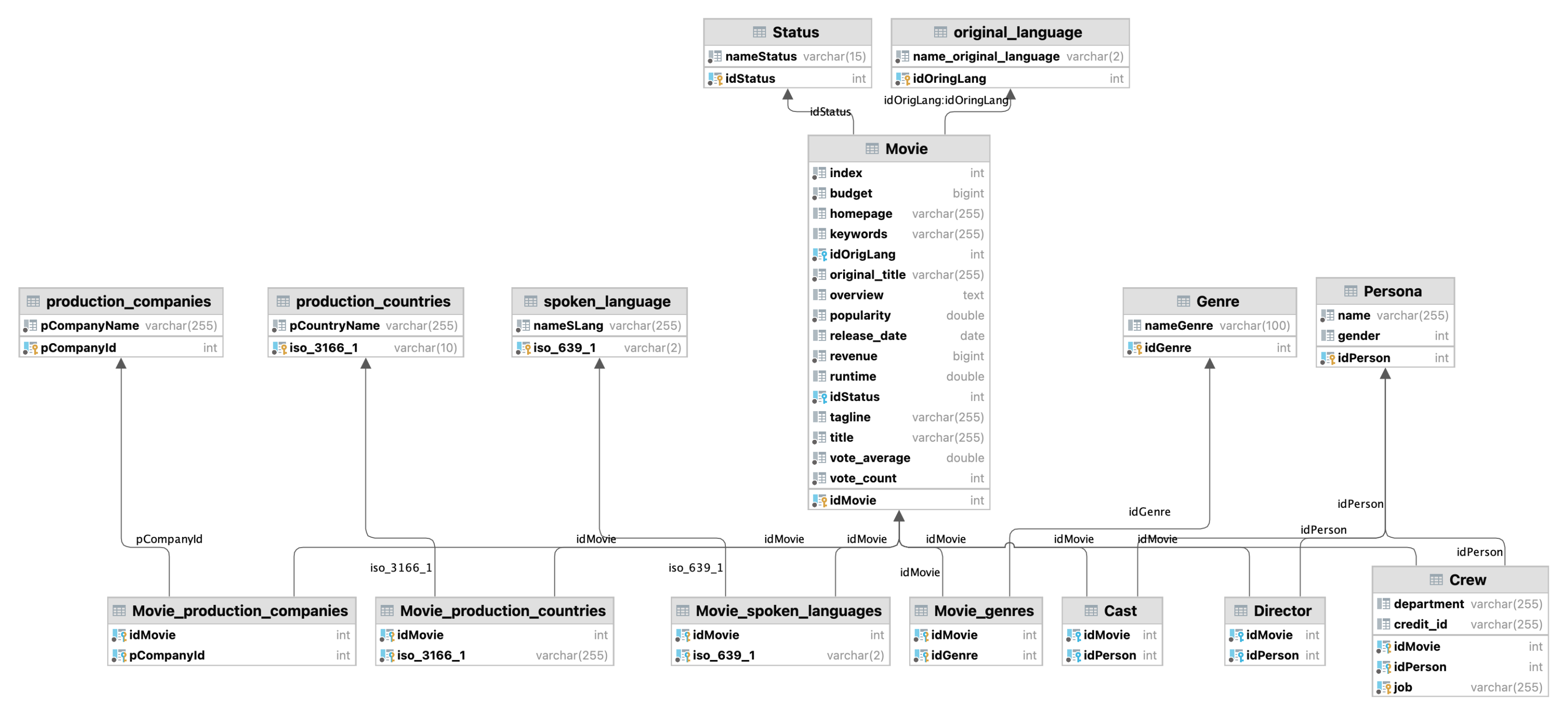
PK,FK1	idMovie	int
PK,FK2	idPerson	int

Director

PK,FK1	idMovie	int
PK,FK2	idPerson	int

Carlos Montero
Isaac Quezada
Felipe Jimenez

MODELO FISICO



Status	
nameStatus	varchar(15)
idStatus	int

original_language	
name_original_language	varchar(2)
idOringLang	int

Movie	
index	int
budget	bigint
homepage	varchar(255)
keywords	varchar(255)
idOrigLang	int
original_title	varchar(255)
overview	text
popularity	double
release_date	date
revenue	bigint
runtime	double
idStatus	int
tagline	varchar(255)
title	varchar(255)
vote_average	double
vote_count	int
idMovie	int

production_companies	
pCompanyName	varchar(255)
pCompanyId	int

production_countries	
pCountryName	varchar(255)
iso_3166_1	varchar(10)

spoken_language	
nameSLang	varchar(255)
iso_639_1	varchar(2)

Genre	
nameGenre	varchar(100)
idGenre	int

Persona	
name	varchar(255)
gender	int
idPerson	int

Movie_production_companies	
idMovie	int
pCompanyId	int

Movie_production_countries	
idMovie	int
iso_3166_1	varchar(255)

Movie_spoken_languages	
idMovie	int
iso_639_1	varchar(2)

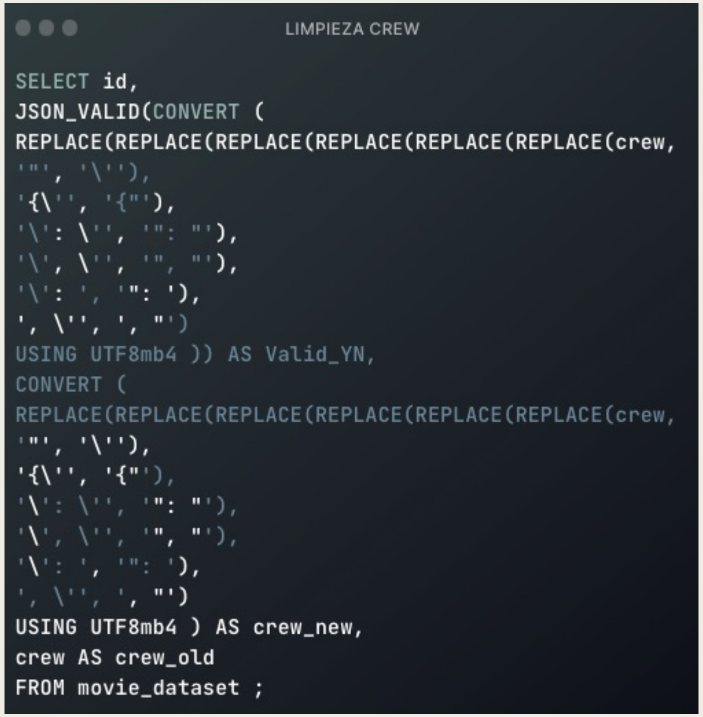
Movie_genres	
idMovie	int
idGenre	int

Cast	
idMovie	int
idPerson	int

Director	
idMovie	int
idPerson	int

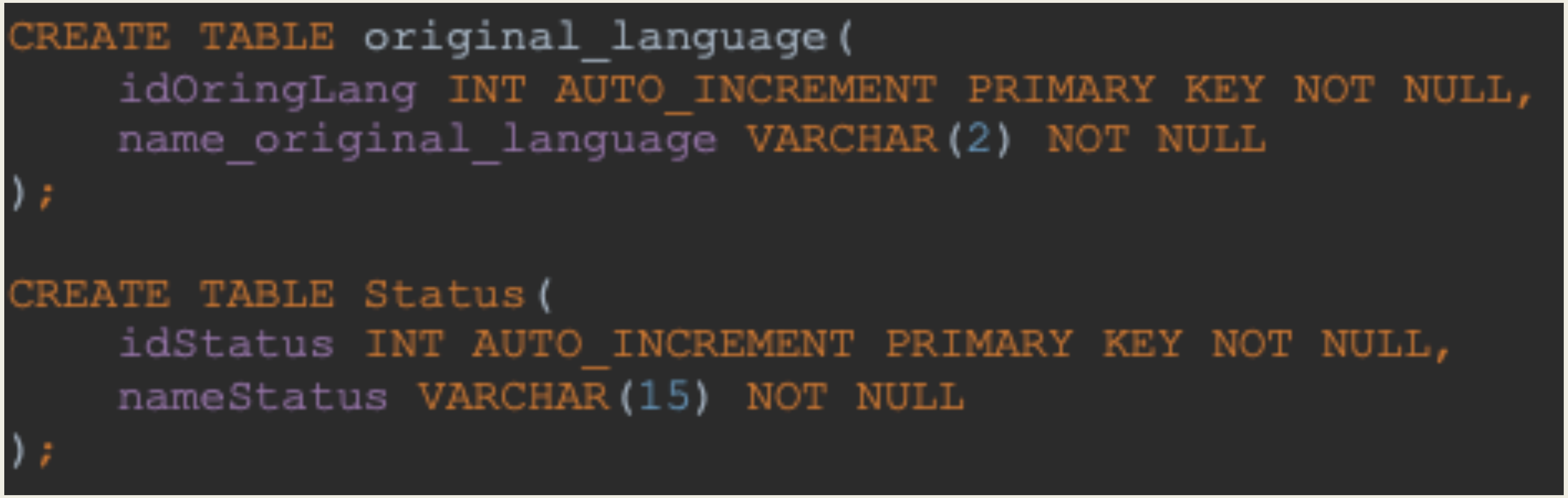
Crew	
department	varchar(255)
credit_id	varchar(255)
idMovie	int
idPerson	int
job	varchar(255)

LIMPIEZA “CREW”



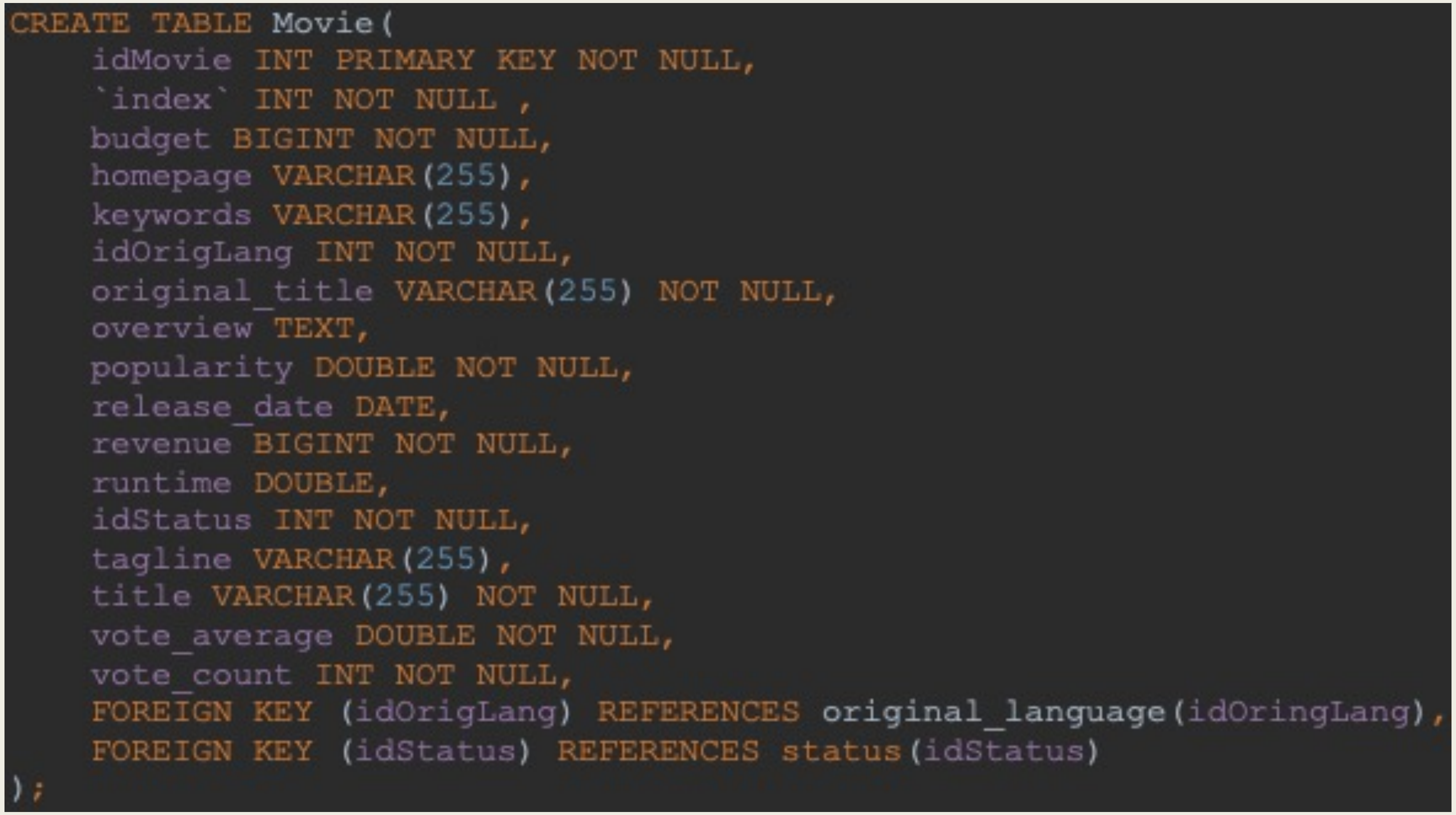
```sql
SELECT id,
JSON_VALID(CONVERT (
REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(crew,
'"', '\''),
'{\'', '{"'),
'\': \'', '": "'),
'\', \'', '", "'),
'\': ', '": '),
', \'', ', "')
USING UTF8mb4 )) AS Valid_YN,
CONVERT (
REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(crew,
'"', '\''),
'{\'', '{"'),
'\': \'', '": "'),
'\', \'', '", "'),
'\': ', '": '),
', \'', ', "')
USING UTF8mb4 ) AS crew_new,
crew AS crew_old
FROM movie_dataset ;
```

CREACION DE TABLAS



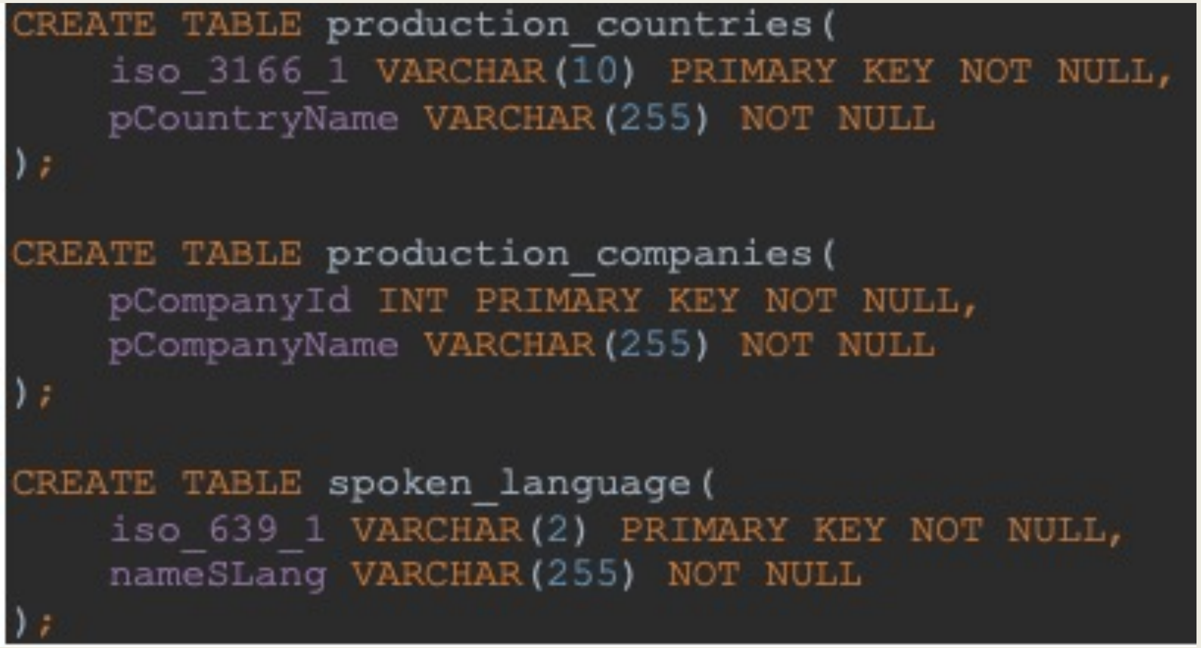
```
CREATE TABLE original_language(
    idOringLang INT AUTO_INCREMENT PRIMARY KEY NOT NULL,
    name_original_language VARCHAR(2) NOT NULL
);

CREATE TABLE Status(
    idStatus INT AUTO_INCREMENT PRIMARY KEY NOT NULL,
    nameStatus VARCHAR(15) NOT NULL
);
```

CREACION DE TABLAS



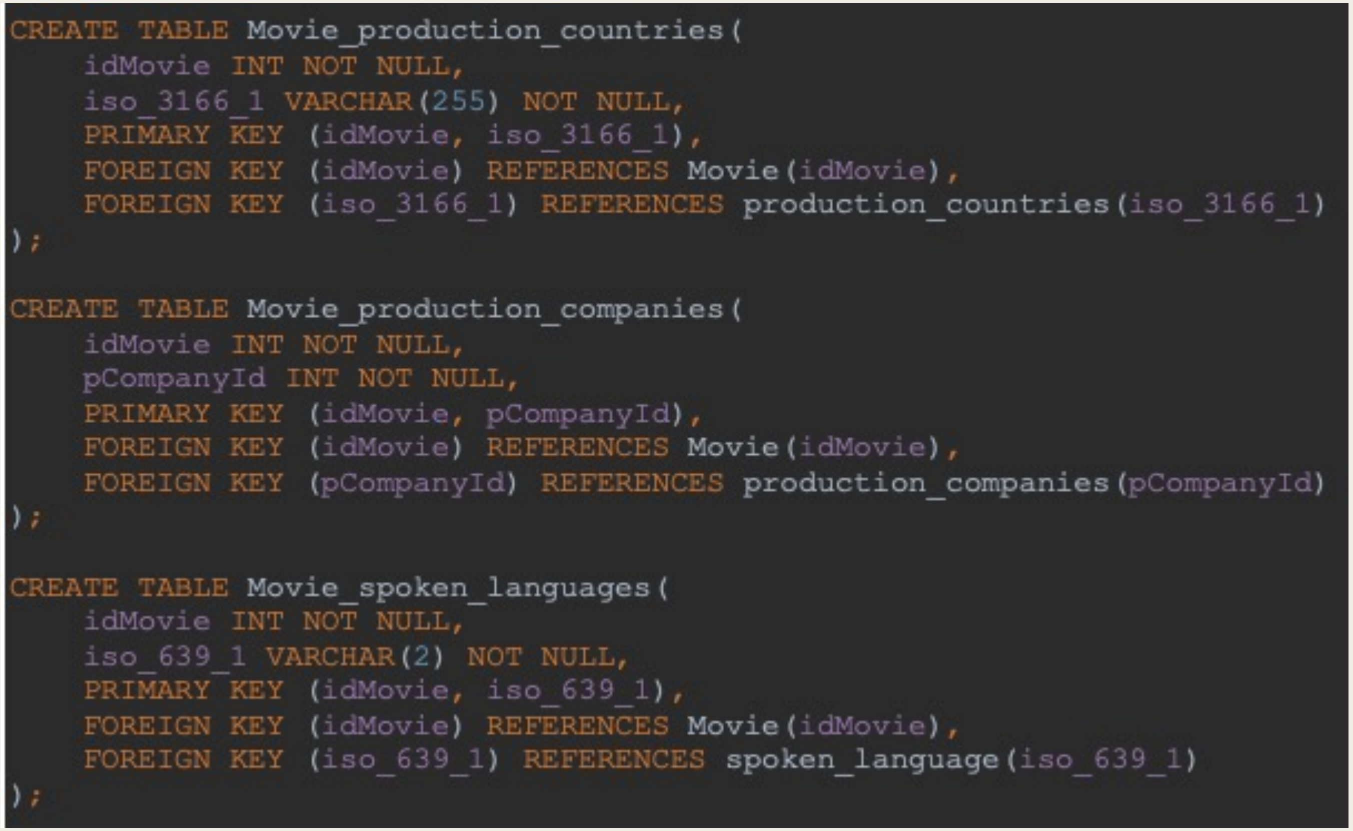
```sql
CREATE TABLE Movie(
    idMovie INT PRIMARY KEY NOT NULL,
    `index` INT NOT NULL ,
    budget BIGINT NOT NULL,
    homepage VARCHAR(255),
    keywords VARCHAR(255),
    idOrigLang INT NOT NULL,
    original_title VARCHAR(255) NOT NULL,
    overview TEXT,
    popularity DOUBLE NOT NULL,
    release_date DATE,
    revenue BIGINT NOT NULL,
    runtime DOUBLE,
    idStatus INT NOT NULL,
    tagline VARCHAR(255),
    title VARCHAR(255) NOT NULL,
    vote_average DOUBLE NOT NULL,
    vote_count INT NOT NULL,
    FOREIGN KEY (idOrigLang) REFERENCES original_language(idOringLang),
    FOREIGN KEY (idStatus) REFERENCES status(idStatus)
);
```

CREACION DE TABLAS



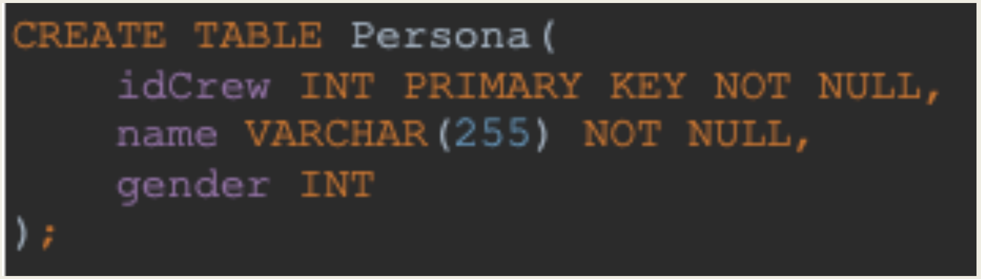
```
CREATE TABLE production_countries(
    iso_3166_1 VARCHAR(10) PRIMARY KEY NOT NULL,
    pCountryName VARCHAR(255) NOT NULL
);

CREATE TABLE production_companies(
    pCompanyId INT PRIMARY KEY NOT NULL,
    pCompanyName VARCHAR(255) NOT NULL
);

CREATE TABLE spoken_language(
    iso_639_1 VARCHAR(2) PRIMARY KEY NOT NULL,
    nameSLang VARCHAR(255) NOT NULL
);
```

CREACION DE TABLAS

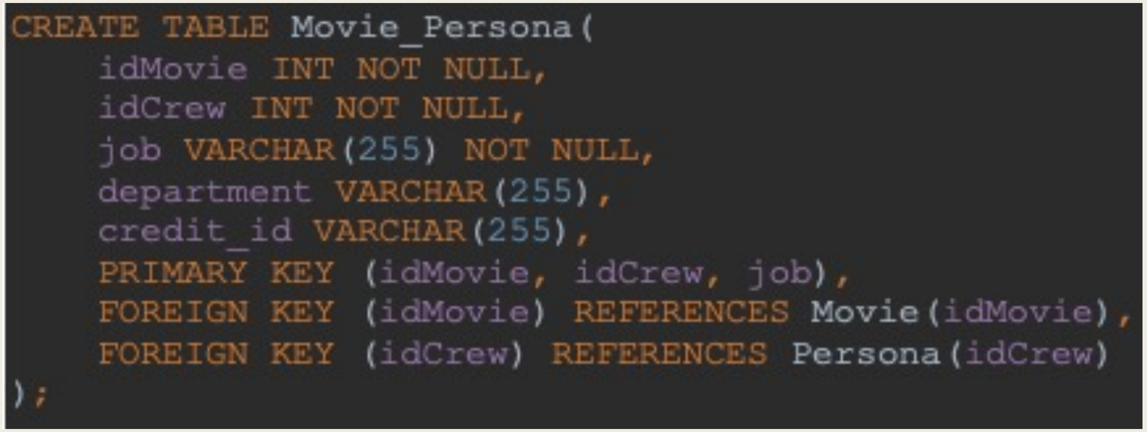


```
CREATE TABLE Movie_production_countries(
    idMovie INT NOT NULL,
    iso_3166_1 VARCHAR(255) NOT NULL,
    PRIMARY KEY (idMovie, iso_3166_1),
    FOREIGN KEY (idMovie) REFERENCES Movie(idMovie),
    FOREIGN KEY (iso_3166_1) REFERENCES production_countries(iso_3166_1)
);

CREATE TABLE Movie_production_companies(
    idMovie INT NOT NULL,
    pCompanyId INT NOT NULL,
    PRIMARY KEY (idMovie, pCompanyId),
    FOREIGN KEY (idMovie) REFERENCES Movie(idMovie),
    FOREIGN KEY (pCompanyId) REFERENCES production_companies(pCompanyId)
);

CREATE TABLE Movie_spoken_languages(
    idMovie INT NOT NULL,
    iso_639_1 VARCHAR(2) NOT NULL,
    PRIMARY KEY (idMovie, iso_639_1),
    FOREIGN KEY (idMovie) REFERENCES Movie(idMovie),
    FOREIGN KEY (iso_639_1) REFERENCES spoken_language(iso_639_1)
);
```

CREACION DE TABLAS

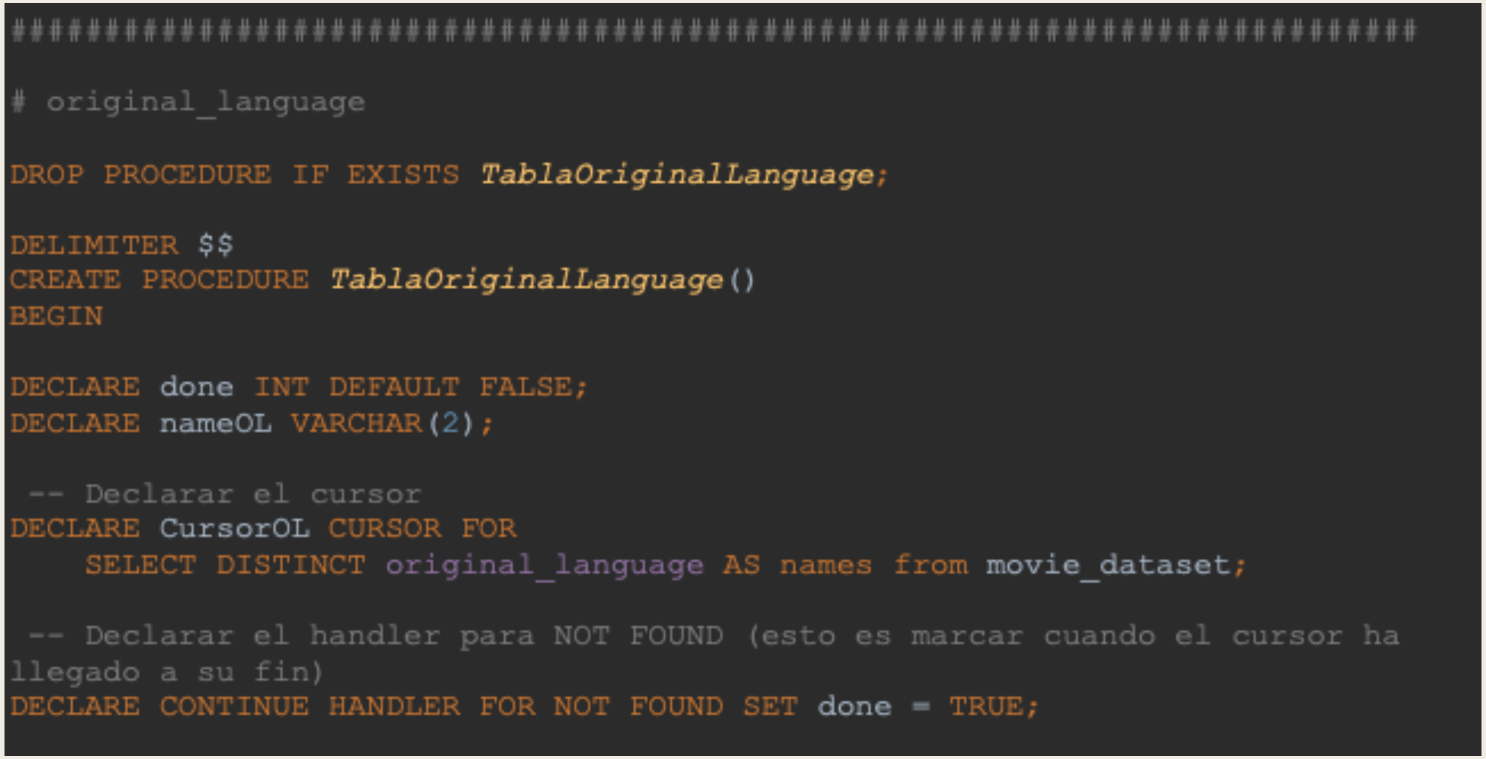


```
CREATE TABLE Persona(
    idCrew INT PRIMARY KEY NOT NULL,
    name VARCHAR(255) NOT NULL,
    gender INT
);
```



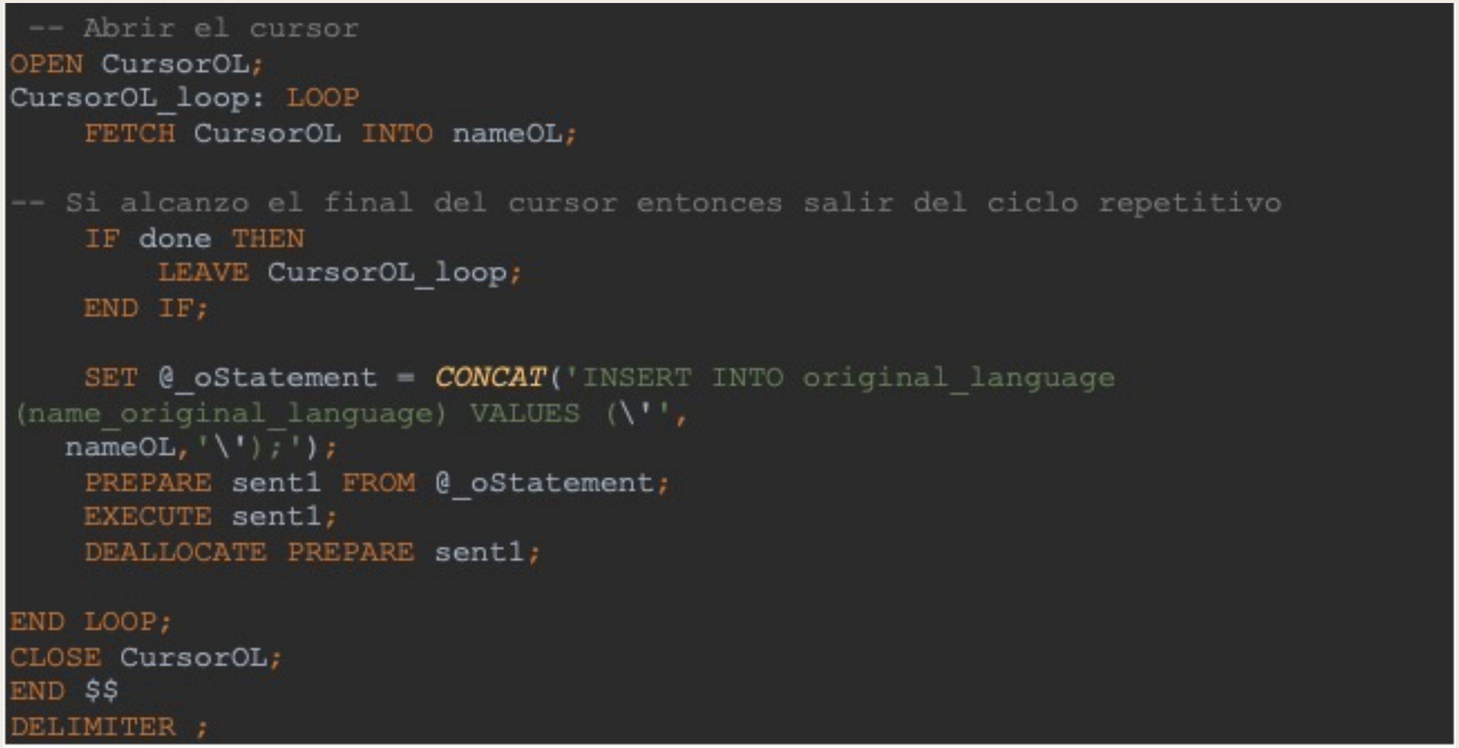
```
CREATE TABLE Movie_Persona(
    idMovie INT NOT NULL,
    idCrew INT NOT NULL,
    job VARCHAR(255) NOT NULL,
    department VARCHAR(255),
    credit_id VARCHAR(255),
    PRIMARY KEY (idMovie, idCrew, job),
    FOREIGN KEY (idMovie) REFERENCES Movie(idMovie),
    FOREIGN KEY (idCrew) REFERENCES Persona(idCrew)
);
```

CURSORES & PROCEDURES

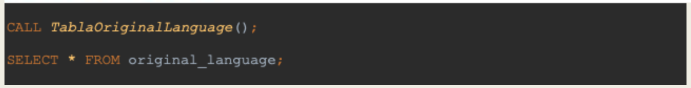


```
###########################################################################

# original_language

DROP PROCEDURE IF EXISTS TablaOriginalLanguage;

DELIMITER $$
CREATE PROCEDURE TablaOriginalLanguage()
BEGIN

DECLARE done INT DEFAULT FALSE;
DECLARE nameOL VARCHAR(2);

 -- Declarar el cursor
DECLARE CursorOL CURSOR FOR
    SELECT DISTINCT original_language AS names from movie_dataset;

 -- Declarar el handler para NOT FOUND (esto es marcar cuando el cursor ha
llegado a su fin)
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

CURSORES & PROCEDURES

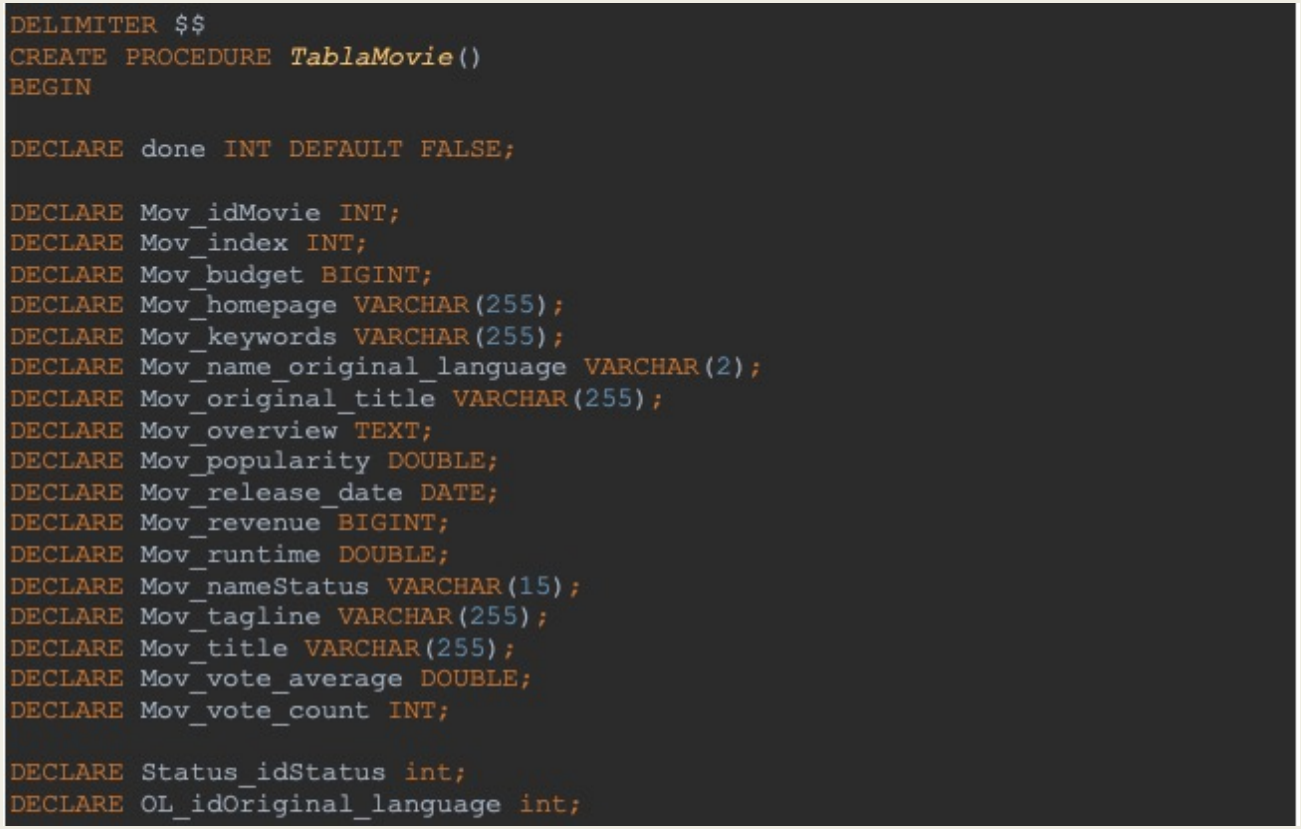


```
 -- Abrir el cursor
OPEN CursorOL;
CursorOL_loop: LOOP
    FETCH CursorOL INTO nameOL;

-- Si alcanzo el final del cursor entonces salir del ciclo repetitivo
    IF done THEN
        LEAVE CursorOL_loop;
    END IF;

    SET @_oStatement = CONCAT('INSERT INTO original_language
(name_original_language) VALUES (\'',
   nameOL,'\');');
    PREPARE sent1 FROM @_oStatement;
    EXECUTE sent1;
    DEALLOCATE PREPARE sent1;

END LOOP;
CLOSE CursorOL;
END $$
DELIMITER ;
```



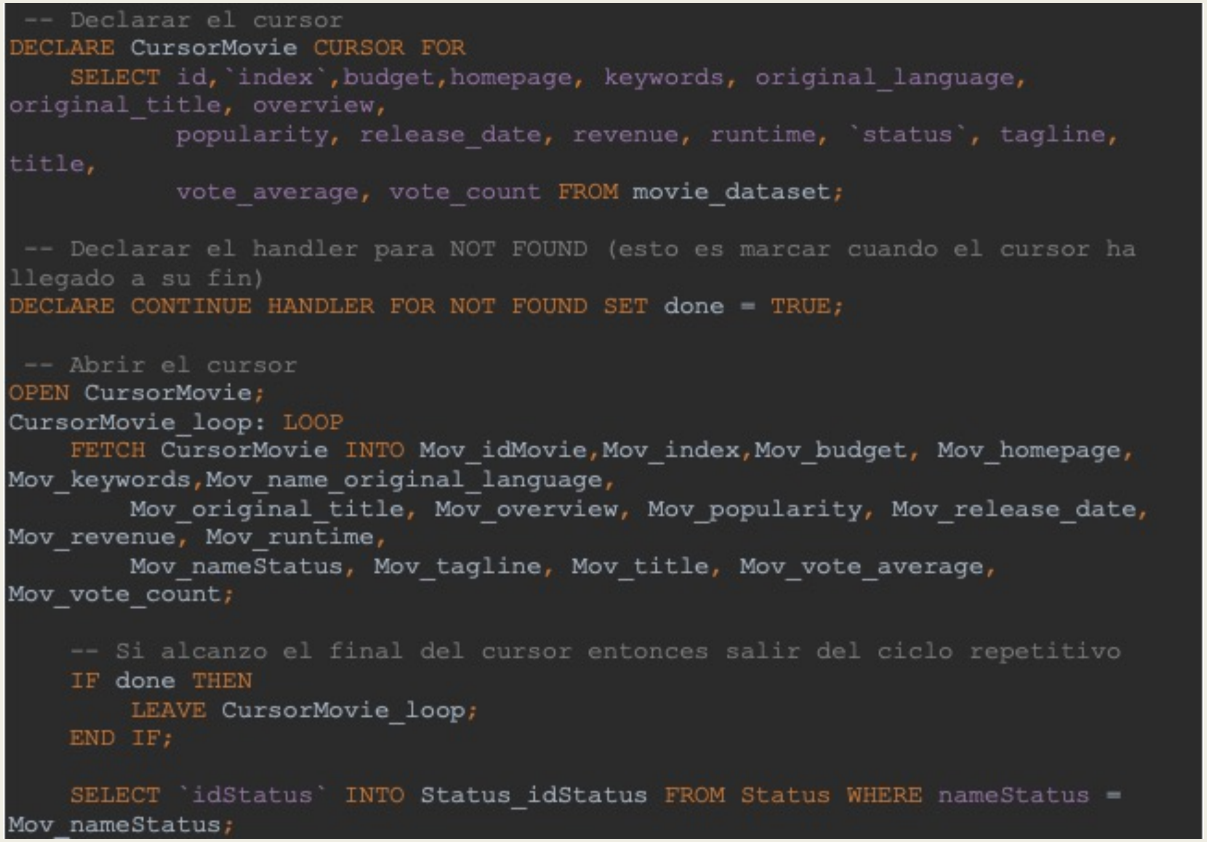
```
CALL TablaOriginalLanguage();

SELECT * FROM original_language;
```

CURSORES & PROCEDURES



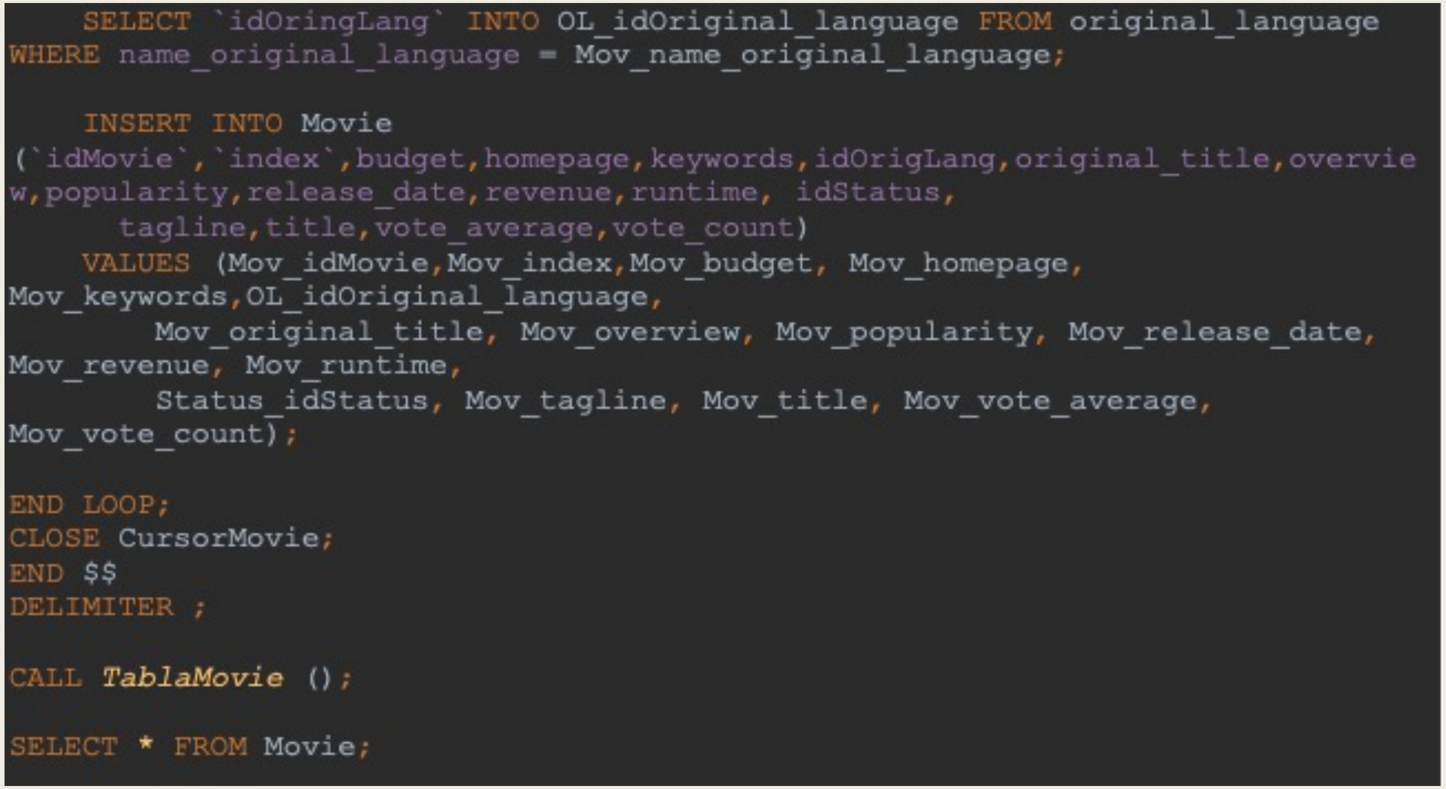
```
DELIMITER $$
CREATE PROCEDURE TablaMovie()
BEGIN

DECLARE done INT DEFAULT FALSE;

DECLARE Mov_idMovie INT;
DECLARE Mov_index INT;
DECLARE Mov_budget BIGINT;
DECLARE Mov_homepage VARCHAR(255);
DECLARE Mov_keywords VARCHAR(255);
DECLARE Mov_name_original_language VARCHAR(2);
DECLARE Mov_original_title VARCHAR(255);
DECLARE Mov_overview TEXT;
DECLARE Mov_popularity DOUBLE;
DECLARE Mov_release_date DATE;
DECLARE Mov_revenue BIGINT;
DECLARE Mov_runtime DOUBLE;
DECLARE Mov_nameStatus VARCHAR(15);
DECLARE Mov_tagline VARCHAR(255);
DECLARE Mov_title VARCHAR(255);
DECLARE Mov_vote_average DOUBLE;
DECLARE Mov_vote_count INT;

DECLARE Status_idStatus int;
DECLARE OL_idOriginal_language int;
```

CURSORES & PROCEDURES



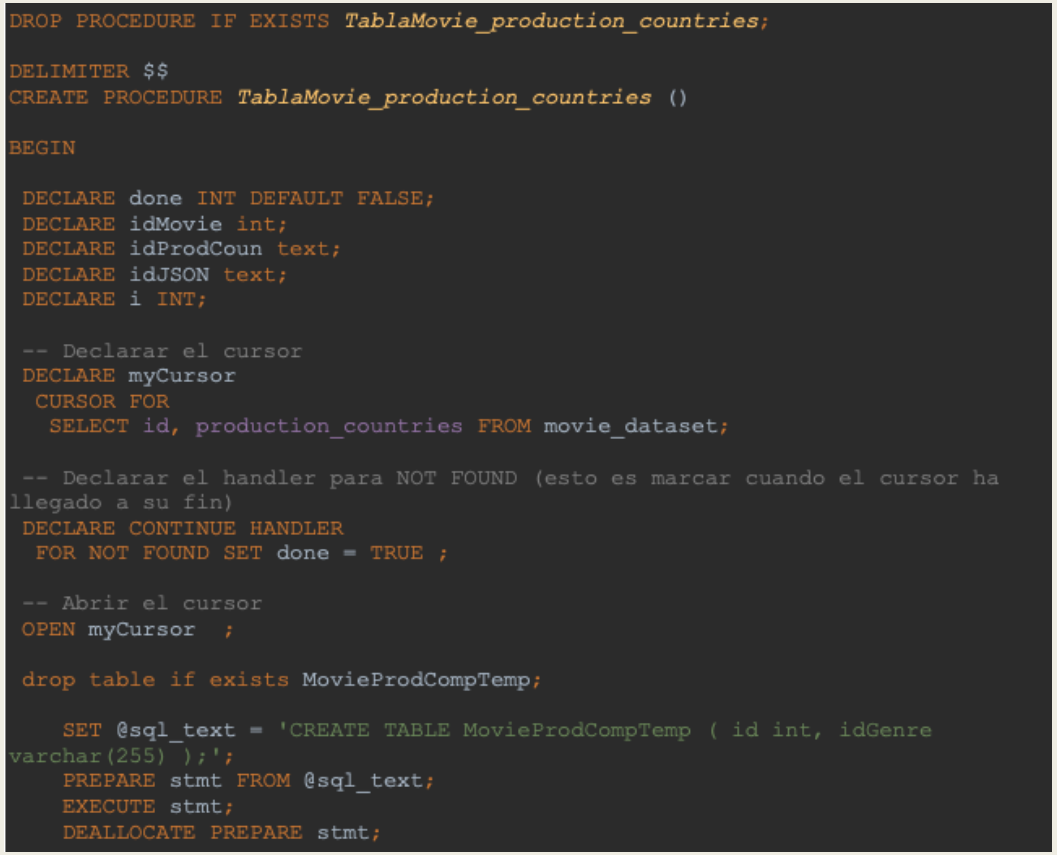
```
 -- Declarar el cursor
DECLARE CursorMovie CURSOR FOR
    SELECT id,`index`,budget,homepage, keywords, original_language,
original_title, overview,
           popularity, release_date, revenue, runtime, `status`, tagline,
title,
           vote_average, vote_count FROM movie_dataset;

 -- Declarar el handler para NOT FOUND (esto es marcar cuando el cursor ha
llegado a su fin)
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

 -- Abrir el cursor
OPEN CursorMovie;
CursorMovie_loop: LOOP
    FETCH CursorMovie INTO Mov_idMovie,Mov_index,Mov_budget, Mov_homepage,
Mov_keywords,Mov_name_original_language,
        Mov_original_title, Mov_overview, Mov_popularity, Mov_release_date,
Mov_revenue, Mov_runtime,
        Mov_nameStatus, Mov_tagline, Mov_title, Mov_vote_average,
Mov_vote_count;

    -- Si alcanzo el final del cursor entonces salir del ciclo repetitivo
    IF done THEN
        LEAVE CursorMovie_loop;
    END IF;

    SELECT `idStatus` INTO Status_idStatus FROM Status WHERE nameStatus =
Mov_nameStatus;
```

CURSORES & PROCEDURES



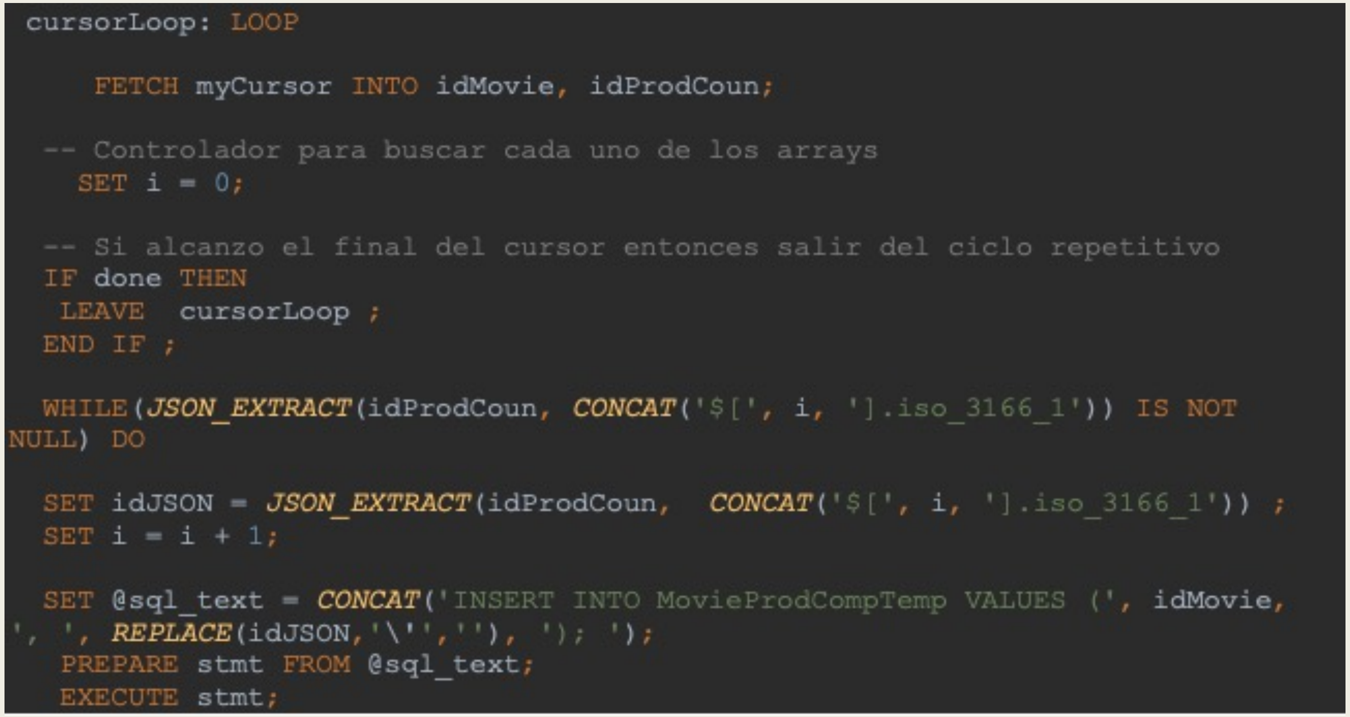
```
    SELECT `idOringLang` INTO OL_idOriginal_language FROM original_language
WHERE name_original_language = Mov_name_original_language;

    INSERT INTO Movie
(`idMovie`,`index`,budget,homepage,keywords,idOrigLang,original_title,overvie
w,popularity,release_date,revenue,runtime, idStatus,
      tagline,title,vote_average,vote_count)
    VALUES (Mov_idMovie,Mov_index,Mov_budget, Mov_homepage,
Mov_keywords,OL_idOriginal_language,
        Mov_original_title, Mov_overview, Mov_popularity, Mov_release_date,
Mov_revenue, Mov_runtime,
        Status_idStatus, Mov_tagline, Mov_title, Mov_vote_average,
Mov_vote_count);

END LOOP;
CLOSE CursorMovie;
END $$
DELIMITER ;

CALL TablaMovie ();

SELECT * FROM Movie;
```

CURSORES & PROCEDURES



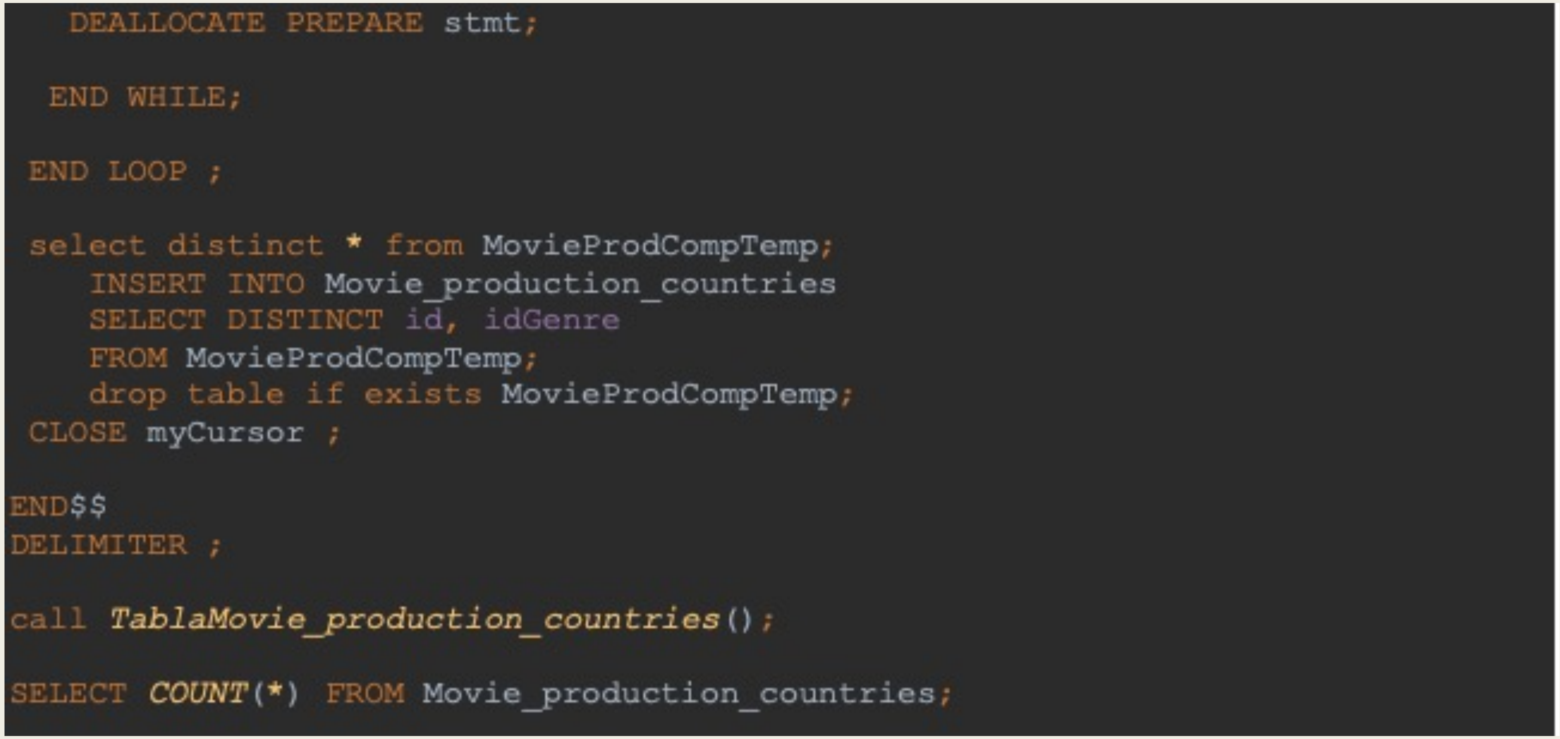
```
DROP PROCEDURE IF EXISTS TablaMovie_production_countries;

DELIMITER $$
CREATE PROCEDURE TablaMovie_production_countries ()

BEGIN

 DECLARE done INT DEFAULT FALSE;
 DECLARE idMovie int;
 DECLARE idProdCoun text;
 DECLARE idJSON text;
 DECLARE i INT;

 -- Declarar el cursor
 DECLARE myCursor
  CURSOR FOR
   SELECT id, production_countries FROM movie_dataset;

 -- Declarar el handler para NOT FOUND (esto es marcar cuando el cursor ha
llegado a su fin)
 DECLARE CONTINUE HANDLER
  FOR NOT FOUND SET done = TRUE ;

 -- Abrir el cursor
 OPEN myCursor  ;

 drop table if exists MovieProdCompTemp;

    SET @sql_text = 'CREATE TABLE MovieProdCompTemp ( id int, idGenre
varchar(255) );';
    PREPARE stmt FROM @sql_text;
    EXECUTE stmt;
    DEALLOCATE PREPARE stmt;
```

CURSORES & PROCEDURES



```
 cursorLoop: LOOP

     FETCH myCursor INTO idMovie, idProdCoun;

  -- Controlador para buscar cada uno de los arrays
    SET i = 0;

  -- Si alcanzo el final del cursor entonces salir del ciclo repetitivo
  IF done THEN
   LEAVE  cursorLoop ;
  END IF ;

  WHILE(JSON_EXTRACT(idProdCoun, CONCAT('$[', i, '].iso_3166_1')) IS NOT
NULL) DO

  SET idJSON = JSON_EXTRACT(idProdCoun,  CONCAT('$[', i, '].iso_3166_1')) ;
  SET i = i + 1;

  SET @sql_text = CONCAT('INSERT INTO MovieProdCompTemp VALUES (', idMovie,
', ', REPLACE(idJSON,'\'',''), '); ');
   PREPARE stmt FROM @sql_text;
   EXECUTE stmt;
```

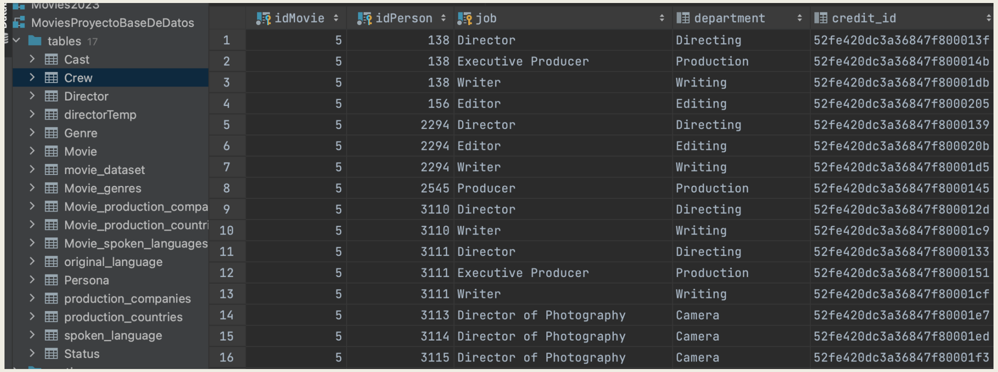
CURSORES & PROCEDURES



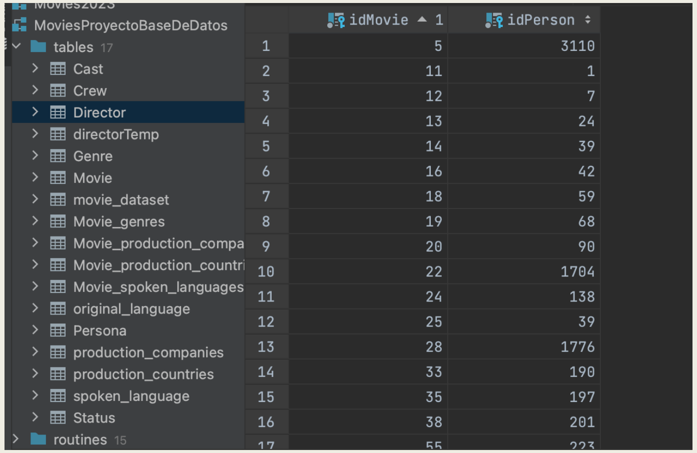
```
      DEALLOCATE PREPARE stmt;

    END WHILE;

  END LOOP ;

  select distinct * from MovieProdCompTemp;
     INSERT INTO Movie_production_countries
     SELECT DISTINCT id, idGenre
     FROM MovieProdCompTemp;
     drop table if exists MovieProdCompTemp;
  CLOSE myCursor ;

END$$
DELIMITER ;

call TablaMovie_production_countries();

SELECT COUNT(*) FROM Movie_production_countries;
```

POBLACIÓN DE TABLAS

Población de tablas

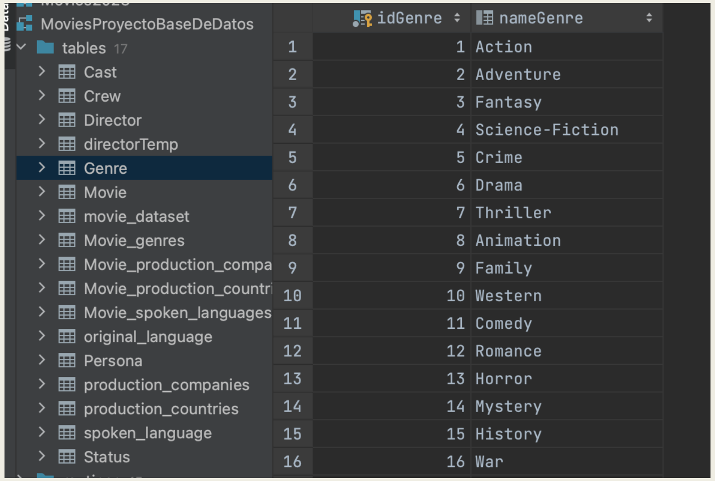


	idMovie	idPerson	job	department	credit_id
1	5	138	Director	Directing	52fe420dc3a36847f800013f
2	5	138	Executive Producer	Production	52fe420dc3a36847f800014b
3	5	138	Writer	Writing	52fe420dc3a36847f80001db
4	5	156	Editor	Editing	52fe420dc3a36847f8000205
5	5	2294	Director	Directing	52fe420dc3a36847f8000139
6	5	2294	Editor	Editing	52fe420dc3a36847f800020b
7	5	2294	Writer	Writing	52fe420dc3a36847f80001d5
8	5	2545	Producer	Production	52fe420dc3a36847f8000145
9	5	3110	Director	Directing	52fe420dc3a36847f800012d
10	5	3110	Writer	Writing	52fe420dc3a36847f80001c9
11	5	3111	Director	Directing	52fe420dc3a36847f8000133
12	5	3111	Executive Producer	Production	52fe420dc3a36847f8000151
13	5	3111	Writer	Writing	52fe420dc3a36847f80001cf
14	5	3113	Director of Photography	Camera	52fe420dc3a36847f80001e7
15	5	3114	Director of Photography	Camera	52fe420dc3a36847f80001ed
16	5	3115	Director of Photography	Camera	52fe420dc3a36847f80001f3

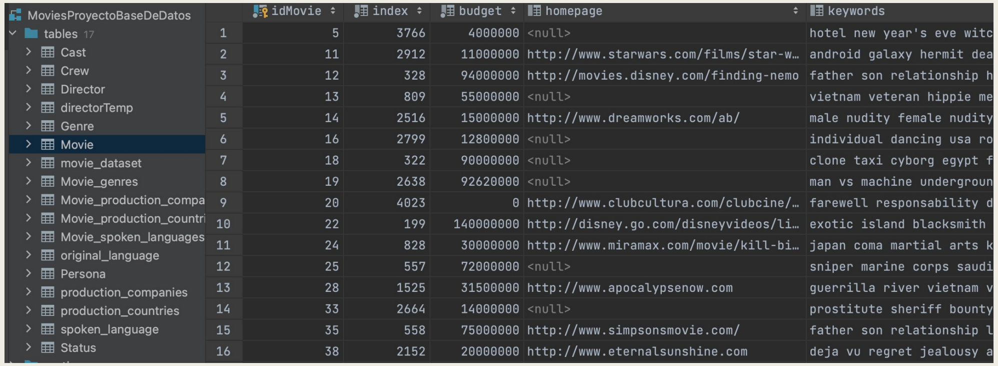


	idMovie ▲ 1	idPerson
1	5	3110
2	11	1
3	12	7
4	13	24
5	14	39
6	16	42
7	18	59
8	19	68
9	20	90
10	22	1704
11	24	138
12	25	39
13	28	1776
14	33	190
15	35	197
16	38	201
17	55	223

Población de tablas

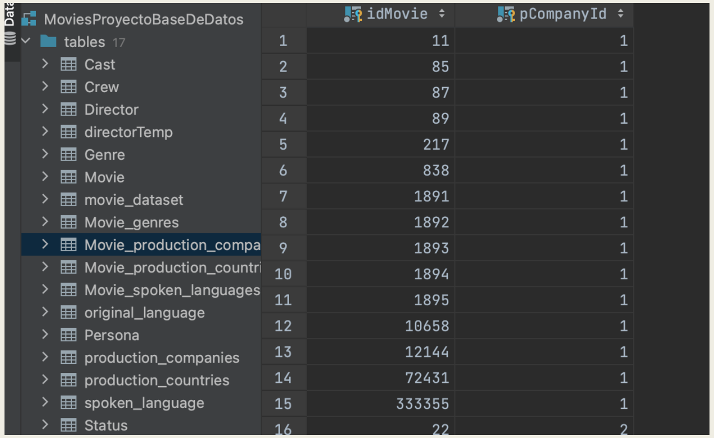


	idGenre	nameGenre
1	1	Action
2	2	Adventure
3	3	Fantasy
4	4	Science-Fiction
5	5	Crime
6	6	Drama
7	7	Thriller
8	8	Animation
9	9	Family
10	10	Western
11	11	Comedy
12	12	Romance
13	13	Horror
14	14	Mystery
15	15	History
16	16	War



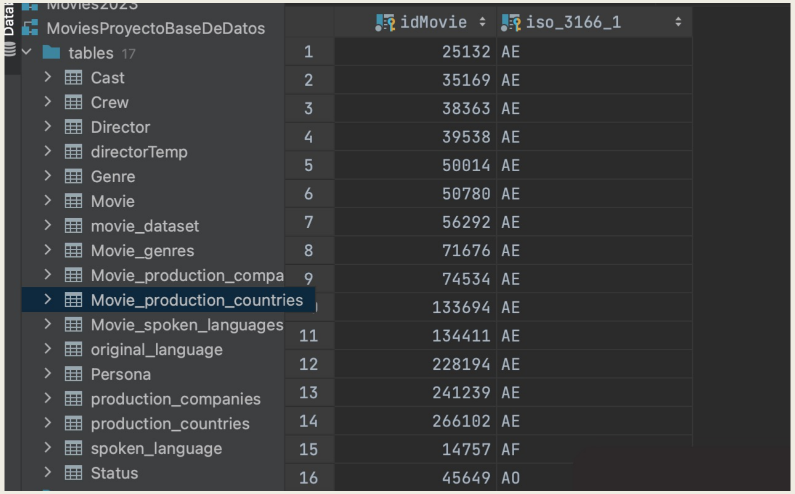
	idMovie	index	budget	homepage	keywords
1	5	3766	4000000	<null>	hotel new year's eve witc
2	11	2912	11000000	http://www.starwars.com/films/star-w…	android galaxy hermit dea
3	12	328	94000000	http://movies.disney.com/finding-nemo	father son relationship h
4	13	809	55000000	<null>	vietnam veteran hippie me
5	14	2516	15000000	http://www.dreamworks.com/ab/	male nudity female nudity
6	16	2799	12800000	<null>	individual dancing usa ro
7	18	322	90000000	<null>	clone taxi cyborg egypt f
8	19	2638	92620000	<null>	man vs machine undergroun
9	20	4023	0	http://www.clubcultura.com/clubcine/…	farewell responsability d
10	22	199	140000000	http://disney.go.com/disneyvideos/li…	exotic island blacksmith
11	24	828	30000000	http://www.miramax.com/movie/kill-bi…	japan coma martial arts k
12	25	557	72000000	<null>	sniper marine corps saudi
13	28	1525	31500000	http://www.apocalypsenow.com	guerrilla river vietnam v
14	33	2664	14000000	<null>	prostitute sheriff bounty
15	35	558	75000000	http://www.simpsonsmovie.com/	father son relationship l
16	38	2152	20000000	http://www.eternalsunshine.com	deja vu regret jealousy a

Población de tablas



MoviesProyectoBaseDeDatos
tables 17
- Cast
- Crew
- Director
- directorTemp
- Genre
- Movie
- movie_dataset
- Movie_genres
- Movie_production_compa
- Movie_production_countri
- Movie_spoken_languages
- original_language
- Persona
- production_companies
- production_countries
- spoken_language
- Status

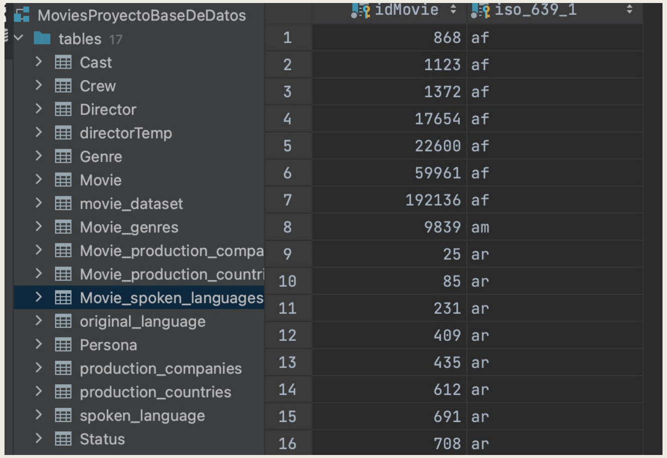
	idMovie	pCompanyId
1	11	1
2	85	1
3	87	1
4	89	1
5	217	1
6	838	1
7	1891	1
8	1892	1
9	1893	1
10	1894	1
11	1895	1
12	10658	1
13	12144	1
14	72431	1
15	333355	1
16	22	2



Movies2023
MoviesProyectoBaseDeDatos
tables 17
- Cast
- Crew
- Director
- directorTemp
- Genre
- Movie
- movie_dataset
- Movie_genres
- Movie_production_compa
- Movie_production_countries
- Movie_spoken_languages
- original_language
- Persona
- production_companies
- production_countries
- spoken_language
- Status

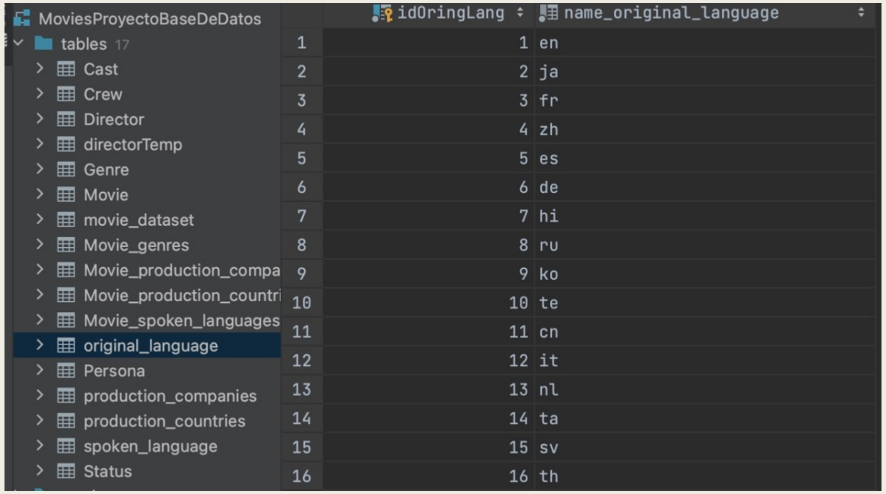
	idMovie	iso_3166_1
1	25132	AE
2	35169	AE
3	38363	AE
4	39538	AE
5	50014	AE
6	50780	AE
7	56292	AE
8	71676	AE
9	74534	AE
	133694	AE
11	134411	AE
12	228194	AE
13	241239	AE
14	266102	AE
15	14757	AF
16	45649	AO

Población de tablas



MoviesProyectoBaseDeDatos
- tables 17
  - Cast
  - Crew
  - Director
  - directorTemp
  - Genre
  - Movie
  - movie_dataset
  - Movie_genres
  - Movie_production_compa
  - Movie_production_countri
  - Movie_spoken_languages
  - original_language
  - Persona
  - production_companies
  - production_countries
  - spoken_language
  - Status

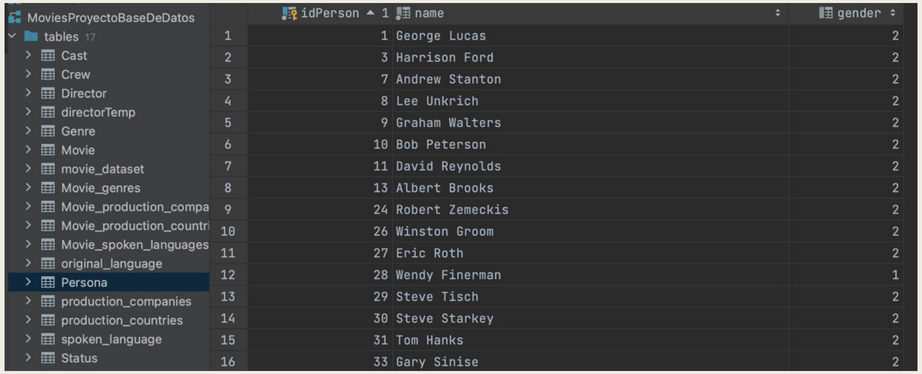
	idMovie	iso_639_1
1	868	af
2	1123	af
3	1372	af
4	17654	af
5	22600	af
6	59961	af
7	192136	af
8	9839	am
9	25	ar
10	85	ar
11	231	ar
12	409	ar
13	435	ar
14	612	ar
15	691	ar
16	708	ar



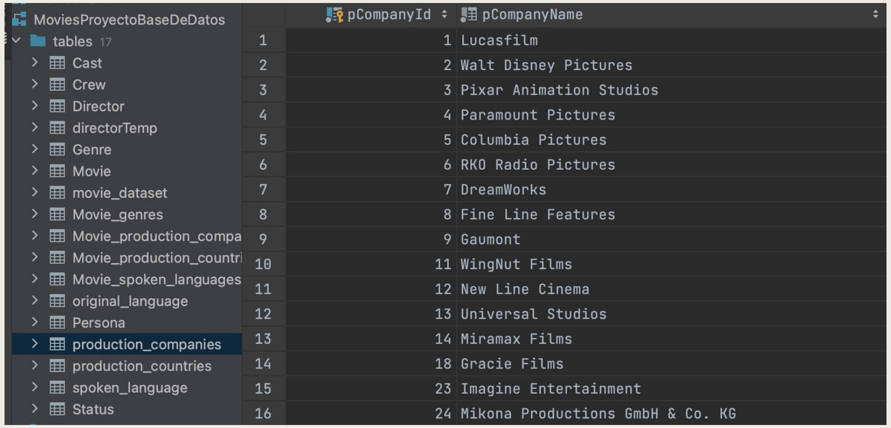
MoviesProyectoBaseDeDatos
- tables 17
  - Cast
  - Crew
  - Director
  - directorTemp
  - Genre
  - Movie
  - movie_dataset
  - Movie_genres
  - Movie_production_compa
  - Movie_production_countri
  - Movie_spoken_languages
  - original_language
  - Persona
  - production_companies
  - production_countries
  - spoken_language
  - Status

	idOringLang	name_original_language
1	1	en
2	2	ja
3	3	fr
4	4	zh
5	5	es
6	6	de
7	7	hi
8	8	ru
9	9	ko
10	10	te
11	11	cn
12	12	it
13	13	nl
14	14	ta
15	15	sv
16	16	th

Población de tablas

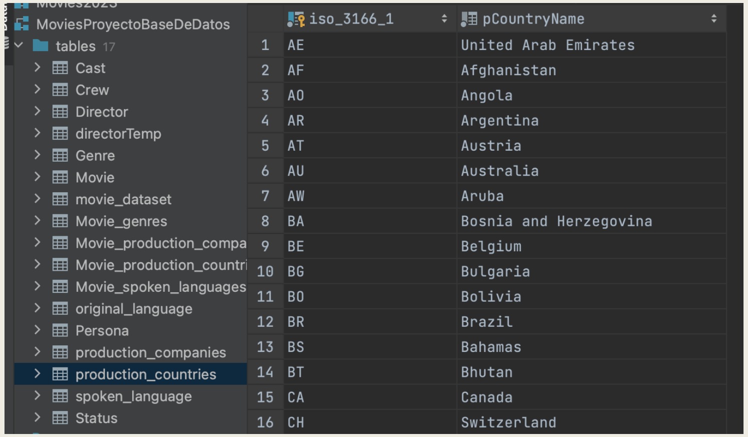


	idPerson	name	gender
1	1	George Lucas	2
2	3	Harrison Ford	2
3	7	Andrew Stanton	2
4	8	Lee Unkrich	2
5	9	Graham Walters	2
6	10	Bob Peterson	2
7	11	David Reynolds	2
8	13	Albert Brooks	2
9	24	Robert Zemeckis	2
10	26	Winston Groom	2
11	27	Eric Roth	2
12	28	Wendy Finerman	1
13	29	Steve Tisch	2
14	30	Steve Starkey	2
15	31	Tom Hanks	2
16	33	Gary Sinise	2

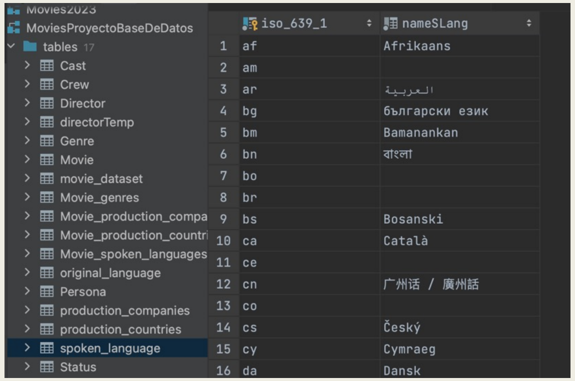


	pCompanyId	pCompanyName
1	1	Lucasfilm
2	2	Walt Disney Pictures
3	3	Pixar Animation Studios
4	4	Paramount Pictures
5	5	Columbia Pictures
6	6	RKO Radio Pictures
7	7	DreamWorks
8	8	Fine Line Features
9	9	Gaumont
10	11	WingNut Films
11	12	New Line Cinema
12	13	Universal Studios
13	14	Miramax Films
14	18	Gracie Films
15	23	Imagine Entertainment
16	24	Mikona Productions GmbH & Co. KG

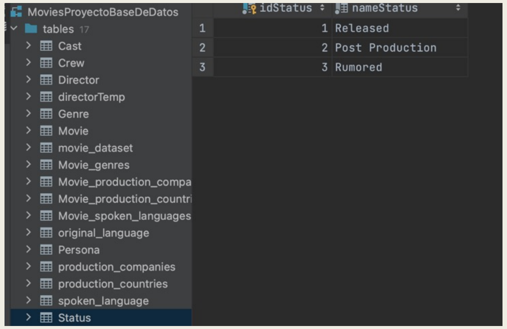
Población de tablas



	iso_3166_1	pCountryName
1	AE	United Arab Emirates
2	AF	Afghanistan
3	AO	Angola
4	AR	Argentina
5	AT	Austria
6	AU	Australia
7	AW	Aruba
8	BA	Bosnia and Herzegovina
9	BE	Belgium
10	BG	Bulgaria
11	BO	Bolivia
12	BR	Brazil
13	BS	Bahamas
14	BT	Bhutan
15	CA	Canada
16	CH	Switzerland



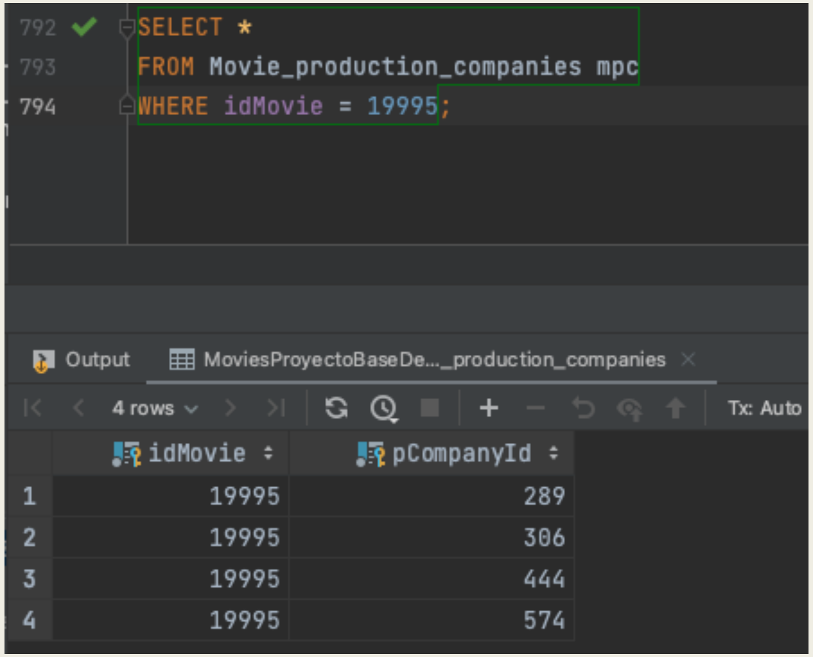
	iso_639_1	nameSLang
1	af	Afrikaans
2	am	
3	ar	العربية
4	bg	български език
5	bm	Bamanankan
6	bn	বাংলা
7	bo	
8	br	
9	bs	Bosanski
10	ca	Català
11	ce	
12	cn	广州话 / 廣州話
13	co	
14	cs	Český
15	cy	Cymraeg
16	da	Dansk



	idStatus	nameStatus
1	1	Released
2	2	Post Production
3	3	Rumored

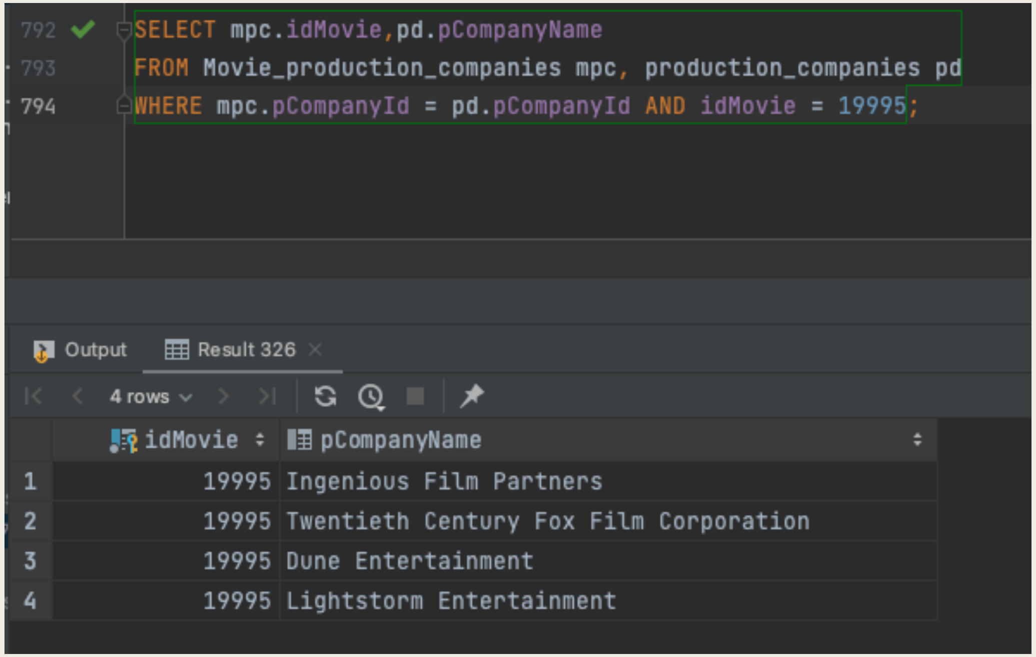
CONSULTAS

CONSULTAS



```sql
SELECT *
FROM Movie_production_companies mpc
WHERE idMovie = 19995;
```

	idMovie	pCompanyId
1	19995	289
2	19995	306
3	19995	444
4	19995	574



```sql
SELECT mpc.idMovie,pd.pCompanyName
FROM Movie_production_companies mpc, production_companies pd
WHERE mpc.pCompanyId = pd.pCompanyId AND idMovie = 19995;
```

	idMovie	pCompanyName
1	19995	Ingenious Film Partners
2	19995	Twentieth Century Fox Film Corporation
3	19995	Dune Entertainment
4	19995	Lightstorm Entertainment