

COMP1204 - Coursework 2

Pavel Stoyanov

ID 31163084

1 Relational Model

1.1 EX1

The Relation has the following attributes:

- dateRep(DATE)
- day(tinyint)
- month(tinyint)
- year(YEAR)
- cases(int)
- deaths(int)
- countriesAndTerritories(VARCHAR(35))
- geoId(VARCHAR(2))
- countryterritoryCode(VARCHAR(3))
- popData2018(int)
- continentExp(VARCHAR(7))

1.2 EX2

countriesAndTerritories, countryTerritoryCode, geoId and popData2018 are interchangeable as they are all unique. List of Functional Dependencies

- dateRep \rightarrow day,month,year
- month \rightarrow year
- countryTerritoryCode \rightarrow geoId,countriesAndTerritories,popData2018,continentExp

- countriesAndTerritories \rightarrow geoId,countryTerritoryCode,popData2018,continentExp
- geoId \rightarrow coutryTerritoryCode,countriesAndTerritories,popData2018,continentExp
- popData2018 \rightarrow coutryTerritoryCode,geoId,countriesAndTerritories,continentExp
- dateRep,countriesAndTerritories \rightarrow cases,deaths

1.3 EX3

The potential keys are the combinations between dateRep and either of geoId, countriesAndTerritories, popData2018 and countryTerritoryCode because with them you can get to the rest attributes. However popData2018 and countryTerritoryCode are null in some entries which leaves geoId and countriesAndTerritories.

1.4 EX4

Primary key should be {dateRep,geoId} as they are the attributes taking the least amount of space which are not null.

2 Normalisation

2.1 EX5

The partial key dependencies are

- dateRep \rightarrow day,month,year
- geoId \rightarrow coutryTerritoryCode,countriesAndTerritories,popData2018,continentExp

2.2 EX6

We can decouple with 2 new Relations Relation1 based on the Date with the following attributes decoupled from the main Relation leaving only the determinant a.k.a. dateRep there :

- dateRep(DATE) serving as primary key to the Relation as it is unique for every entry and determinant for the other attributes
- day(tinyint)
- month(tinyint)
- year(YEAR)

Relation2 based on country with the following attributes decoupled from the main Relation leaving only the determinant a.k.a. geoId there :

- geoId(VARCHAR(2)) serving as primary key to the Relation as it is unique for every entry not null and determinant for the other attributes
- countriesAndTerritories(VARCHAR(35))
- countryTerritoryCode(VARCHAR(3))
- popData2018(int)
- continentExp(VARCHAR(7))

2.3 EX7

We have transitive dependency month to year.

2.4 EX8

We decouple Relation1.1 from Relation1 with the following attributes leaving only the determinant a.k.a. month there :

- month(tinyint)
- year(YEAR)

2.5 EX9

It is BCNF because every candidate key determines the whole row in every relation.

3 Modelling

3.1 EX10

Using the following lines to import and dump the database

```
.mode csv
.import dataset.csv dataset
.once dataset.sql
.dump
```

3.2 EX11

Creating the schema with ".read" and ex11.sql :

```
BEGIN TRANSACTION;
CREATE TABLE monthYear(
  "month" tinyint PRIMARY KEY,
  "year" YEAR
);
```

```

CREATE TABLE dates(
  "dateRep" DATE PRIMARY KEY,
  "day" tinyint ,
  "month" tinyint ,
  FOREIGN KEY (month)
  REFERENCES "monthYear" (month)
  ON DELETE CASCADE
  ON UPDATE NO ACTION
);
CREATE TABLE countries(
  "countriesAndTerritories" VARCHAR(35),
  "geoId" VARCHAR(2) PRIMARY KEY,
  "countryterritoryCode" VARCHAR(3),
  "popData2018" int ,
  "continentExp" VARCHAR(10)
);
CREATE TABLE casesbydate(
  "dateRep" DATE,
  "cases" int ,
  "deaths" int ,
  "geoId" VARCHAR(2),
  PRIMARY KEY (dateRep,geoId),
  FOREIGN KEY (dateRep)
  REFERENCES "dates"(dateRep)
  ON DELETE CASCADE
  ON UPDATE NO ACTION,
  FOREIGN KEY (geoId)
  REFERENCES "countries"(geoId)
  ON DELETE CASCADE
  ON UPDATE NO ACTION

);
COMMIT;

```

Then dumping into dataset2.sql

3.3 EX12

```

BEGIN TRANSACTION;
INSERT INTO monthYear SELECT DISTINCT month,year FROM dataset;
INSERT INTO dates SELECT DISTINCT dateRep,day,month FROM dataset;
INSERT INTO countries SELECT DISTINCT countriesAndTerritories,geoId,coun
INSERT INTO casesbydate SELECT dateRep,cases,deaths,geoId FROM dataset;
COMMIT;

```

Then dumping into dataset3.sql

3.4 EX13

After testing on fresh database

4 Querying

4.1 EX14

```
SELECT SUM( cases ),SUM( deaths )
FROM casesbydate;
```

4.2 EX15

```
SELECT dateRep , cases
FROM casesbydate
WHERE geoId LIKE "UK"
ORDER BY dateRep ASC;
```

4.3 EX16

```
SELECT M.dateRep , SUM(M.cases) ,SUM(M.deaths) ,C.continentExp
FROM casesbydate AS M, countries AS C
GROUP BY C.continentExp
ORDER BY M.dateRep ASC;
```

4.4 EX17

```
SELECT countries.countriesAndTerritories ,
(1.0*SUM(casesbydate.cases))/countries.popData2018 ,
(1.0*SUM(casesbydate.deaths))/countries.popData2018
FROM countries , casesbydate
GROUP BY countries.geoId;
```

4.5 EX18

```
SELECT countries.countriesAndTerritories ,
(1.0*SUM(casesbydate.deaths))/(1.0*SUM(casesbydate.cases)) as perc
FROM countries ,casesbydate
GROUP BY countries.geoId
ORDER BY perc DESC
LIMIT 10;
```