Using sqlite to manage databases

Piers Jonckers, pj2g19, 31081347

May 2020

1   1. The Relational Model

1.1   EX1

dataSet(dateRep(date), day(int), month(int), year(int), cases(int), deaths(int),
countriesAndTerritories(varchar(255)), geoId(varchar(255)), countryterritoryCode(varchar(3)),
popData2018(int), continentExp(varchar(255)))

1.2   EX2

I can't use the arrow symbol in latex, so I will replace it with "points to". I also
couldn't use the squiggly brackets in latex.
I am making the assumption that popData2018 will be updated with new popu-
lation data, meaning in the future two countries may have the same population.
I am also assuming that no countries share a name.
I created a new attribute in order to differentiate the country codes, geoId and
population counts from other countries as some of them are unknown.
dateRep points to day, month, year
day
month
year
cases
deaths
countriesAndTerritories points to geoId, dataKey, countryterritoryCode, pop-
Data2018, continentExp
geoId points to countriesAndTerritories, dataKey, countryterritoryCode, pop-
Data2018, continentExp (one territory has N/A as its geoId, so geoId shouldnt
be relied on)
dataKey points to countryterritoryCode and popData2018
countryterritoryCode
popData2018
continentExp

1.3 EX3

dateRep, countriesAndTerritories
dateRep, dataKey
day, month, year, countriesAndTerritories
day, month, year, dataKey

1.4 EX4

dateRep, countriesAndTerritories
I will use this composite key as a primary key as it allows for all of the columns
to be accessed as the country name is unique per country and the date only
occurs once for each country.
I cannot use the country code or geoId as there are some territories which do
not have one.

2 2. Normalization

2.1 EX5

There are two partial dependencies, dataKey and continentExp rely only on
countriesAndTerritories.
day, moth and year all rely on dateRep.

2.2 EX6

casesAndDeaths

dateRep	countriesAndTerritories	cases	deaths
[key]date	[key]varchar(255)	int	int

date

dateRep	day	month	year
[key]date	int	int	int

countries

countriesAndTerritories	geoId	dataKey	countryterritoryCode	popData2018	continentExp
[key]varchar(255)	varchar(255)	int	varchar(3)	int	varchar(255)

2.3   EX7

geoId, countryterritoryCode and popData2018 rely on dataKey.

2.4   EX8

Remove geoId, countryterritoryCode and popData2018 from the last relation and include a new relation.

countryData

dataKey	geoId	countryterritoryCode	popData2018
[key]int	varchar(255)	varchar(3)	int

2.5   EX9

Yes, neither of the parts of the parts of the candidate keys are dependent on any other attributes.

3   3. Modelling

3.1   EX10

dataset.sql is included in the tar.gz.

3.2   EX11

I have made countriesAndTerritories the primary key of countries
dateRep is the primary key of date
dataKey is the primary key of countryData
countries includes dataKey as a foreign key
casesAndDeaths includes countriesAndTerritories and dateRep as foreign keys
ex11.sql is included in the tar.gz.

3.3   EX12

ex12.sql is included in the tar.gz.

3.4   EX13

It is possible to read dataset.sql, followed by ex11.sql and ex12.sql to get the complete database.

4 4. Querying

4.1 EX14

ex14.sql is included.

4.2 EX15

ex15.sql is included.

4.3 EX16

ex16.sql is inlucded.

4.4 EX17

ex17.sql is included.

4.5 EX18

ex18.sql is included.

4.6 EX19

ex19.sql is included.
Uses window functions.