Question 1:

* 1. Create table first in sql 🡪 Country

-- Table: public.Country\_data

-- DROP TABLE public."Country\_data";

CREATE TABLE IF NOT EXISTS public."Country\_data"

(

country character varying COLLATE pg\_catalog."default",

images\_file character varying COLLATE pg\_catalog."default",

image\_url character varying COLLATE pg\_catalog."default",

alpha\_2 character varying COLLATE pg\_catalog."default",

alpha\_3 character varying COLLATE pg\_catalog."default",

country\_code integer,

iso\_3166\_2 character varying COLLATE pg\_catalog."default",

region character varying COLLATE pg\_catalog."default",

sub\_region character varying COLLATE pg\_catalog."default",

intermediate\_region character varying COLLATE pg\_catalog."default",

region\_code integer,

sub\_region\_code integer,

intermediate\_region\_code integer

)

TABLESPACE pg\_default;

ALTER TABLE public."Country\_data"

OWNER to postgres;

* 1. Create table structure of Happiness

-- Table: public.Happiness\_tbl

-- DROP TABLE public."Happiness\_tbl";

CREATE TABLE IF NOT EXISTS public."Happiness\_tbl"

(

"Year" character varying COLLATE pg\_catalog."default",

"Country\_region" character varying COLLATE pg\_catalog."default",

"Happiness\_Score" character varying COLLATE pg\_catalog."default",

"GDP\_Per\_Capital" character varying COLLATE pg\_catalog."default",

"Family" character varying COLLATE pg\_catalog."default",

"Social\_Support" character varying COLLATE pg\_catalog."default",

"Healthy\_life\_Expectency" character varying COLLATE pg\_catalog."default",

"Freedom\_to\_make\_life" character varying COLLATE pg\_catalog."default",

"Generosity" character varying COLLATE pg\_catalog."default",

"Perception\_of\_Corruption" character varying COLLATE pg\_catalog."default"

)

TABLESPACE pg\_default;

ALTER TABLE public."Happiness\_tbl"

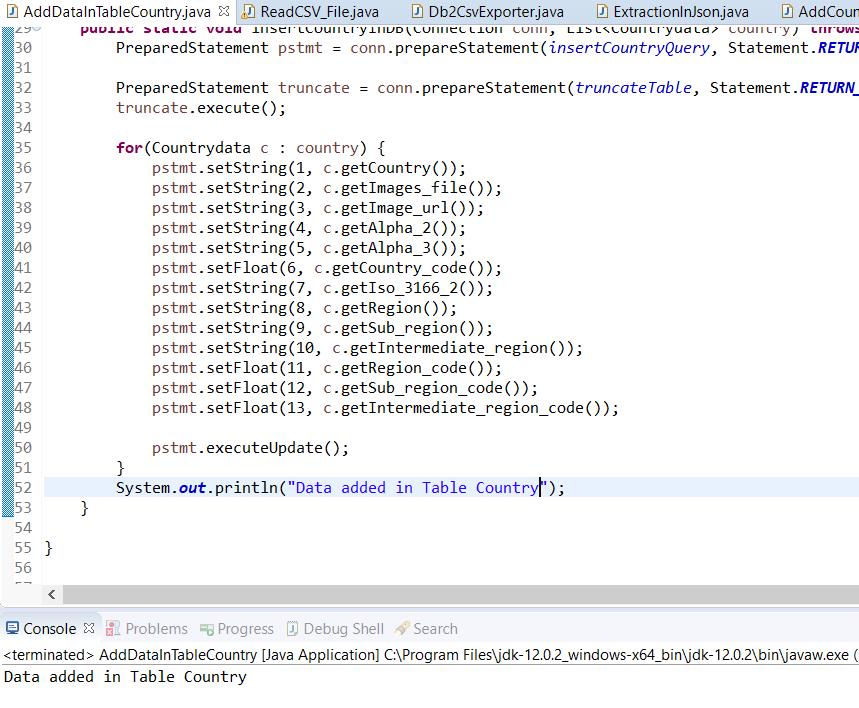
OWNER to postgres;

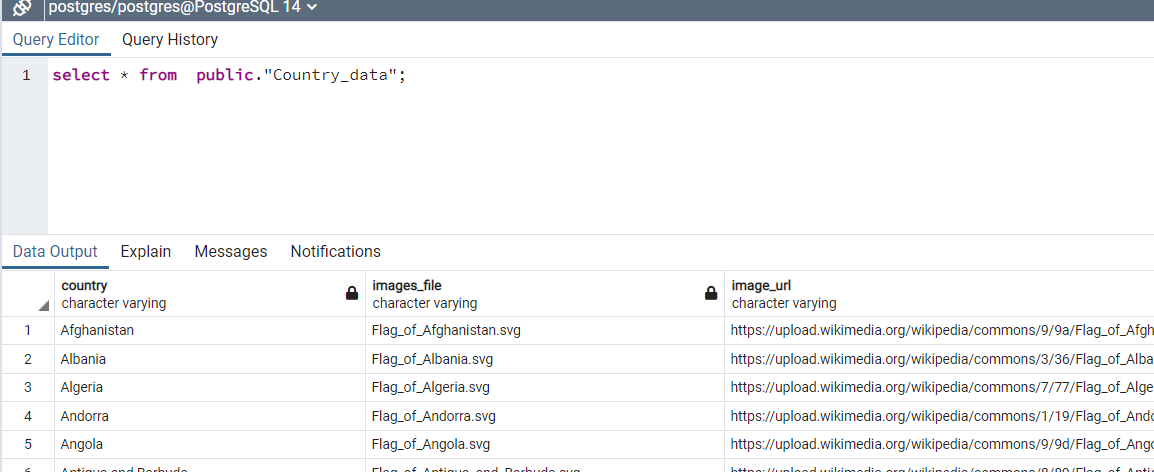
2. Question 2

Insert data in table Country:

Class 🡪 AddDataInTableCountry

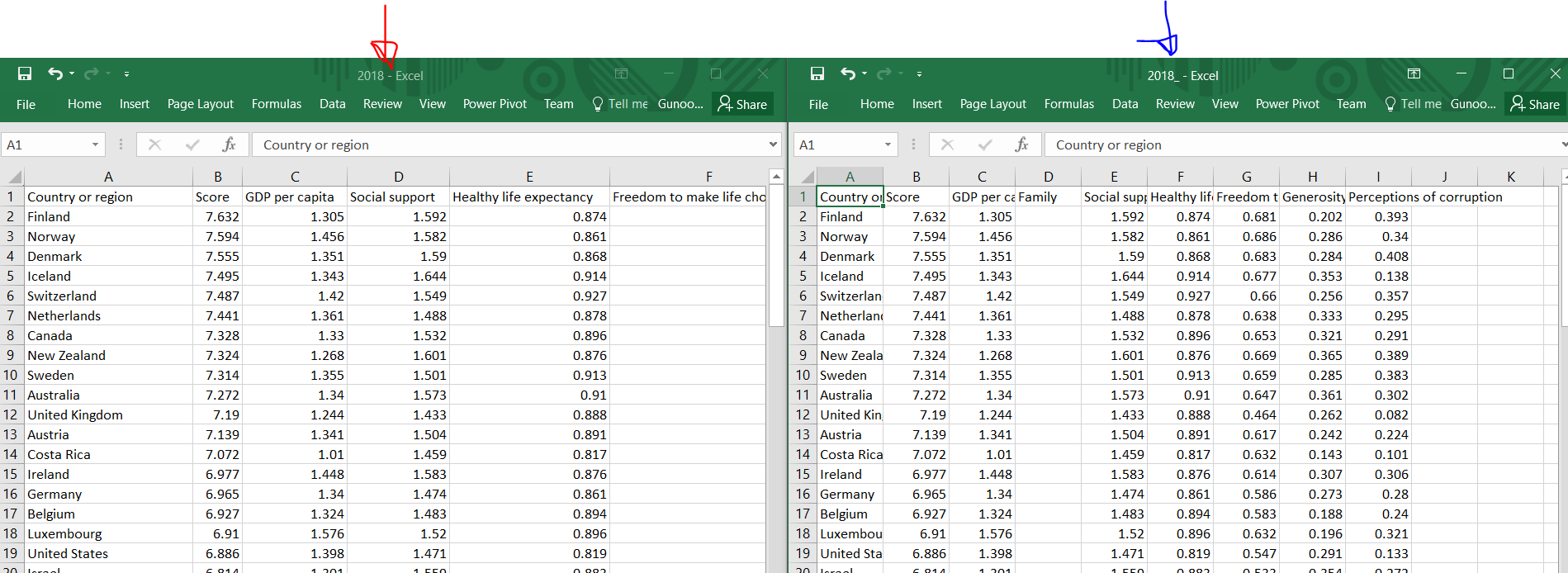
Each time this class runs, it will truncate the table first then add data to prevent duplicate





2.2 Adding Happiness Data in table

First data in raw excel need to be formatted in a proper way due to naming convention used.

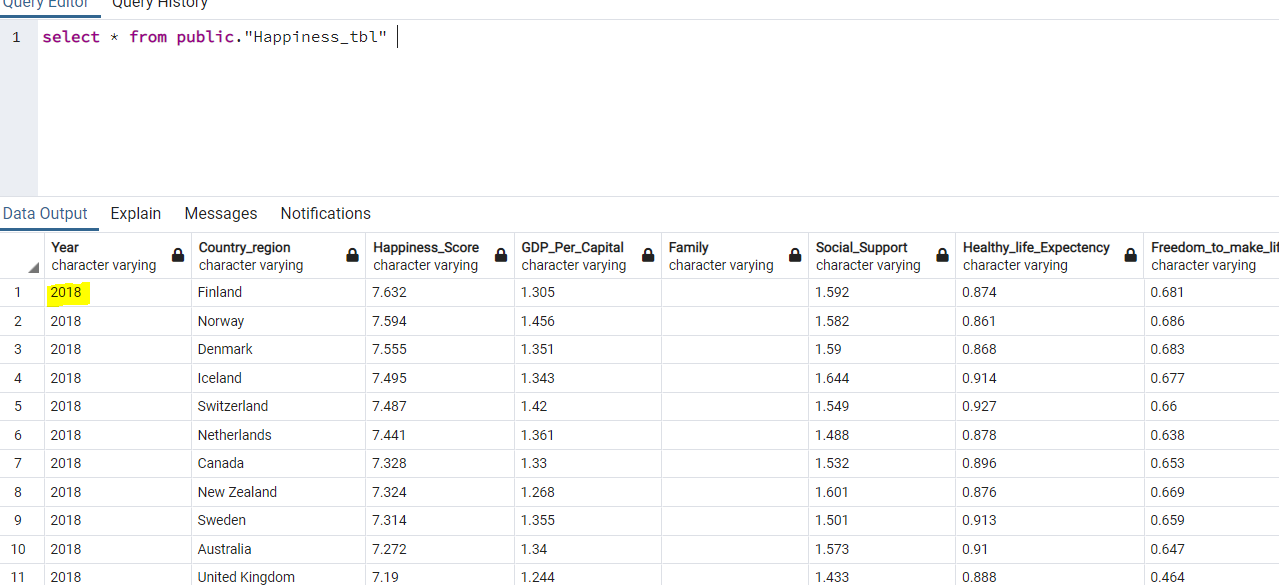


Arrow in red is raw files and arrow in Blue is processed file

Class responsible to load data in Table 🡪 ReadCSV\_File



This class is responsible to add data in table Happiness. If data for the year is already present in table first it will delete those data for that particular year then reinsert.



Question 3

3.1 Extraction of data in CSV files

Query 🡪

select

s."Year",

c.country as Country,

c.image\_url as Country\_Url,

c.region\_code as Region\_code,

case when c.region ='' then 'Nan'else (upper(c.region)) end as Region,

rank() over(

partition by s."Year" , c.region

order by cast(s."Happiness\_Score" as decimal) desc) as Rank\_Per\_Region,

rank() over(

partition by s."Year"

order by cast(s."Happiness\_Score" as decimal) desc) as Overall\_rank,

s."Happiness\_Score",

case

when cast(s."Happiness\_Score" as decimal) > 5.6 then'Green'

when cast(s."Happiness\_Score" as decimal) > 2.6 and cast(s."Happiness\_Score" as decimal)< 5.6 then 'Amber'

when cast(s."Happiness\_Score" as decimal) < 2.6 then 'Red'

end

as "Happiness\_Status",

"GDP\_Per\_Capital",

"Family",

"Social\_Support",

"Healthy\_life\_Expectency",

"Freedom\_to\_make\_life",

"Generosity",

"Perception\_of\_Corruption"

FROM public."Happiness\_tbl" s

left join public."Country\_data" c

on c.country = s."Country\_region"

where

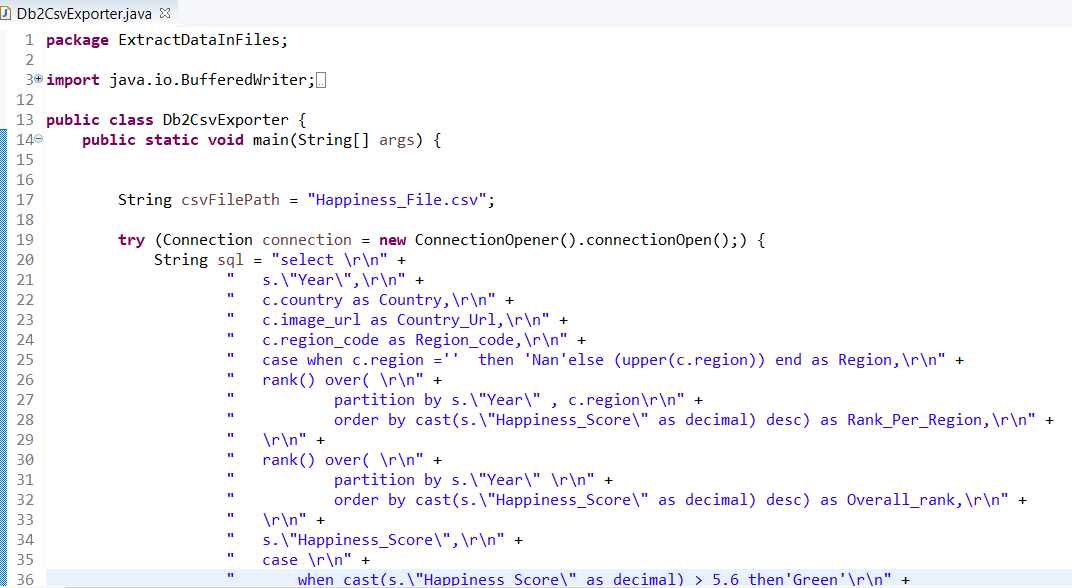
c.country is not null

order by

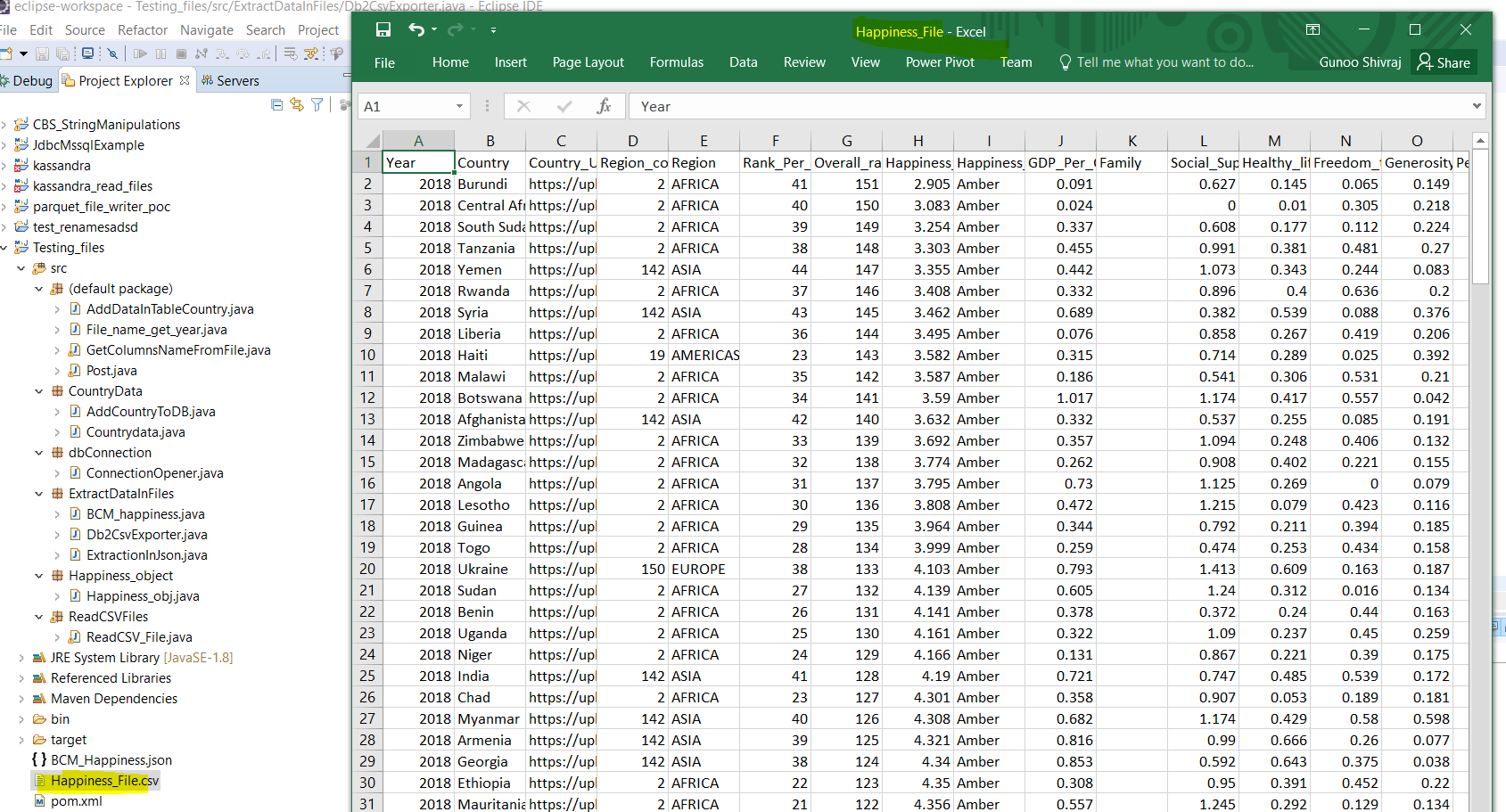
"Happiness\_Status",

rank() over( order by cast(s."Happiness\_Score" as decimal) asc )

Class for extraction of data 🡪 ReadCSV\_File.java



After running the code🡪



Question 4

4.1 🡪 Query:

with rank\_ as (

select

s."Year",

c.country as Country,

c.image\_url as Country\_Url,

c.region\_code as Region\_code,

case when c.region ='' then 'Nan'else (upper(c.region)) end as Region,

rank() over(

partition by s."Year" , c.region

order by cast(s."Happiness\_Score" as decimal) desc) as Rank\_Per\_Region,

rank() over(

partition by s."Year"

order by cast(s."Happiness\_Score" as decimal) desc) as Overall\_rank,

cast(s."Happiness\_Score" as decimal) as Happiness\_Score,

case

when cast(s."Happiness\_Score" as decimal) > 5.6 then'Green'

when cast(s."Happiness\_Score" as decimal) > 2.6 and cast(s."Happiness\_Score" as decimal)< 5.6 then 'Amber'

when cast(s."Happiness\_Score" as decimal) < 2.6 then 'Red'

end

as "Happiness\_Status",

"GDP\_Per\_Capital",

"Family",

"Social\_Support",

"Healthy\_life\_Expectency",

"Freedom\_to\_make\_life",

"Generosity",

"Perception\_of\_Corruption"

FROM public."Happiness\_tbl" s

left join public."Country\_data" c

on c.country = s."Country\_region"

where

c.country is not null

order by

"Happiness\_Status",

rank() over( order by cast(s."Happiness\_Score" as decimal) asc )

)

select

r.country,

max(r.Overall\_rank) as Highest\_Rank,

min(r.Overall\_rank) as Lowest\_Rank,

max(r.Happiness\_Score) as Highest\_Happines\_score,

min(r.Happiness\_Score) as Lowest\_Happiness\_score

from rank\_ r

group by

r.country

Code 🡪 ExtractionInJson.java



