CS 306

PROJECT STEP 3 by GROUP HAVUCLAR

https://github.com/tansylu/Havuclar-CS306

- Ada Yılmaz 29364
- Alım Gürbüz 28954
- Fatih Mutlu 27997
- Tansylu Akhmetova 30517
- Yusuf Erkam Köksal 29512

PART 1_ Discover Insights:
Five different views of a dataset were created for the homework assignment. The views and their explanations are as follows:

View One

This view displays the row with the lowest internet users and earliest year per country and the corresponding electricity access, urban population, and GDP values. For each country, the row in which internet users first appear is found, and the urbanization and electricity access for that row are displayed. By sorting the results descending by year, the countries that were late to adopt the internet can be seen.

```
CREATE
    ALGORITHM = UNDEFINED
    DEFINER = `root`@`localhost`
    SQL SECURITY DEFINER
VIEW `stats_for_lowest_internet_usage_year_by_country` AS
    SELECT
        `i`.`Code` AS `Code`,
        `i`.`Country_name` AS `Country_name`,
        `i`.`Internet_users` AS `Internet_users`,
        `i`.`Year` AS `min_internet_users_year`,
        `i`.`Urban_population` AS `Urban_population`,
        `i`.`Electricity_access_urban` AS `Electricity_access_urban`,
        `g`.`GDP` AS `GDP`
    FROM
```

```
        (`infrastructure` `i`
    JOIN `government` `g` ON (((`i`.`Code` = `g`.`Code`)
        AND (`i`.`Year` = `g`.`Year`))))
    WHERE
        (`i`.`Internet_users` = (SELECT
                MIN(`infrastructure`.`Internet_users`)
            FROM
                `infrastructure`
            WHERE
                (`infrastructure`.`Code` = `i`.`Code` AND
`infrastructure`.`Internet_users`>0)))
    ORDER BY `i`.`Year` DESC
```

View Two

This view displays the row with the highest internet users and most recent year per country and the corresponding electricity access, urban population, and GDP values. For each country, the row in which internet users most recently appear is found, and the urbanization and electricity access for that row are displayed. By sorting the results descending by year, the latest situation in the world can be seen.

```
CREATE
    ALGORITHM = UNDEFINED
    DEFINER = `root`@`localhost`
    SQL SECURITY DEFINER
VIEW `step3`.`stats_for_highest_internet_usage_year_by_country` AS
    SELECT
        `i`.`Code` AS `Code`,
        `i`.`Country_name` AS `Country_name`,
        `i`.`Internet_users` AS `Internet_users`,
        `i`.`Year` AS `max_internet_users_year`,
        `i`.`Urban_population` AS `Urban_population`,
        `i`.`Electricity_access_urban` AS `Electricity_access_urban`,
        `g`.`GDP` AS `GDP`
    FROM
        (`step3`.`infrastructure` `i`
        JOIN `step3`.`government` `g` ON (((`i`.`Code` = `g`.`Code`)
            AND (`i`.`Year` = `g`.`Year`))))
    WHERE
```

```
        ((`i`.`Internet_users` , `i`.`Year`) = (SELECT
                MAX(`step3`.`infrastructure`.`Internet_users`),
                    MAX(`step3`.`infrastructure`.`Year`)
            FROM
                `step3`.`infrastructure`
            WHERE
                (`step3`.`infrastructure`.`Code` = `i`.`Code`)))
    ORDER BY `i`.`Year` DESC
```

View Three

This view selects countries with a freedom and democracy estimate higher than average, then selects the latest year for which there is data and displays it in a sorted way by democracy field.

```
CREATE
VIEW step3.most_free_and_democratic_countries AS
    SELECT
        p.Code AS Code,
        p.Country_name AS Country_name,
        p.Year AS Year,
        p.Democracy_estimate AS Democracy_estimate,
        p.Freedom_estimate AS Freedom_estimate
    FROM
        step3.policy p
    WHERE
        ((p.Year = (SELECT
                MAX(p2.Year)
            FROM
                step3.policy p2
            WHERE
                (p2.Code = p.Code AND p2.Year <= 2020)))
            AND (p.Democracy_estimate > (SELECT
                AVG(p3.Democracy_estimate)
            FROM
                step3.policy p3))
            AND (p.Freedom_estimate > (SELECT
                AVG(p4.Freedom_estimate)
            FROM
                step3.policy p4)))
```

```
        GROUP BY p.Code , p.Year

        ORDER BY p.Freedom_estimate DESC , p.Democracy_estimate DESC
```

View Four

This view pulls the landline internet subscriptions and mobile cellular subscriptions numbers for the rows in view three to see the most recent data for the most democratic countries.

```
CREATE

VIEW step3.democracy_vs_subscriptions AS

    SELECT

        m.Code AS Code,

        m.Country_name AS Country_name,

        m.Year AS Year,

        m.Mobile_cellular_subscriptions AS
Mobile_cellular_subscriptions,

        m.Landline_internet_subscriptions AS
Landline_internet_subscriptions,

        step3.fdc.Freedom_estimate AS Freedom_estimate,

        step3.fdc.Democracy_estimate AS Democracy_estimate

    FROM

        (step3.isp m

        JOIN step3.most_free_and_democratic_countries fdc ON
(((m.Code = step3.fdc.Code)

            AND (m.Year = step3.fdc.Year))))

    ORDER BY m.Landline_internet_subscriptions DESC ,
step3.fdc.Freedom_estimate DESC , step3.fdc.Democracy_estimate DESC
```

View Five

This is a view from the community table that compares mobile subscriptions with internet users number for each country.This view calculates the ratio of mobile cellular subscriptions to the number of internet users for each country and year, and orders the results by country and year. This view allows us to see the countries in which people mostly prefer mobile, down to countries in which internet landline is more popular than cellular.

```
CREATE
VIEW step3.community_mobile_vs_internet AS
    SELECT
        step3.community.Code AS Code,
        step3.community.Country_name AS Country_name,
        step3.community.Year AS Year,
        (step3.community.Mobile_cellular_subscriptions /
step3.community.Number_of_Internet_users) AS Mobile_vs_Internet
    FROM
        step3.community
    GROUP BY step3.community.Code , step3.community.Year
    ORDER BY Mobile_vs_Internet DESC
```

PART 1.b:

In this part I found the set difference of high democracy level countries and low urban population countries using the EXCEPT keyword. Then, I found the same result by using outer join.
This SQL query selects the country names and the corresponding maximum democracy level for each year from the `policy` table and joins with the infrastructure table to filter out high urban population countries.

```
-- finds the set difference of high democracy level countries and low
urban population countries
select Country_name, Democracy_estimate
from db.policy
where Year between 1990 and 2022 and Democracy_estimate >= 8.0
except
select Country_name, Urban_population
from db.infrastructure
where Year between 1990 and 2022 and Urban_population <= 10;


-- code below finds the same result using outer join
select p.Country_name, p.Democracy_estimate
from db.policy p
left join db.infrastructure i on p.Code = i.Code and p.Year = i.Year
where p.Year between 1990 and 2022 and p.Democracy_estimate >= 8.0
and (i.Urban_population is null or i.Urban_population <= 10);
```

PART 1.c: The following query *** using a nested select statement with the IN operator, and then the same code with the EXISTS operator. Both have the same results with the same number of rows which is 174.

```
-- using the IN operator:
select * from db.policy
where Code in (
```

```
  select Code from db.government
  where Year = 2022
) and Year = 2022;



-- using the EXISTS operator:
select * from db.policy p
where exists (
  select * from db.government g
  where p.Code = g.Code and g.Year = 2022
) and p.Year = 2022;
```

PART 1.d:

a) This query shows for how many years each country spent having freedom and democracy levels larger than 75 percent.

```
select countries.Country_name, count(*) AS Freedom_years from
db.policy p
join db.countries on p.Code = countries.Code
WHERE p.Freedom_estimate > 75 and p.Democracy_estimate > 75
GROUP BY p.Code
HAVING COUNT(*) > 0;
```

b)This query shows the internet arrival year and the gdp in that year for each country where the internet arrival year is older than average.

```
SELECT C.Country_name, C.Internet_arrival_year, G.GDP FROM
db.countries C, db.government G
WHERE C.Code = G.Code AND G.GDP IS NOT NULL AND G.Year =
C.Internet_arrival_year
GROUP BY G.GDP,G.Code, G.Year
HAVING C.Internet_arrival_year < (SELECT
AVG(C2.Internet_arrival_year) FROM db.countries C2);
```

c)This query shows the total landline phone subscriptions each year in descending order to show that landline subscriptions have increased until 2007 and then started decreasing.

```
select i.Year, sum(i.Landline_phone_subscriptions) as
total_subscriptions
from db.isp i
join db.countries c on i.Code = c.Code
group by i.Year
having total_subscriptions > 1000
order by total_subscriptions desc;
```

d)This query shows which country had the minimum urban electricity access each year, and the corresponding urban population of that country in that year.

```
select i.Year, c.Country_name, i.Urban_population,
i.Electricity_access_urban
from db.infrastructure i
join db.countries c on c.Code = i.Code
where i.Electricity_access_urban = (
    select min(Electricity_access_urban)
    from db.infrastructure
    where Year = i.Year
)
order by Year;
```

e)This query shows the countries' internet user and cellphone subscriber amount in 2020. I accidentally deleted the max keyword?

```
select c.Country_name, com.Number_of_Internet_users as
internet_users,
isp.Mobile_cellular_subscriptions as cell_subscriptions
from db.countries c
join(
  select Code, Number_of_Internet_users
  From db.community
  where Year = 2020
  group by Code, Number_of_Internet_users
) com on c.Code = com.Code
join (
  select Code, Mobile_cellular_subscriptions
  from db.isp
  where Year = 2020
  group by Code, Mobile_cellular_subscriptions
) isp on c.Code = isp.Code;
```

PART 2_ Constraints and Triggers:

In this part the chosen table and numeric field are community and internet access year respectively because they are the main focus of the database.

```
SELECT MIN(Internet_arrival_year) AS min_year, MAX(Internet_arrival_year) AS
max_year
FROM countries;
```

The minimum year is 1982 and the maximum is 2017, observed from this mySQL code.

```sql
-- Drop the check constraint if it already exists
ALTER TABLE `countries`
DROP CONSTRAINT IF EXISTS `chk_Internet_arrival_year`;

-- Add a new check constraint to the table
ALTER TABLE `countries`
ADD CONSTRAINT `chk_Internet_arrival_year`
CHECK (Internet_arrival_year >= 1982 AND Internet_arrival_year <= 2017);

-- Create the BEFORE INSERT trigger
DELIMITER //
CREATE TRIGGER `tr_countries_before_insert` BEFORE INSERT ON `countries` FOR
EACH ROW
BEGIN
    IF NEW.`Internet_arrival_year` < 1982 THEN
        SET NEW.`Internet_arrival_year` = 1982;
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'The Internet_arrival_year
value was fixed to the minimum value of 1982';
    ELSEIF NEW.`Internet_arrival_year` > 2017 THEN
        SET NEW.`Internet_arrival_year` = 2017;
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'The Internet_arrival_year
value was fixed to the maximum value of 2017';
    END IF;
END //
DELIMITER ;

-- Create the BEFORE UPDATE trigger
DELIMITER //
CREATE TRIGGER `tr_countries_before_update` BEFORE UPDATE ON `countries` FOR
EACH ROW
BEGIN
    IF NEW.`Internet_arrival_year` < 1982 THEN
        SET NEW.`Internet_arrival_year` = 1982;
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'The Internet_arrival_year
value was fixed to the minimum value of 1982';
    ELSEIF NEW.`Internet_arrival_year` > 2017 THEN
        SET NEW.`Internet_arrival_year` = 2017;
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'The Internet_arrival_year
value was fixed to the maximum value of 2017';
    END IF;
END //
DELIMITER ;
```

PART 3_ Stored Procedure:

In this part, we have two different procedures called "findInternetUsersByIsoAndYear" and "findInternetArrivalByIso". In the first procedure "findInternetUsersByIsoAndYear", the procedure takes two parameters called iso (string type) and ye (integer type) that correspond to Code and Year columns in datasets. Then the procedure returns the number of internet users from the internet_Users column according to the inputs.

```
DELIMITER //
create procedure findInternetUsersByIsoAndYear(in iso VARCHAR(8), in ye INT)
begin
      select Internet_users, Country_name, Code from infrastructure where Code
= iso and Year = ye;
end //
DELIMITER //
```

In the second procedure called "findInternetArrivalByIso", the procedure takes one parameter called iso (string type) that corresponds to the Code column in the dataset. Then the procedure returns the internet arrival year with given iso input.

```
DELIMITER //
create procedure findInternetArrivalByIso(in iso VARCHAR(8))
begin
      select * from countries where Code = iso;
end //
DELIMITER //
```