



Αποθήκες Δεδομένων
Μέρας Γεώργιος
gmeras@uth.gr

Εργασία Εαρινού εξαμήνου 2025

Θέμα:

**Η Μόλυνση παγκοσμίως. Ενεργειακή απόδοση σε σχέση ΑΕΠ
και κατανάλωσης από το 1990 έως 2023.**

Περιεχόμενα

1. Εύρεση και κατανόηση του χώρου εφαρμογής της Αποθήκης Δεδομένων.....	2
2. Εύρεση και κατανόηση των δεδομένων που θα αποθηκευτούν στην Αποθήκη Δεδομένων.....	2
3. Σχεδιασμός Αποθήκης Δεδομένων.....	2
4. Διαδικασία ELT (Extract, Transform, Load).....	5
4.1 Extract.....	5
4.2 Transform.....	5
4.3 Load.....	6
4.4 Πλήρης Κώδικας ETL.....	6
5. Ερωτήματα-Οπτικοποίηση.....	7
1) Ποία είναι η χώρα με τους περισσότερους ρύπους μεθανίου και διοξειδίου του άνθρακα.....	7
2) Ποια χώρα είναι η πιο ενεργειακά αποδοτική.....	9
3) Ποιος ο παράγοντας με τους περισσότερους ρύπους.....	10
4) Ποια είναι η πιο ρυπογόνα χώρα τα τελευταία 30 χρόνια και ποιο το ποσό των ρύπων;.....	11
5) Ποια η ενεργειακή απόδοση σε σχέση με τους ρύπους τη χρονιά 2022 (ένταση ρύπων).....	11
6) Οι πιο μολυσμένες χώρες ανα κάτοικο.....	12
7) Ποια Ήπειρος είναι η πιο ρυπογόνα από το 1990-2023.....	13
8) Οι πιο μολυσματικές χώρες ανα ήπειρο σε αναλογία με τη συνολική κατανάλωσή τους.....	13
9) Η μέση ενεργειακή παραγωγή ανανεώσιμων πηγών, από το 1990-2023 της Ευρώπης.....	14
10) Η χώρα με την μεγαλύτερη ανάγκη στην Ευρώπη από το 1990-2023.....	15
11) Ποσοστό ανανεώσιμης ενέργειας επί της συνολικής ανάγκης των Ευρωπαϊκών χωρών.....	15
6. Συμπεράσματα.....	16

1. Εύρεση και κατανόηση του χώρου εφαρμογής της Αποθήκης Δεδομένων

Η αποθήκη δεδομένων αναφέρεται στις εκπομπές ρύπων, διοξειδίου του άνθρακα, μεθανίου και μικροσωματιδίων PM2.5, 202 χωρών παγκοσμίως, από το 1990 έως το 2023. Επίσης συλλέξαμε δεδομένα οικονομικού τυπου, ώστε να δούμε πόσο ενεργειακά αποδοτικές είναι οι χώρες με αυξημένο ΑΕΠ, αλλά και πόσο ρυπογόνες επίσης. Συγκεκριμένα για την Ευρώπη κάναμε πιο λεπτομερή ανάλυση, με δεδομένα ανανεώσιμων πηγών ενέργειας, ορυκτών, αλλά και ανάγκης ποσού ενέργειας.

Για τον σκοπό αυτό συλλέξαμε 362.853 (εγγραφές της ΑΔ) δεδομένα από

World Bank Group: [Indicators | Data](#)

Ember: [Monthly Electricity Data | Ember](#)

Dataset Publishing Language: [countries.csv | Dataset Publishing Language | Google for Developers](#)

2. Εύρεση και κατανόηση των δεδομένων που θα αποθηκευτούν στην Αποθήκη Δεδομένων

Κύριος σκοπός της εργασίας και των δεδομένων είναι να αναδείξει την ένταση της ρύπανσης παγκοσμίως, ποιές χώρες είναι οι λιγότερο φιλικές για την ανθρώπινη υγεία, το ποσό των ρύπων κάθε χώρας και την τάξη μεγέθους αυτών. Θα δούμε επίσης πόσο αποδοτικές είναι οι χώρες σε αναλογία ΑΕΠ, ρύπων και κατανάλωσης ορυκτών.

3. Σχεδιασμός Αποθήκης Δεδομένων

Η ΑΔ υλοποιήθηκε στο MySQL Workbench σχεδιάστηκε με δύο Fact Tables, το πρώτο (fact) καταγράφει τις μετρήσεις των ρύπων-pollutant_value, μετρήσεις οικονομίας-economy_value, κατανάλωση ορυκτών-oil_kg. Οι διαστάσεις είναι ο χρόνος, χώρα, πηγές ΑΕΠ της κάθε χώρας, τύπος και πηγές μόλυνσης.

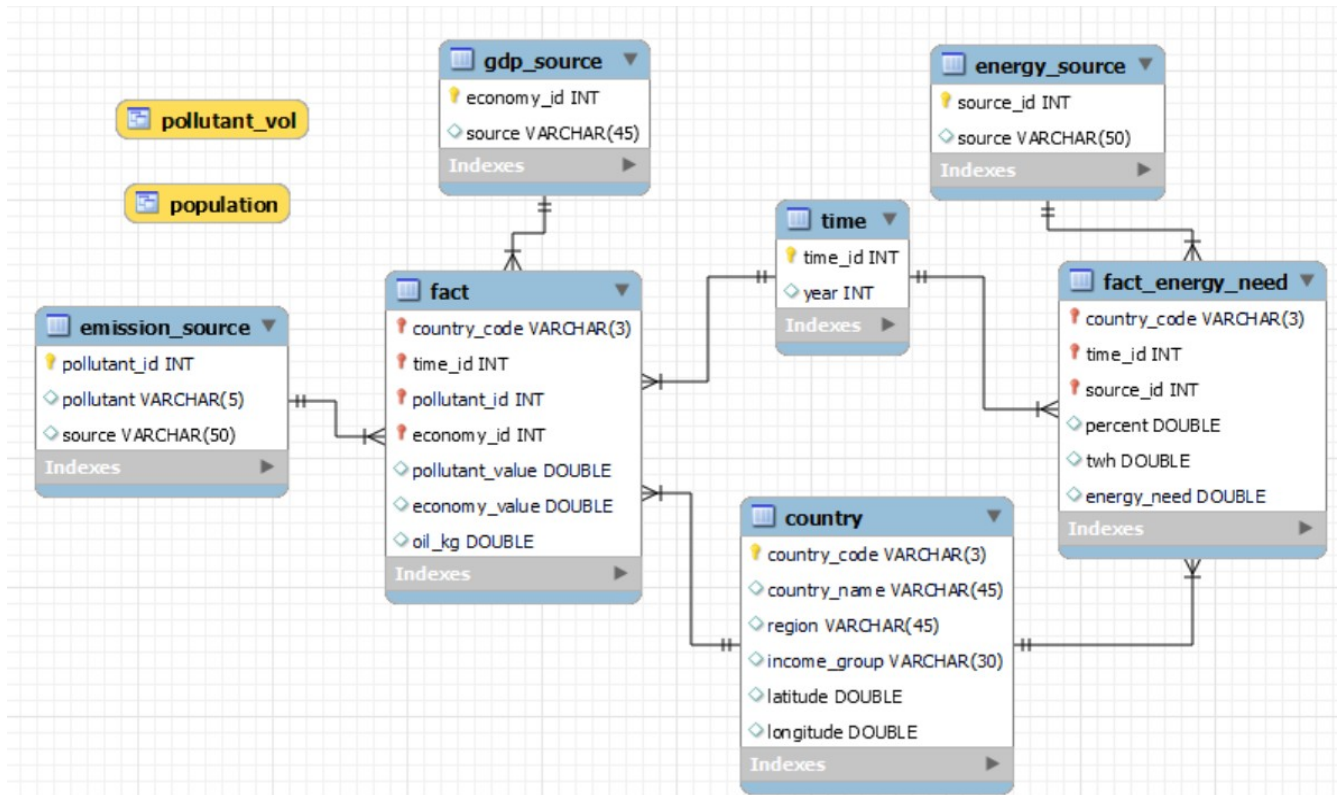
Ξένα κλειδιά του fact		
Column	Reference Table	Reference Column
country_code	country	country_code
time_id	time	time_id
pollutant_id	emission_source	pollutant_id
economy_id	economy	economy_id

Το δεύτερο Fact Table (fact_energy_need) καταγραφεί, μόνο για τις Ευρωπαϊκές χώρες, τις πηγές ενέργειας, το ποσοστό επί της συνολικής παραγώμενης ενέργειας percent, την τιμή σε Terawatt ώρες twh, και το ποσό ανάγκης της ενέργειας σε Megawatt ώρες energy_need. Οι διαστάσεις είναι η χώρα, ο χρόνος και πηγές παραγόμενης ενέργειας. Η ανάγκη ενέργειας εξαρτάται από μέρος του πρωτεύοντως κλειδιού country_code και time_id. Τα δύο fact tables τα χωρίσαμε καθώς το πρώτο περιέχει δεδομένα για 202 χώρες παγκοσμίως, ενώ το δεύτερο έχει περιοριστεί στις 35 τις Ευρώπης.

Ξένα κλειδιά του fact_energy_need		
Column	Reference Table	Reference Column
country_code	country	country_code
time_id	time	time_id
source_id	energy_source	source_id

Τελικά το σχήμα που κάναμε, ήταν ένας αστερισμός, με δύο fact tables και 5 διαστάσεις, εκ των οποίων, οι δύο ήταν κοινές (χρόνος, χώρα).

Τα δεδομένα που είχαμε, ηδικά από την οικονομία των χωρών, όπως κατακεφαλήν ΑΕΠ και ΑΕΠ, μας βοήθησαν να εξάγουμε τον αντίστοιχο πληθυσμό κάθε χώρας χωρίς να τον αποθηκεύσουμε ή να ψάξουμε περαιτέρω. Επομένως για τον πληθυσμό θα φτιάξουμε ένα View population. Αξίζει να σημειωθεί πως στο View που κατασκευάσαμε, κάναμε την αντιπαράβολή του πληθυσμού της Ελλάδας με δεδομένα της ΕΛΣΑΤ, όπου και βγαίνουν ίδια.



Σχήμα αστερισμού

Τα πρωτεύοντα κλειδιά σε όλες τις διαστάσεις εκτός απο το Country είναι auto increment, για το country χρησιμοποιήσαμε το country_code, αφού ήταν μοναδικό. Η εισαγωγή τους στον πίνακα fact και fact_energy_need έγινε με Python. Τα year_mapping, pollutant_mapping κλπ αποτέλεσαν τις τιμές για τους πίνακες διαστάσεων (έχει γραφεί αντιστοιχο script για τον fact_energy_need).

4. Διαδικασία ELT (Extract, Transform, Load)

Για τη διαδικασία ETL χρησιμοποιήσαμε Python, πιο συγκεκριμένα τη βιβλιοθήκη Pandas, με την οποία αντλήσαμε πληροφορίες οι οποίες μας ήταν χρήσιμες και τις αποθηκεύσαμε σε αρχεία CSV, με μορφή τη μορφή των πινάκων που είχαμε στο SQL, απο τα οποία έγινε η τελική φόρτωση στην αποθήκη δεδομένων. Χρειάστηκε να αναστρέψουμε τις γραμμές με τις στήλες των παραπάνω αρχείων, με κλειδιά τα οποία έπρεπε να μείνουν ανέπαφα, όπως (country_code, year). Κάποιες από τις μεθόδους που μας βοήθησαν είναι οι melt, merge, replace, factorize και άλλες. Ένω αρχικά αφαιρέσαμε δύο attributes τα οποία ήταν οι ανανεώσιμες πηγές ενέργειας και ενέργεια που χρειάζεται η κάθε χώρα, καθώς είχαμε δεδομένα για μόνο 35 χώρες, σε σχέση με τις 202 αρχικές, τελικά φορτώθηκαν σε ένα δεύτερο fact table.

4.1 Extract

Η εξαγωγή των δεδομένων έγινε απο τρεις διαφορετικές πηγές. Η πρώτη ήταν η World Bank Group, δεύτερη η Ember και τρίτη η Dataset Publishing Language. Όλα τα δεδομένα ήταν σε μορφή CSV. Συλλέξαμε περίπου 40 διαφορετικά αρχεία απο τις παραπάνω πηγές.

4.2 Transform

Για το μετασχηματισμό των δεδομένων χρησιμοποιήσαμε την Python, πιο συγκεκριμένα τη βιβλιοθήκη Pandas. Κατά τη διαδικασία αυτή μας βοήθησαν οι μέθοδοι melt, merge, replace, rinot και άλλες. Παρακάτω, παρατίθεται ένα κομμάτι από τον κώδικα, το οποίο είναι γραμμένο σε Jupyter Nootebook.

```
df=pd.read_excel('./clean_data/CO2/co2_logw_metakinhsewn_se_Megaton.xlsx')
df_material=pd.read_excel('./clean_data/CO2/ch4_from_waste_material.xlsx')
```

Τα παραπάνω αρχεία δεν είχαν την επιθυμητή μορφή, οπότε εφαρμόσαμε τη μέθοδο melt, ώστε να μετατρέψουμε το αρχείο, σε μια βολική μορφή.

```
df_material = df_material.melt(id_vars=['Country Name', 'Country Code'],var_name='Year')
df_pol = df.melt(id_vars=['Country Name', 'Country Code'],var_name='Year',)
```

Ενσωματώσαμε τα δύο αρχεία σε ένα DataFrame, με τη μέθοδο merge() και αλλάξαμε τα ονόματα των στηλών.

```
df_final = df_pol.merge(df_material, on = ['Country Name', 'Country Code','Year'], how = 'left')
df_final = df_final.rename(columns={'value_x':'Transport','value_y':'Waste_Material'})
```

Θα φτιάξουμε ένα πεδίο “pollutant_source”, το οποίο θα περιγράφει την πηγή της μόλυνσης, καθώς το pollutant_source αποτελεί διάσταση στον fact_table.

```
df_final = df_final.melt(id_vars=['Country Name', 'Country Code','Year'],var_name='pollutant_source')
```

Έπειτα, πρέπει να δημιουργήσουμε τα DataFrames για τις διαστάσεις. Αυτό θα υλοποιηθεί με τη μέθοδο `factorize()`, η οποία για κάθε μια τιμή στη στήλη που εφαρμόζεται, φτιάχνει ένα μοναδικό ακέραιο και αποθηκεύει τις τιμές που έχει αντικαταστήσει, σε ένα tuple. Παρακάτω φαίνεται η διαδικασία, για τη διάσταση-χρόνο. Το `year_map` περιέχει όλες τις χρονίες. Χρησιμοποιούμε `df_final['Year']+=1` ώστε ο δείκτης να ξεκινά από το 1 και όχι το 0.

```
df_final['Year'], year_map = df_final['Year'].factorize()
df_final['Year']+=1
```

Τελικά, δημιουργούμε τους πίνακες για τις διαστάσεις, αποθηκεύοντας αρχικά τα δεδομένα σε δυο DataFrames.

```
df_year = df_final['Year'].unique()
dict = {}
for year in df_year:
    dict[int(year)] = year_map[year-1]
year_table = pd.DataFrame.from_dict(dict,orient='index')
year_table = year_table.reset_index().rename(columns={'index':'id',0:'year'})
```

Ομοίως εργαζόμαστε για τη στήλη “`pollutant_source`”, αφού, θέλουμε να την κάνουμε διάσταση.

```
df_final['pollutant_source'], pollutant_map = df_final['pollutant_source'].factorize()
df_final['pollutant_source']+=1

dict_pollutant = {}
for pollutant in df_final['pollutant_source'].unique():
    dict_pollutant[int(pollutant)]=pollutant_map[pollutant-1]
pollutant_table = pd.DataFrame.from_dict(dict_pollutant,orient='index')
pollutant_table = pollutant_table.reset_index().rename(columns={'index':'id',0:'pollutant'})
```

4.3 Load

Η διαδικασία της φόρτωσης, έγινε με τη βιβλιοθήκη `sqlalchemy`. Παρακάτω, φαίνεται πως δημιουργούμε τη σύνδεση της βάσης, με την Python και αποθηκεύουμε στο sql.

```
from sqlalchemy import create_engine
engine = create_engine('mysql+mysqlconnector://user:pass@host:port/db')

pollutant_table.to_sql('pollutant_table', engine, if_exists='replace')
year_table.to_sql('year_table', engine, if_exists='replace')
df_final.to_sql('df_final', engine, if_exists='replace')
```

4.4 Πλήρης Κώδικας ETL

Ο πλήρης κώδικας είναι στο GitHub [εδώ](#)

5. Ερωτήματα-Οπτικοποίηση

Θα χρειαστούμε τον πληθυσμό της εκάστοτε χώρας, οπότε χρησιμοποιούμε το παρακάτω View.

```
create or replace view population as
select distinct(f.country_code),t.year,round(f.economy_value/cap.economy_value) as population
from fact f
join time t on f.year = t.id
join gdp_source g on f.economy_id = g.id
join (
select distinct(f.country_code),t.year,economy_value
from fact f
join time t on f.year = t.id
join gdp_source g on f.economy_id = g.id
where g.source in ('GDP per capita')) as cap on f.country_code = cap.country_code and t.year =
cap.year
where g.source in ('GDP');
```

1) Ποία είναι η χώρα με τους περισσότερους ρύπους μεθανίου και διοξειδίου του άνθρακα, απο βιομηχανία, βιομηχανία ενέργειας, μετακινήσεις, ΧΥΤΑ (απόβλητα) το 2022;

```
select c.country_code,c.country_name,sum(f.pollutant_value) as total_pollutant,c.latitude,c.longitude
from fact f
join emission_source es on f.pollutant_id = es.id
join time t on f.year=t.id
join country c on f.country_code=c.country_code
where es.pollutant in ('CO2','CH4') and es.source not in ('Per capita (Tons)','Smoke from natural
sources, cars, industry (Mt)') and t.year = '2022' and c.longitude is not null
group by c.country_code
order by total_pollutant desc;
```

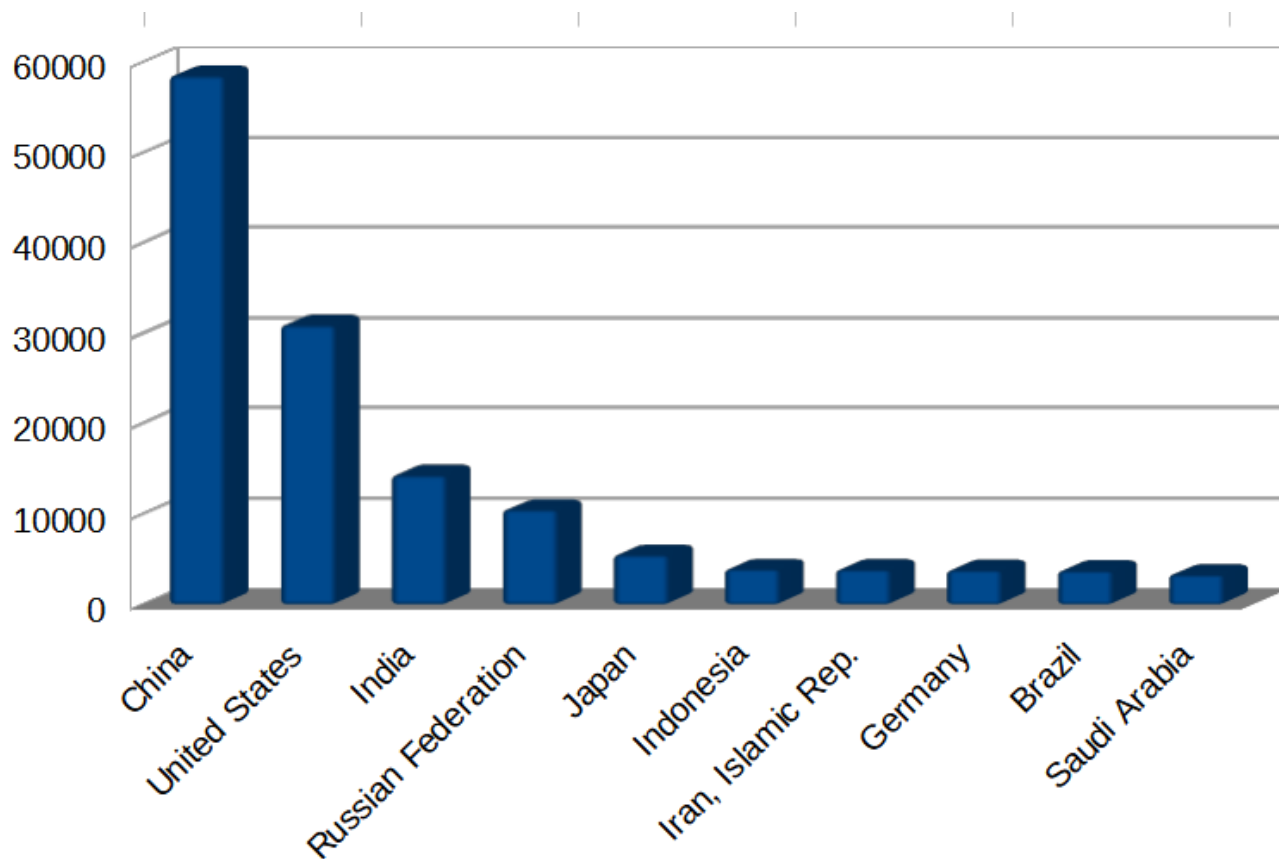


Figure 1: Οι 10 πιο ρυπογόνες χώρες.

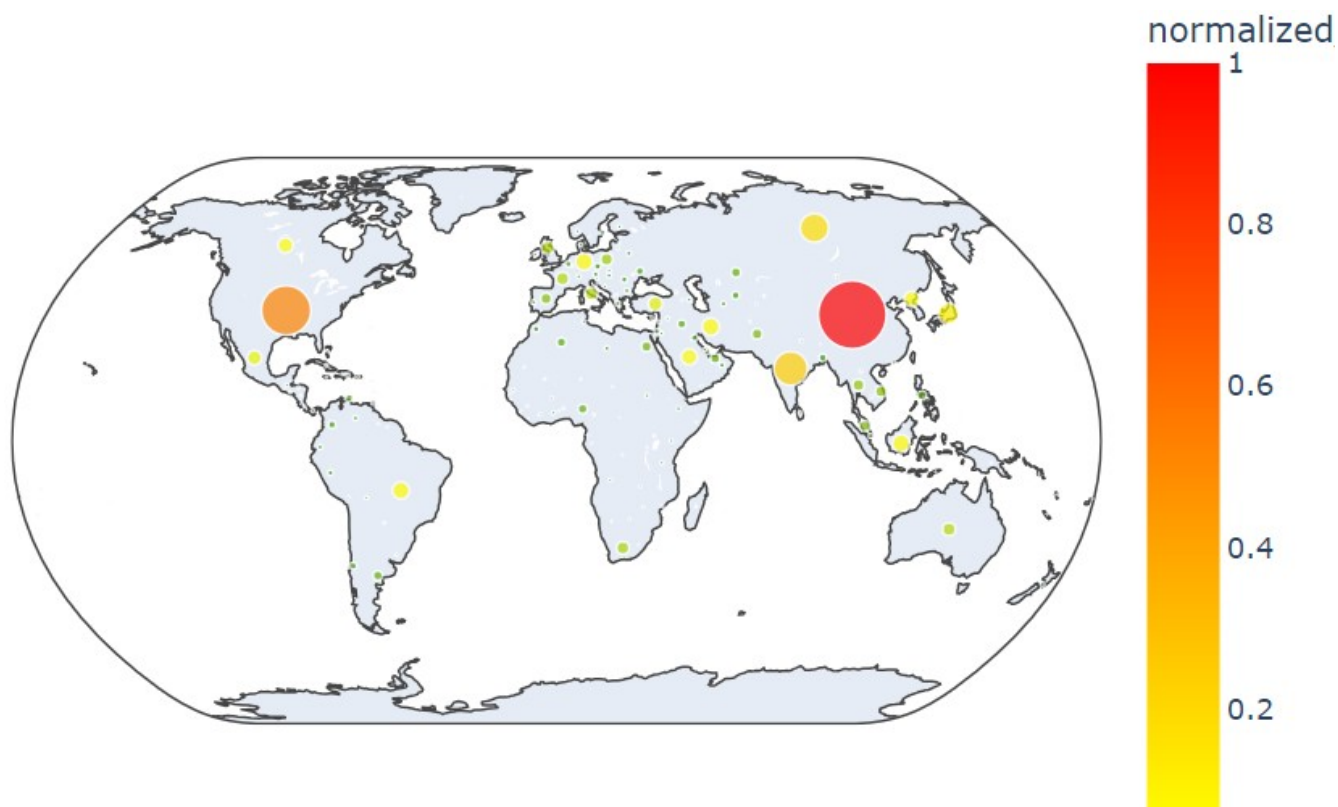


Figure 2: Παγκόσμιος χάρτης με κανονικοποιήμε τη ρύπανση 0-1.

2) Ποια χώρα είναι η πιο ενεργειακά αποδοτική ως προς το ΑΕΠ σε σχέση με την συνολική κατανάλωση ορυκτών τη χρονία 2022.

```
select distinct(f.country_code),c.country_name,f.economy_value/(p.popullation*f.oil_kg) as  
performance,c.latitude,c.longitude  
from fact f  
join gdp_source g on f.economy_id = g.id  
join country c on f.country_code = c.country_code  
join time t on f.year = t.id  
join popullation p on f.country_code=p.country_code and t.year=p.year  
where g.source = 'GDP' and t.year = '2022' and c.latitude is not null and p.popullation>10000000  
order by performance desc;
```

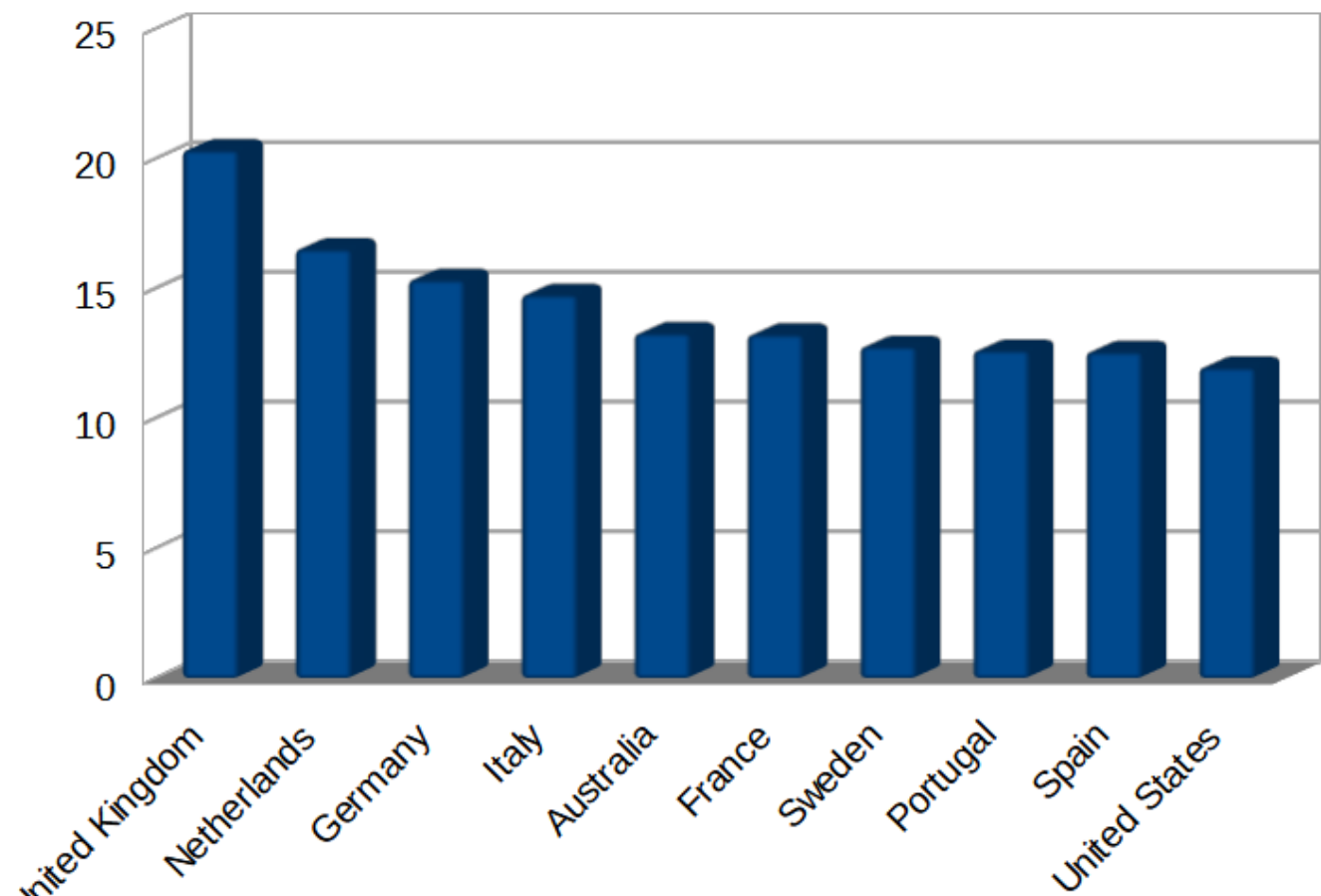


Figure 3: Οι 10 πιο αποδοτικές χώρες σε σχέση ΑΕΠ και κατανάλωσης.

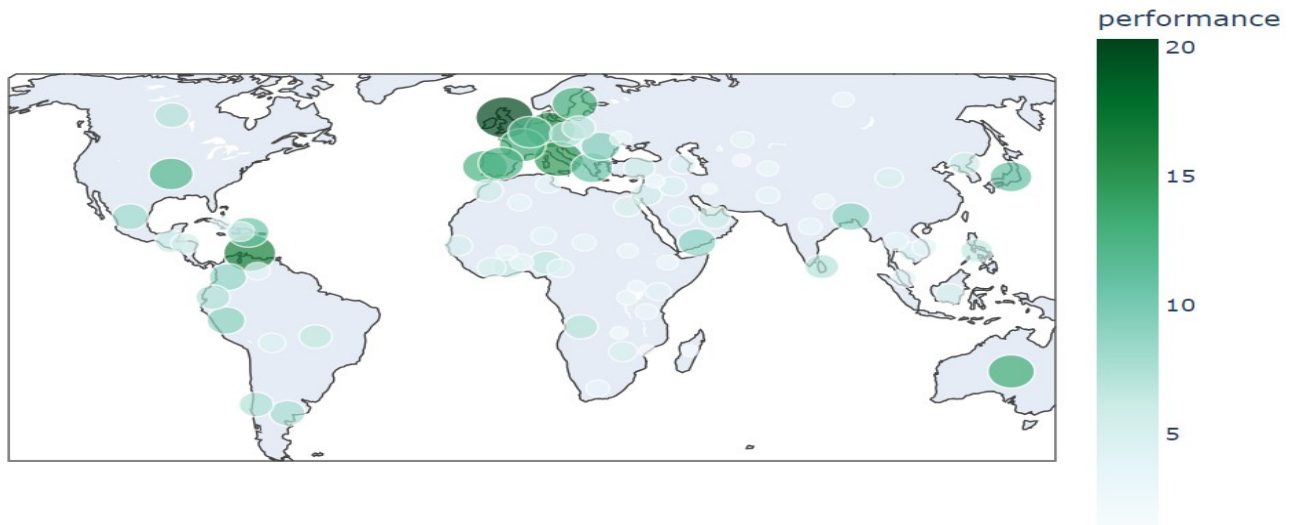


Figure 4: Οι πιο αποδοτικές σε σχέση ΑΕΠ και κατανάλωση χώρες παγκοσμίως.

3) Ποιος ο παράγοντας με τους περισσότερους ρύπους (Μετακίνηση,Βιομηχανία,...)

```
select es.source,sum(f.pollutant_value) total_pollutant
from fact f
join emission_source es on f.pollutant_id=es.id
join time t on f.year = t.id
where t.year = '2022' and es.source not in ('Per capita (Tons)','Smoke from natural sources, cars,
industry (Mt)')
group by es.source
order by total_pollutant desc;
```

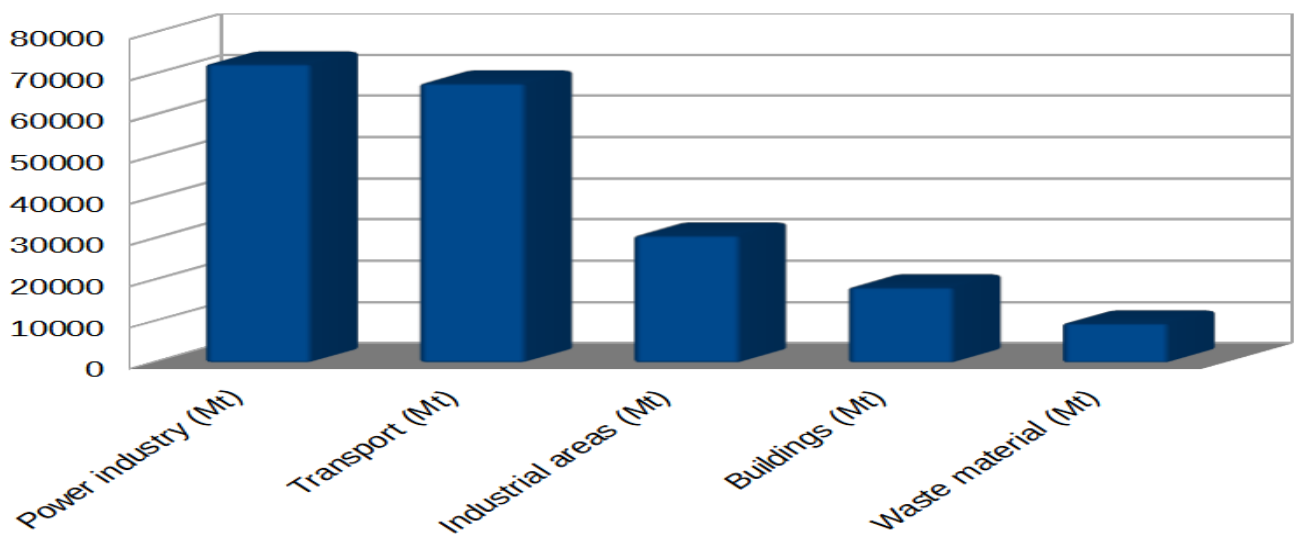
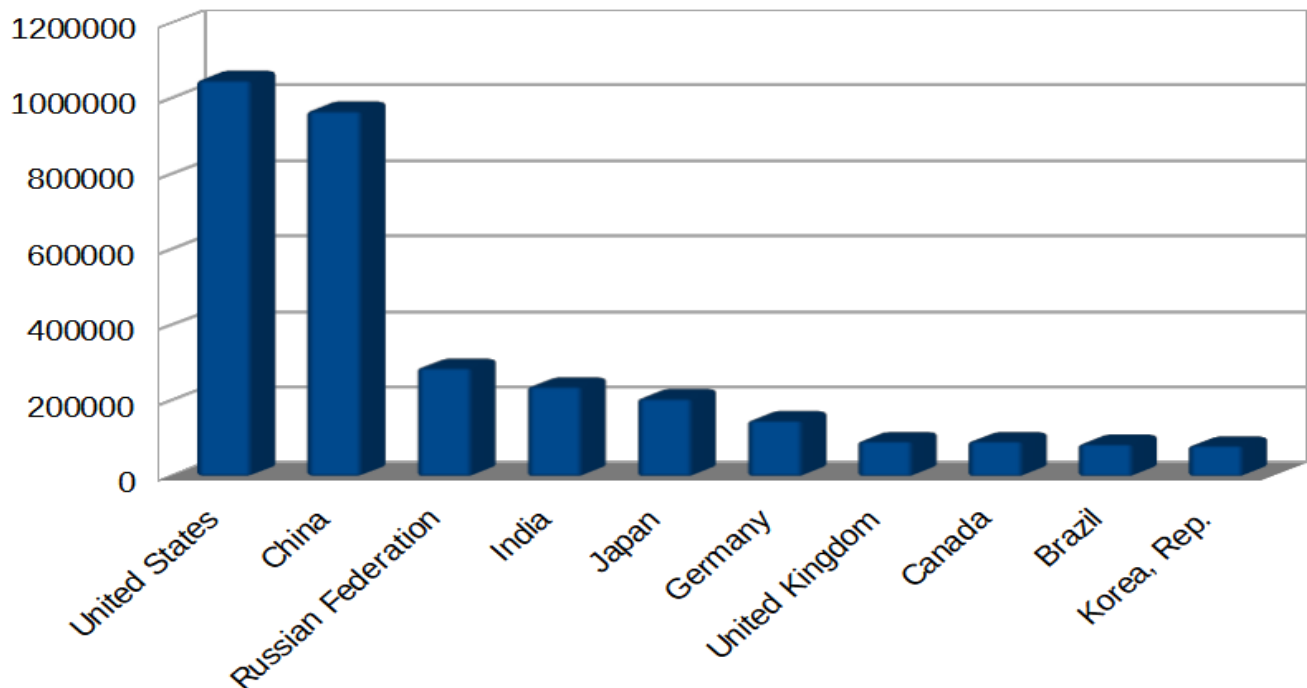


Figure 5: Παράγοντας με τη μεγαλύτερη μόλυνση.

4) Ποια είναι η πιο ρυπογόνα χώρα τα τελευταία 30 χρόνια και ποιο το ποσό των ρύπων;

```
select c.country_code,c.country_name,sum(f.pollutant_value) as poll,c.latitude,c.longitude
from fact f
join country c on f.country_code = c.country_code
join time t on f.year=t.id join emission_source es on f.pollutant_id=es.id
where t.year>=1990 and t.year<=2020 and es.source not in ('Per capita (Tons)','Smoke from natural
sources, cars, industry (Mt)') and c.latitude is not null
group by c.country_code
order by poll desc;
```



5) Ποια η ενεργειακή απόδοση σε σχέση με τους ρύπους τη χρονία 2022 (ένταση ρύπων).

```
select c.country_code,c.country_name,(sum(f.pollutant_value)*1000000)/sum(f.oil_kg*p.population)
as ind
from fact f
join country c on f.country_code = c.country_code
join time t on f.year = t.id
join emission_source es on f.pollutant_id=es.id
join population p on f.country_code = p.country_code and t.year = p.year
where t.year = 2022 and es.source not in ('Per capita (Tons)','Smoke from natural sources, cars,
industry (Mt)') and p.population>10000000
group by c.country_code
order by ind desc;
```

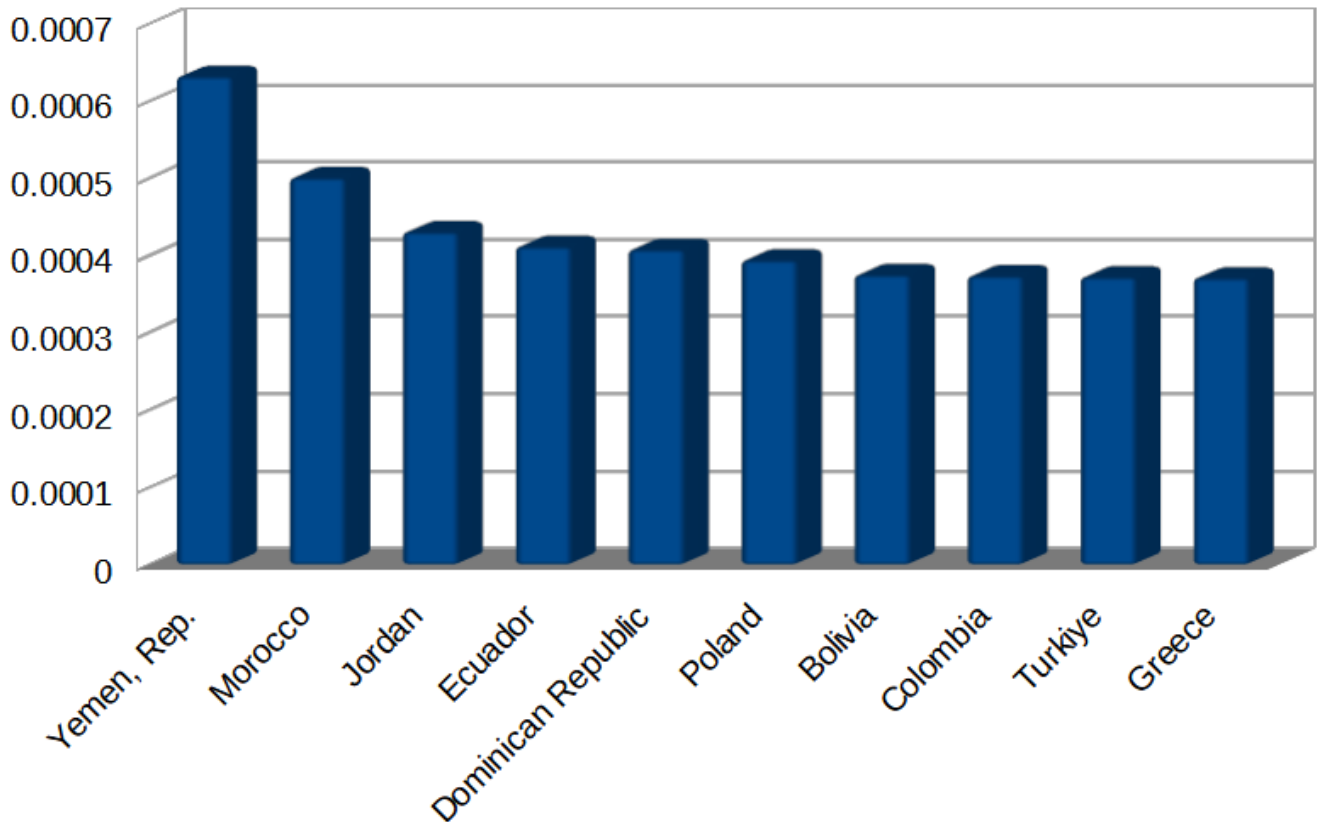


Figure 6: Οι 10 λιγότερο αποδοτικές χώρες σε σχέση με την κατανάλωση ορυκτών.

6) Οι πιο μολυσμένες χώρες ανα κάτοικο.

```
select c.country_code,c.country_name,sum(f.pollutant_value)/min(p.popullation)*1000000 as
emission_per_capita,c.latitude,c.longitude
from fact f
join country c on f.country_code = c.country_code
join time t on f.year = t.id
join emission_source es on f.pollutant_id=es.id
join popullation p on f.country_code = p.country_code and t.year = p.year
where t.year = 2022 and es.source not in ('Per capita (Tons)','Smoke from natural sources, cars,
industry (Mt)') and p.popullation>1000000 and c.latitude is not null
group by c.country_code
order by emission_per_capita desc;
```

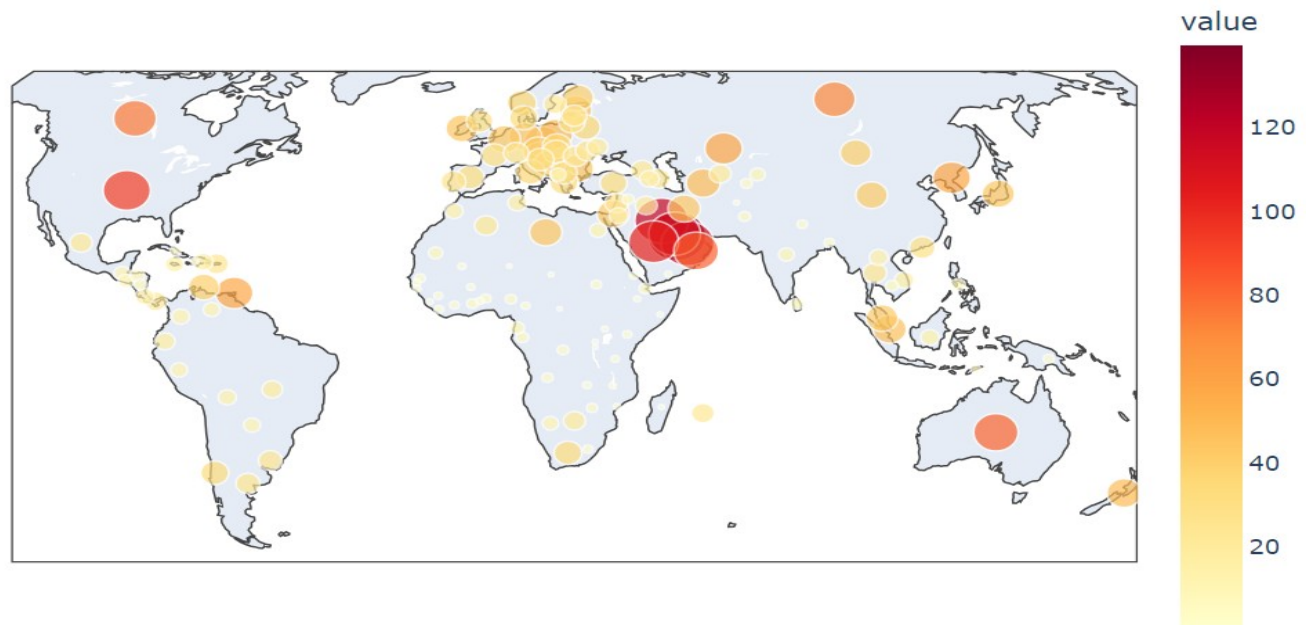


Figure 7: Οι πιο μολυσμένες χώρες ανα κάτοικο παγκοσμίως.

7) Ποια Ήπειρος είναι η πιο ρυπογόνα από το 1990-2023.

```
select t.year,c.region,sum(f.pollutant_value) as total
from fact f
join country c on f.country_code=c.country_code
join time t on f.time_id=t.time_id
group by t.year,c.region
order by t.year;
```

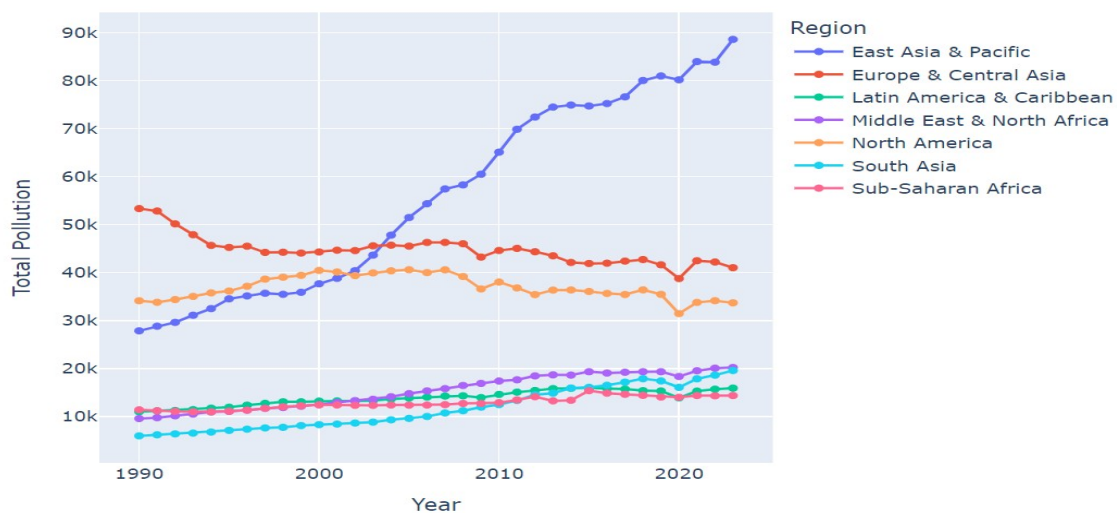


Figure 8: Η άνοδος της ρύπανσης ανά έτος.

8) Οι πιο μολυσματικές χώρες ανα ήπειρο σε αναλογία με τη συνολική κατανάλωσή τους.

Έχει γίνει View το ερώτημα 5 σε pollutant_vol

```

select r.region,r.country_name,r.ind
from pollutant_vol r
join
(select vp.region as region,max(vp.ind) as ind
from pollutant_vol vp
group by vp.region) as m on m.region = r.region and m.ind=r.ind;

```

region	country_name	ind
East Asia & Pacific	Australia	0.0003695944837609916
Europe & Central Asia	Poland	0.0003942207270789668
Latin America & Caribbean	Ecuador	0.0004120848839659522
Middle East & North Africa	Yemen, Rep.	0.0006321861373119797
North America	United States	0.00031602680146851857
South Asia	Sri Lanka	0.00034263457008035404
Sub-Saharan Africa	South Africa	0.00035845943431322703

Figure 9: Οι πιο μολυσμένες χώρες ανα Ήπειρο.

9) Η μέση ενεργειακή παραγωγή ανανεώσιμων πηγών, απο το 1990-2023 και οι πρώτες χώρες της Ευρώπης

```

select c.country_name,es.source,avg(fen.twh) avg_need,c.latitude,c.longitude
from fact_energy_need fen
join country c on fen.country_code = c.country_code
join time t on fen.time_id = t.time_id
join energy_source es on es.source_id = fen.source_id
where es.source = 'Hydro, bioenergy and other renewables'
group by c.country_code
order by avg_need desc;

```

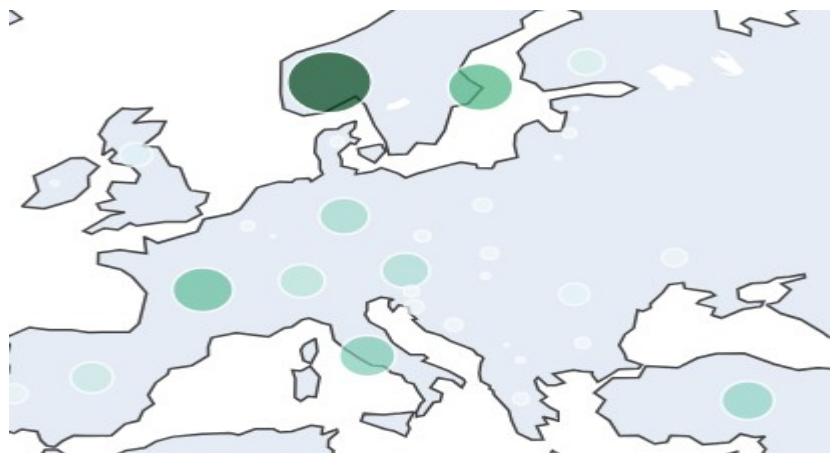


Figure 10: Η χώρα με την μεγαλύτερη παραγωγή ενέργειας από ανανεώσιμες πηγές.

10) Η χώρας με την μεγαλύτερη ανάγκη στην Ευρώπη από το 1990-2023.

```
select c.country_name,es.source,avg(fen.energy_need) avg_need,c.latitude,c.longitude
from fact_energy_need fen
join country c on fen.country_code = c.country_code
join time t on fen.time_id = t.time_id
join energy_source es on es.source_id = fen.source_id
where es.source = 'Hydro, bioenergy and other renewables'
group by c.country_code
order by avg_need desc;
```

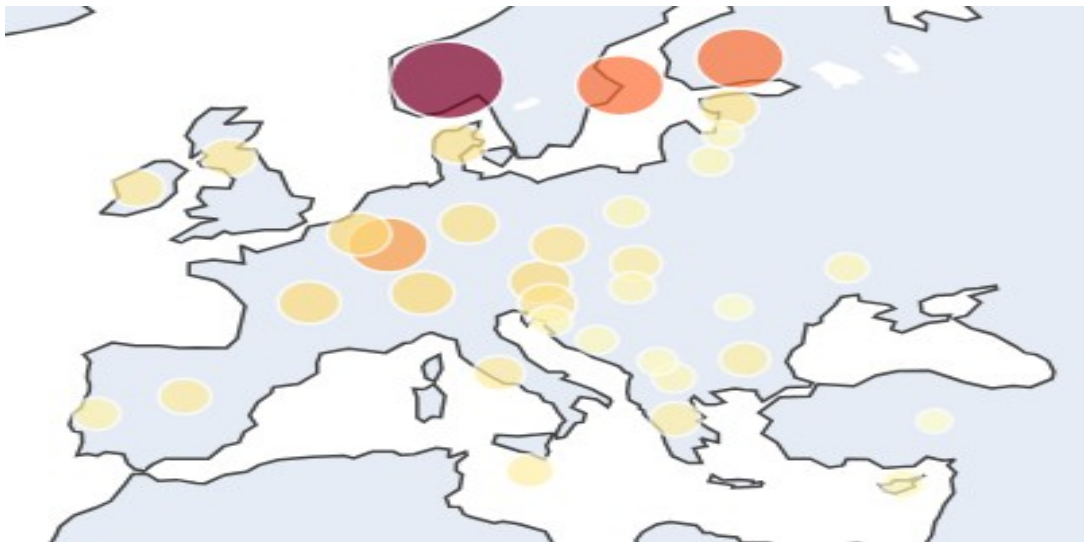


Figure 11: Οι χώρες με την μεγαλύτερη ανάγκη ενέργειας στην Ευρώπη.

11) Ποσοστό ανανεώσιμης ενέργειας επί της συνολικής ανάγκης των Ευρωπαϊκών χωρών

```
select c.country_name,es.source,fen.twh/(fen.energy_need*1000000)
avg_need,c.latitude,c.longitude
from fact_energy_need fen
join country c on fen.country_code = c.country_code
join time t on fen.time_id = t.time_id
join energy_source es on es.source_id = fen.source_id
where es.source = 'Hydro, bioenergy and other renewables' and t.year = 2022
order by avg_need desc;
```


country_name	source	avg_need
Germany	Hydro, bioenergy and other renewables	0.000010347352024922119
Italy	Hydro, bioenergy and other renewables	0.000009566420664206642
United Kingdom	Hydro, bioenergy and other renewables	0.000008899141630901288
France	Hydro, bioenergy and other renewables	0.000007651577503429355
Sweden	Hydro, bioenergy and other renewables	0.00000622655663915979
Norway	Hydro, bioenergy and other renewables	0.000005274220032840722
Romania	Hydro, bioenergy and other renewables	0.00000501027397260274
Austria	Hydro, bioenergy and other renewables	0.000004875
Spain	Hydro, bioenergy and other renewables	0.000004366726296958855
Switzerland	Hydro, bioenergy and other renewables	0.0000043333333333333333
Ukraine	Hydro, bioenergy and other renewables	0.00000432

Figure 12: Ποσοστό ανανεώσιμης ενέργειας σε σχέση με τη συνολική ανάγκη σε φθίνουσα σειρά.

6. Συμπεράσματα

- Οι πιο ρυπογόνες χώρες είναι η Κίνα και η Αμερική και όπως παρατηρήθηκε με μεγάλη απόκλιση από τις υπόλοιπες χώρες, με τάξη μεγέθους δεκάδες χιλιάδες, σε σχέση με τις υπόλοιπες που είναι στις χιλιάδες (Figure 1, Figure 5).
- Οι μεγαλύτερες πηγές ρυπογόνων ουσιών προέρχονται από τη Βιομηχανία ενέργειας και τις ανθρώπινες μετακινήσεις (Figure 4).
- Στα τελευταία 30 χρόνια παρατηρούμε πως η Κίνα και η Αμερική πρωταγωνιστούν στη μόλυνση, με τεράστια διαφορά παγκοσμίως.
- Ανάμεσα στις 10 λιγότερο αποδοτικές χώρες κατατάσσεται η Ελλάδα.
- Η μικρότερη ένταση των ρύπων είναι στην Ευρώπη, με το Ηνωμένο Βασίλειο να ξεχωρίζει.
- Μια παρατήρηση που κάνουμε από Figure 7 είναι ότι η Μέση Ανατολή, Αμερική, Μεξικό και Αυστραλία είναι οι λιγότερο φιλικές για την ανθρώπινη υγεία.
- Η ανατολική Ασία έχει συνεχόμενη άνοδο ρύπανσης και μάλιστα έχει τριπλασιαστεί από το 1990 μέχρι και σήμερα.
- Η Ήπειρος Αμερική έχει σταθερή πορεία ανάμεσα στα χρόνια.
- Η πιο ανησυχητικές περιοχές παγκοσμίως είναι οι Ανατολική Ασία, Αμερική και Κεντρική Ευρώπη.
- Οι πιο μολυσματικές χώρες ανά ήπειρο είναι οι Australlia, Polland, Ecuador, Yemen, United States, Sri Lanka, South Africa, όπως φαίνεται στο Figure 9.
- Οι δύο πρώτες χώρες σε παραγωγή ανανεώσιμης ενέργειας είναι οι σκανδιναβικές χώρες.
- Οι χώρες με τις μεγαλύτερες ανάγκες στην Ευρώπη είναι επίσης στη σκανδιναβία.
- Με βάση το Figure 12, αν μια χώρα προσπαθούσε να ζήσει μόνο από την ανανεώσιμη ενέργεια, δεν θα μπορούσε, καθώς καλύπτει στην καλύτερη εκδοχή το 0.00001 των αναγκών της.

Σημείωση: Οι παραπάνω χάρτες αντικατοπτρίζουν τα κέντρα των πρωτευουσών της κάθε χώρας. Για τα (Figure 2, Figure 5) έχουμε πάρει πληθυσμό μεγαλύτερο των 10 εκατομμυρίων για να μην προκύψουν ακραία σημεία (outliers).

Μέρας Γεώργιος
MSc Candidate in Software Engineering
Πανεπιστήμιο Θεσσαλίας
Τμήμα Ψηφιακών Συστημάτων
gmeras@uth.gr