BLG 317E

DATABASE SYSTEMS PROJECT REPORT 2023

GROUP MEMBERS:

Kerem Er	:	820210313
Hakan Duran	:	150200091
Rengin Helin Yalçın	:	150220767
Oğuz Eren Kacar	:	150200018

1 Introduction

1.1 Project Overview

The purpose of this project is to design and implement a web application using Python Flask as the back-end framework and MySQL as the database management system. The website aims to provide a platform for global socio-economic and environmental indicators between years 1990 and 2021.

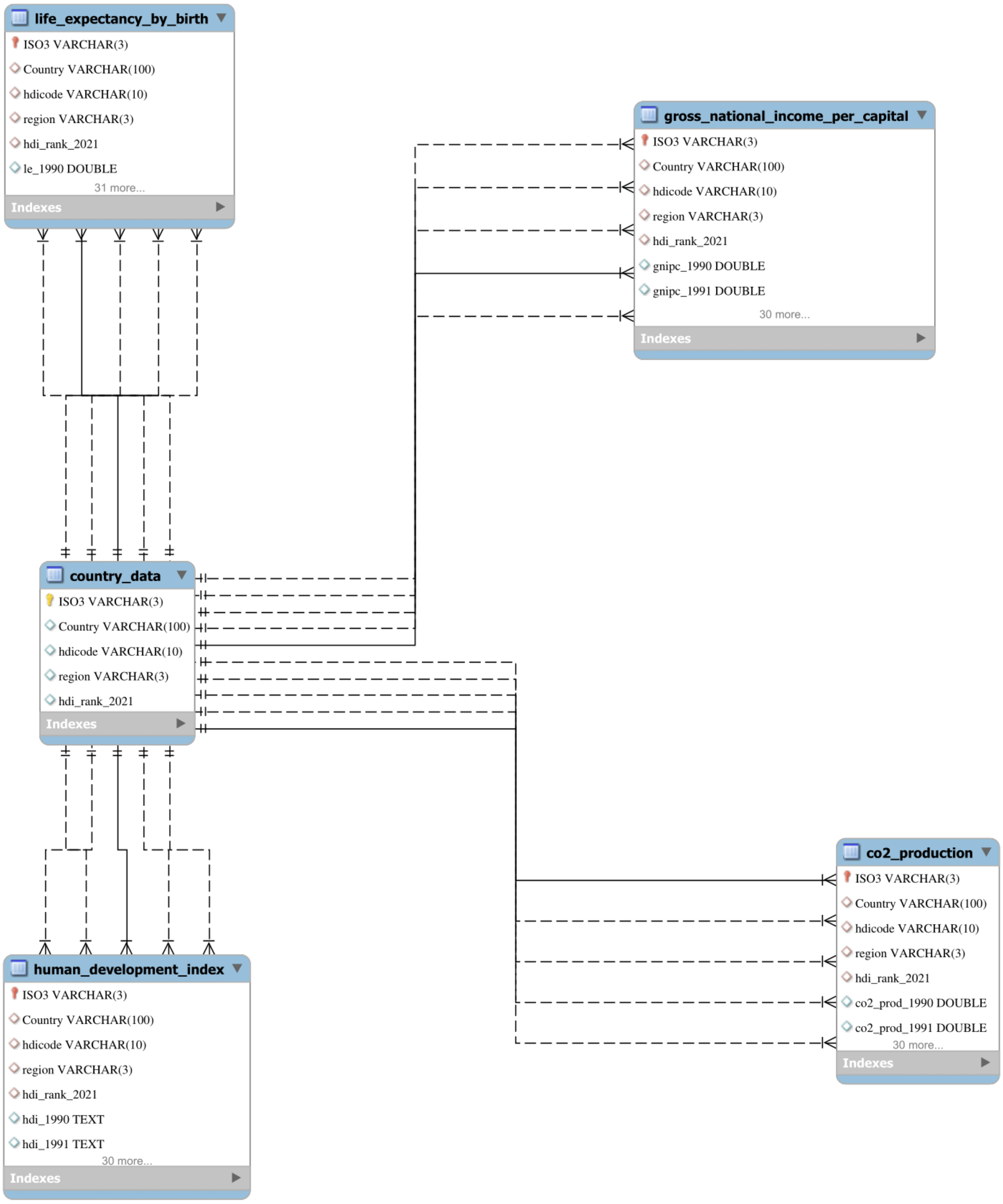
1.2 Objectives

- Develop a responsive and user-friendly website.
- Implement a robust back-end using Python Flask.
- Design and optimize a relational database using SQL.

2 Design

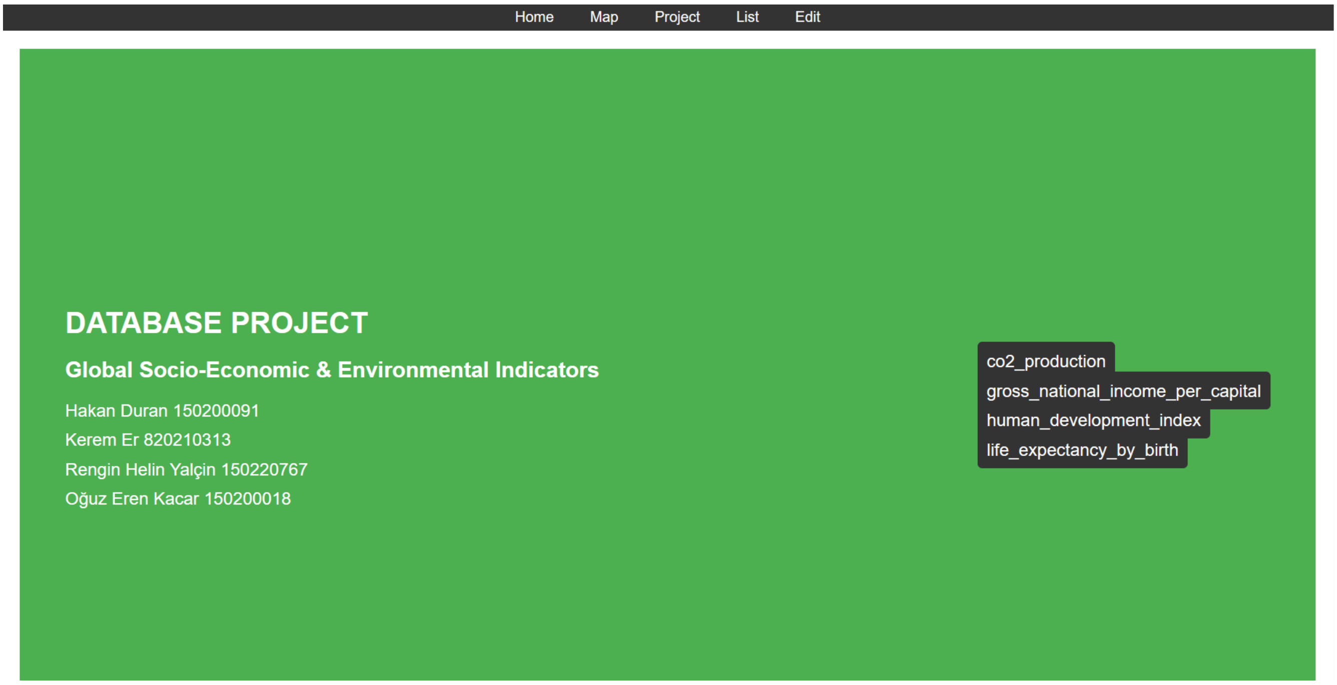
2.1 Database Design

Here is the Entity Relationship Diagram (ERD) of our database :



2.2 Front-end Design

Here are some screenshots of our website's front end design :

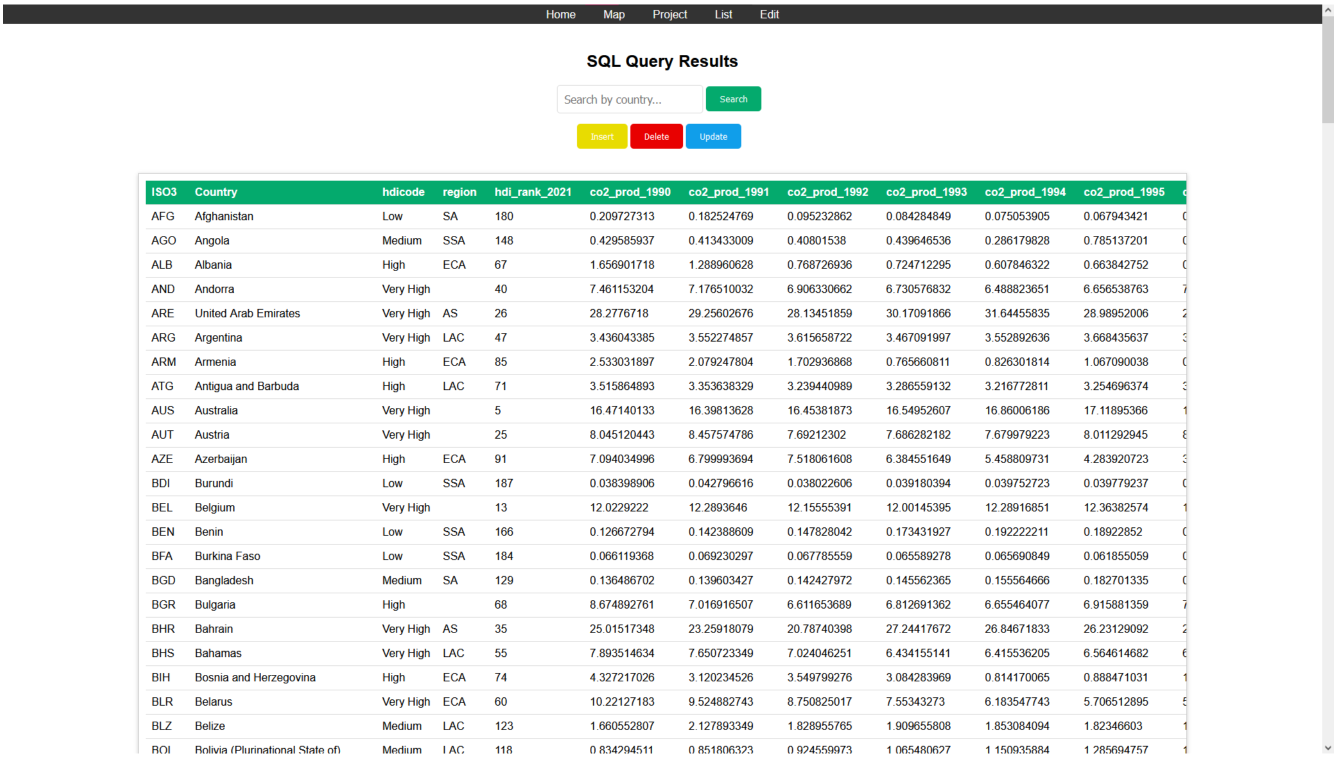


DATABASE PROJECT

Global Socio-Economic & Environmental Indicators

Hakan Duran 150200091
Kerem Er 820210313
Rengin Helin Yalçin 150220767
Oğuz Eren Kacar 150200018

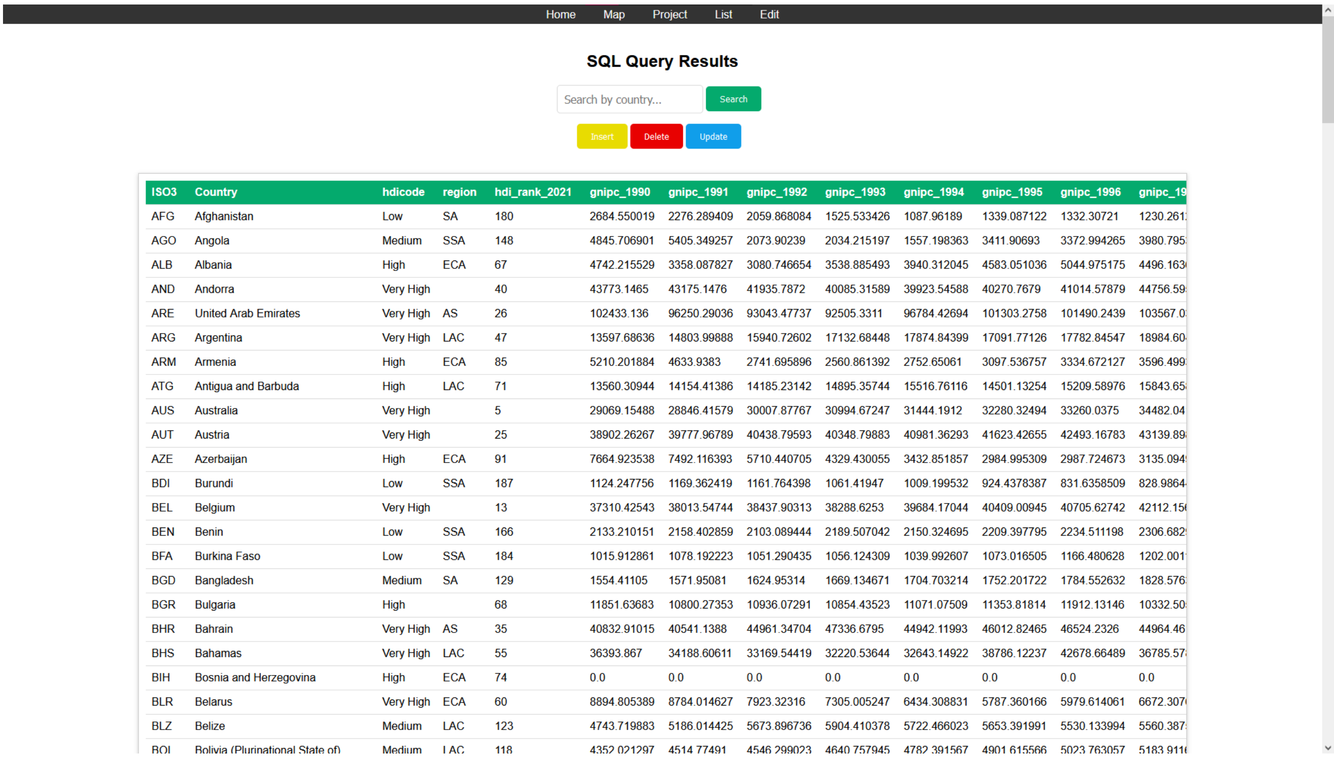
Figure 1: Home Page



SQL Query Results

ISO3	Country	hdicode	region	hdi_rank_2021	co2_prod_1990	co2_prod_1991	co2_prod_1992	co2_prod_1993	co2_prod_1994	co2_prod_1995
AFG	Afghanistan	Low	SA	180	0.209727313	0.182524769	0.095232862	0.084284849	0.075053905	0.067943421
AGO	Angola	Medium	SSA	148	0.429585937	0.413433009	0.40801538	0.439646536	0.286179828	0.785137201
ALB	Albania	High	ECA	67	1.656901718	1.288960628	0.768726936	0.724712295	0.607846322	0.663842752
AND	Andorra	Very High		40	7.461153204	7.176510032	6.906330662	6.730576832	6.488823651	6.656538763
ARE	United Arab Emirates	Very High	AS	26	28.2776718	29.25602676	28.13451859	30.17091866	31.64455835	28.98952006
ARG	Argentina	Very High	LAC	47	3.436043385	3.552274857	3.615658722	3.467091997	3.552892636	3.668435637
ARM	Armenia	High	ECA	85	2.533031897	2.079247804	1.702936868	0.765660811	0.826301814	1.067090038
ATG	Antigua and Barbuda	High	LAC	71	3.515864893	3.353638329	3.239440989	3.286559132	3.216772811	3.254696374
AUS	Australia	Very High		5	16.47140133	16.39813628	16.45381873	16.54952607	16.86006186	17.11895366
AUT	Austria	Very High		25	8.045120443	8.457574786	7.69212302	7.686282182	7.679979223	8.011292945
AZE	Azerbaijan	High	ECA	91	7.094034996	6.799993694	7.518061608	6.384551649	5.458809731	4.283920723
BDI	Burundi	Low	SSA	187	0.038398906	0.042796616	0.038022606	0.039180394	0.039752723	0.039779237
BEL	Belgium	Very High		13	12.0229222	12.2893646	12.15555391	12.00145395	12.28916851	12.36382574
BEN	Benin	Low	SSA	166	0.126672794	0.142388609	0.147828042	0.173431927	0.192222211	0.18922852
BFA	Burkina Faso	Low	SSA	184	0.066119368	0.069230297	0.067785559	0.065589278	0.065690849	0.061855059
BGD	Bangladesh	Medium	SA	129	0.136486702	0.139603427	0.142427972	0.145562365	0.155564666	0.182701335
BGR	Bulgaria	High		68	8.674892761	7.016916507	6.611653689	6.812691362	6.655464077	6.915881359
BHR	Bahrain	Very High	AS	35	25.01517348	23.25918079	20.78740398	27.24417672	26.84671833	26.23129092
BHS	Bahamas	Very High	LAC	55	7.893514634	7.650723349	7.024046251	6.434155141	6.415536205	6.564614682
BIH	Bosnia and Herzegovina	High	ECA	74	4.327217026	3.120234526	3.549799276	3.084283969	0.814170065	0.888471031
BLR	Belarus	Very High	ECA	60	10.22127183	9.524882743	8.750825017	7.55343273	6.183547743	5.706512895
BLZ	Belize	Medium	LAC	123	1.660552807	2.127893349	1.828955765	1.909655808	1.853084094	1.82346603
BOL	Bolivia (Plurinational State of)	Medium	LAC	118	0.834294511	0.851806323	0.924559973	1.065480627	1.150935884	1.285694757

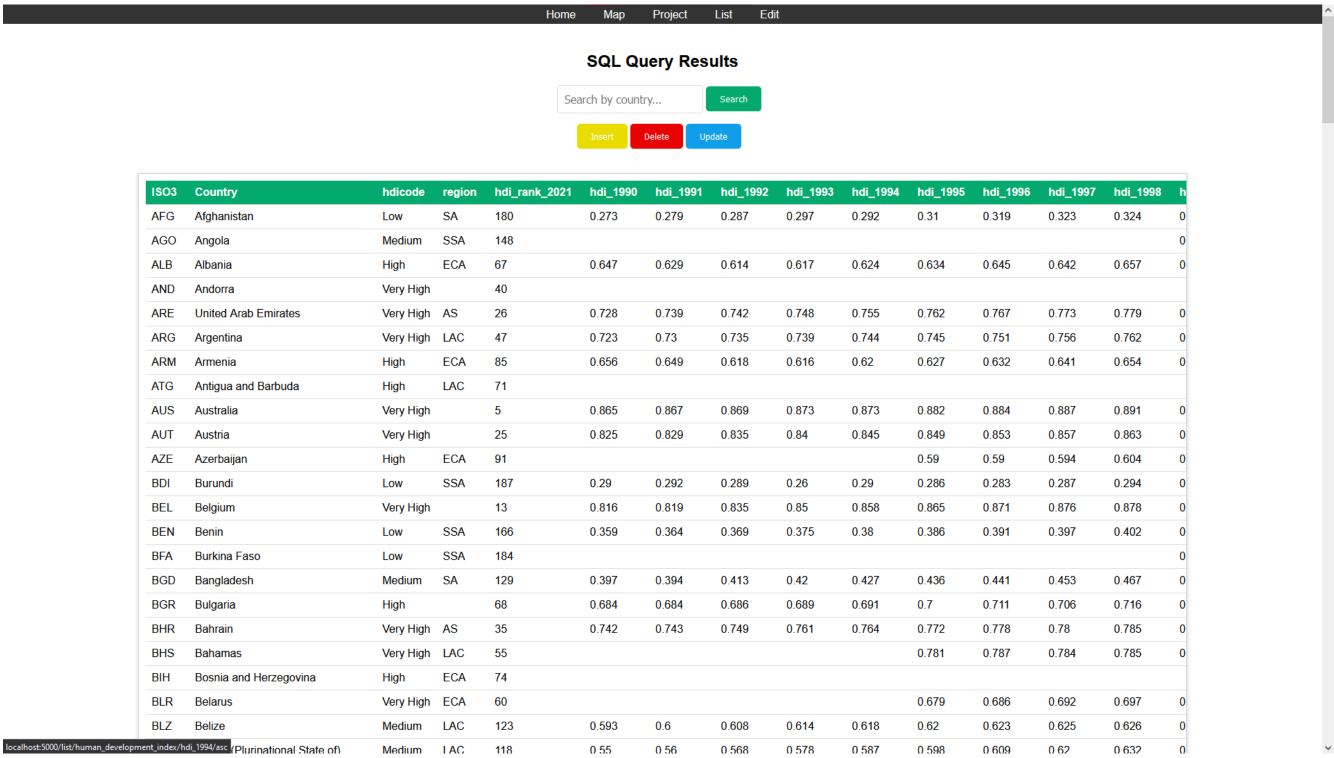
Figure 2: CO2 Production



SQL Query Results

ISO3	Country	hdicode	region	hdi_rank_2021	gnipc_1990	gnipc_1991	gnipc_1992	gnipc_1993	gnipc_1994	gnipc_1995	gnipc_1996	gnipc_19
AFG	Afghanistan	Low	SA	180	2684.550019	2276.289409	2059.868084	1525.533426	1087.96189	1339.087122	1332.30721	1230.261
AGO	Angola	Medium	SSA	148	4845.706901	5405.349257	2073.90239	2034.215197	1557.198363	3411.90693	3372.994265	3980.795
ALB	Albania	High	ECA	67	4742.215529	3358.087827	3080.746654	3538.885493	3940.312045	4583.051036	5044.975175	4496.163
AND	Andorra	Very High		40	43773.1465	43175.1476	41935.7872	40085.31589	39923.54588	40270.7679	41014.57879	44756.59
ARE	United Arab Emirates	Very High	AS	26	102433.136	96250.29036	93043.47737	92505.3311	96784.42694	101303.2758	101490.2439	103567.0
ARG	Argentina	Very High	LAC	47	13597.68636	14803.99888	15940.72602	17132.68448	17874.84399	17091.77126	17782.84547	18984.60
ARM	Armenia	High	ECA	85	5210.201884	4633.9383	2741.695896	2560.861392	2752.65061	3097.536757	3334.672127	3596.499
ATG	Antigua and Barbuda	High	LAC	71	13560.30944	14154.41386	14185.23142	14895.35744	15516.76116	14501.13254	15209.58976	15843.65
AUS	Australia	Very High		5	29069.15488	28846.41579	30007.87767	30994.67247	31444.1912	32280.32494	33260.0375	34482.04
AUT	Austria	Very High		25	38902.26267	39777.96789	40438.79593	40348.79883	40981.36293	41623.42655	42493.16783	43139.89
AZE	Azerbaijan	High	ECA	91	7664.923538	7492.116393	5710.440705	4329.430055	3432.851857	2984.995309	2987.724673	3135.094
BDI	Burundi	Low	SSA	187	1124.247756	1169.362419	1161.764398	1061.41947	1009.199532	924.4378387	831.6358509	828.9864
BEL	Belgium	Very High		13	37310.42543	38013.54744	38437.90313	38288.6253	39684.17044	40409.00945	40705.62742	42112.15
BEN	Benin	Low	SSA	166	2133.210151	2158.402859	2103.089444	2189.507042	2150.324695	2209.397795	2234.511198	2306.682
BFA	Burkina Faso	Low	SSA	184	1015.912861	1078.192223	1051.290435	1056.124309	1039.992607	1073.016505	1166.480628	1202.001
BGD	Bangladesh	Medium	SA	129	1554.41105	1571.95081	1624.95314	1669.134671	1704.703214	1752.201722	1784.552632	1828.576
BGR	Bulgaria	High		68	11851.63683	10800.27353	10936.07291	10854.43523	11071.07509	11353.81814	11912.13146	10332.50
BHR	Bahrain	Very High	AS	35	40832.91015	40541.1388	44961.34704	47336.6795	44942.11993	46012.82465	46524.2326	44964.46
BHS	Bahamas	Very High	LAC	55	36393.867	34188.60611	33169.54419	32220.53644	32643.14922	38786.12237	42678.66489	36785.57
BIH	Bosnia and Herzegovina	High	ECA	74	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BLR	Belarus	Very High	ECA	60	8894.805389	8784.014627	7923.32316	7305.005247	6434.308831	5787.360166	5979.614061	6672.307
BLZ	Belize	Medium	LAC	123	4743.719883	5186.014425	5673.896736	5904.410378	5722.466023	5653.391991	5530.133994	5560.387
BOL	Bolivia (Plurinational State of)	Medium	LAC	118	4352.021297	4514.77491	4546.299023	4640.757945	4782.391567	4901.615566	5023.763057	5183.911

Figure 3: Gross National Income Per Capital

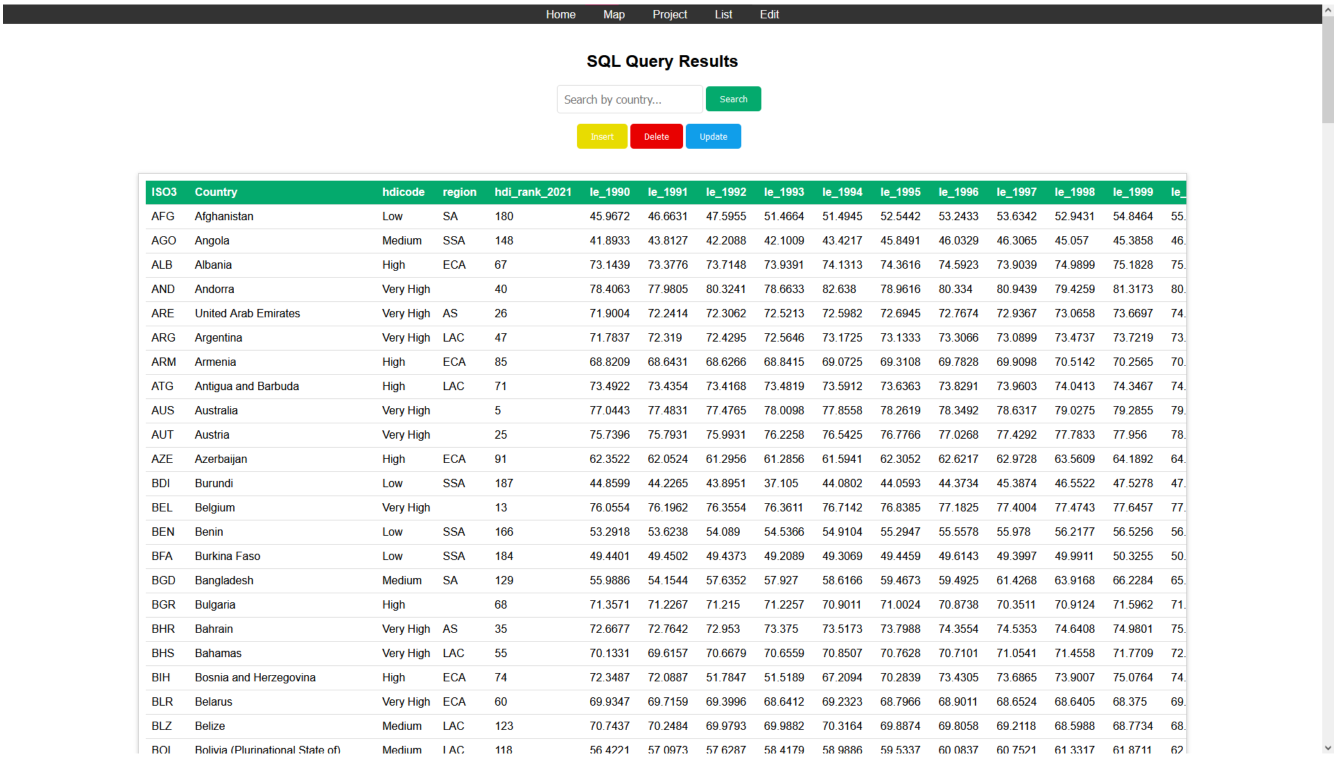


SQL Query Results

ISO3	Country	hdicode	region	hdi_rank_2021	hdi_1990	hdi_1991	hdi_1992	hdi_1993	hdi_1994	hdi_1995	hdi_1996	hdi_1997	hdi_1998	h
AFG	Afghanistan	Low	SA	180	0.273	0.279	0.287	0.297	0.292	0.31	0.319	0.323	0.324	0
AGO	Angola	Medium	SSA	148										0
ALB	Albania	High	ECA	67	0.647	0.629	0.614	0.617	0.624	0.634	0.645	0.642	0.657	0
AND	Andorra	Very High		40										
ARE	United Arab Emirates	Very High	AS	26	0.728	0.739	0.742	0.748	0.755	0.762	0.767	0.773	0.779	0
ARG	Argentina	Very High	LAC	47	0.723	0.73	0.735	0.739	0.744	0.745	0.751	0.756	0.762	0
ARM	Armenia	High	ECA	85	0.656	0.649	0.618	0.616	0.62	0.627	0.632	0.641	0.654	0
ATG	Antigua and Barbuda	High	LAC	71										
AUS	Australia	Very High		5	0.865	0.867	0.869	0.873	0.873	0.882	0.884	0.887	0.891	0
AUT	Austria	Very High		25	0.825	0.829	0.835	0.84	0.845	0.849	0.853	0.857	0.863	0
AZE	Azerbaijan	High	ECA	91						0.59	0.59	0.594	0.604	0
BDI	Burundi	Low	SSA	187	0.29	0.292	0.289	0.26	0.29	0.286	0.283	0.287	0.294	0
BEL	Belgium	Very High		13	0.816	0.819	0.835	0.85	0.858	0.865	0.871	0.876	0.878	0
BEN	Benin	Low	SSA	166	0.359	0.364	0.369	0.375	0.38	0.386	0.391	0.397	0.402	0
BFA	Burkina Faso	Low	SSA	184										0
BGD	Bangladesh	Medium	SA	129	0.397	0.394	0.413	0.42	0.427	0.436	0.441	0.453	0.467	0
BGR	Bulgaria	High		68	0.684	0.684	0.686	0.689	0.691	0.7	0.711	0.706	0.716	0
BHR	Bahrain	Very High	AS	35	0.742	0.743	0.749	0.761	0.764	0.772	0.778	0.78	0.785	0
BHS	Bahamas	Very High	LAC	55						0.781	0.787	0.784	0.785	0
BIH	Bosnia and Herzegovina	High	ECA	74										
BLR	Belarus	Very High	ECA	60						0.679	0.686	0.692	0.697	0
BLZ	Belize	Medium	LAC	123	0.593	0.6	0.608	0.614	0.618	0.62	0.623	0.625	0.626	0
	(Plurinational State of)	Medium	LAC	118	0.55	0.56	0.568	0.578	0.587	0.598	0.609	0.62	0.632	0

localhost:5000/list/human_development_index/hdi_1994/asc

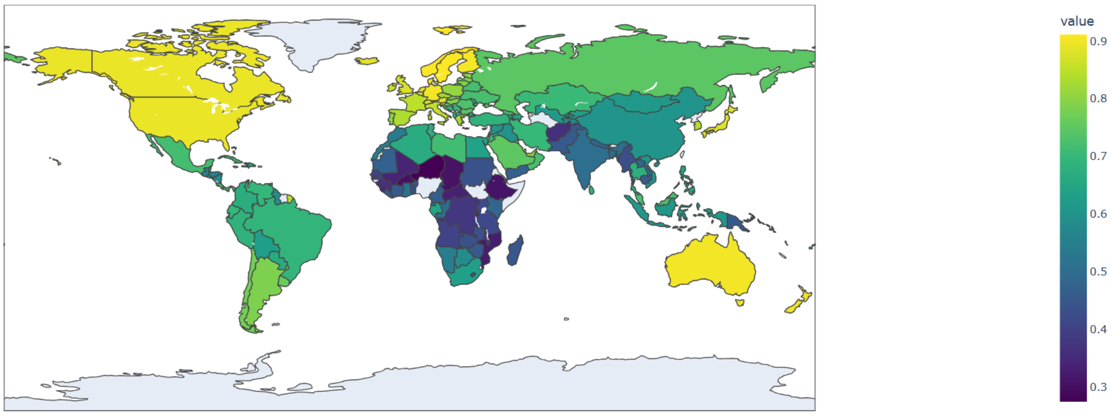
Figure 4: Human Development Index



SQL Query Results

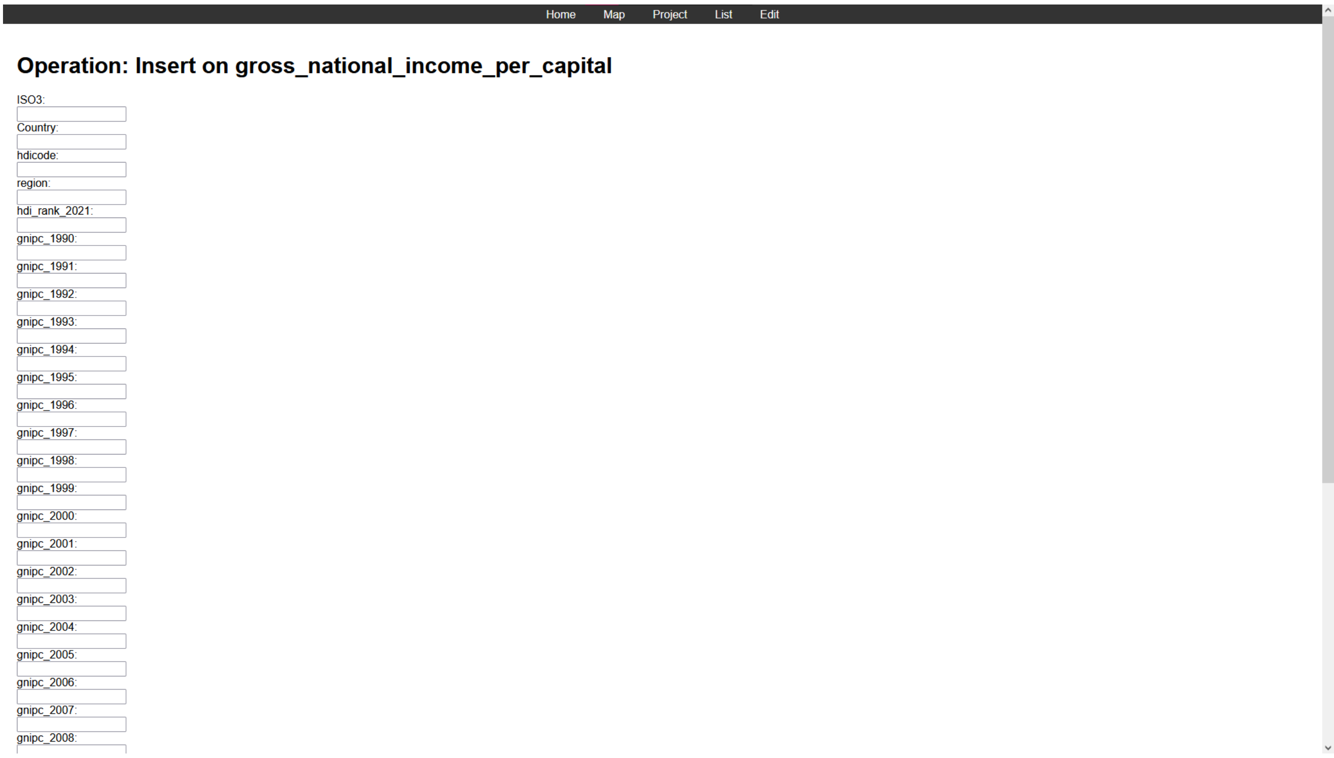
ISO3	Country	hdicode	region	hdi_rank_2021	le_1990	le_1991	le_1992	le_1993	le_1994	le_1995	le_1996	le_1997	le_1998	le_1999	le_
AFG	Afghanistan	Low	SA	180	45.9672	46.6631	47.5955	51.4664	51.4945	52.5442	53.2433	53.6342	52.9431	54.8464	55.
AGO	Angola	Medium	SSA	148	41.8933	43.8127	42.2088	42.1009	43.4217	45.8491	46.0329	46.3065	45.057	45.3858	46.
ALB	Albania	High	ECA	67	73.1439	73.3776	73.7148	73.9391	74.1313	74.3616	74.5923	73.9039	74.9899	75.1828	75.
AND	Andorra	Very High		40	78.4063	77.9805	80.3241	78.6633	82.638	78.9616	80.334	80.9439	79.4259	81.3173	80.
ARE	United Arab Emirates	Very High	AS	26	71.9004	72.2414	72.3062	72.5213	72.5982	72.6945	72.7674	72.9367	73.0658	73.6697	74.
ARG	Argentina	Very High	LAC	47	71.7837	72.319	72.4295	72.5646	73.1725	73.1333	73.3066	73.0899	73.4737	73.7219	73.
ARM	Armenia	High	ECA	85	68.8209	68.6431	68.6266	68.8415	69.0725	69.3108	69.7828	69.9098	70.5142	70.2565	70.
ATG	Antigua and Barbuda	High	LAC	71	73.4922	73.4354	73.4168	73.4819	73.5912	73.6363	73.8291	73.9603	74.0413	74.3467	74.
AUS	Australia	Very High		5	77.0443	77.4831	77.4765	78.0098	77.8558	78.2619	78.3492	78.6317	79.0275	79.2855	79.
AUT	Austria	Very High		25	75.7396	75.7931	75.9931	76.2258	76.5425	76.7766	77.0268	77.4292	77.7833	77.956	78.
AZE	Azerbaijan	High	ECA	91	62.3522	62.0524	61.2956	61.2856	61.5941	62.3052	62.6217	62.9728	63.5609	64.1892	64.
BDI	Burundi	Low	SSA	187	44.8599	44.2265	43.8951	37.105	44.0802	44.0593	44.3734	45.3874	46.5522	47.5278	47.
BEL	Belgium	Very High		13	76.0554	76.1962	76.3554	76.3611	76.7142	76.8385	77.1825	77.4004	77.4743	77.6457	77.
BEN	Benin	Low	SSA	166	53.2918	53.6238	54.089	54.5366	54.9104	55.2947	55.5578	55.978	56.2177	56.5256	56.
BFA	Burkina Faso	Low	SSA	184	49.4401	49.4502	49.4373	49.2089	49.3069	49.4459	49.6143	49.3997	49.9911	50.3255	50.
BGD	Bangladesh	Medium	SA	129	55.9886	54.1544	57.6352	57.927	58.6166	59.4673	59.4925	61.4268	63.9168	66.2284	65.
BGR	Bulgaria	High		68	71.3571	71.2267	71.215	71.2257	70.9011	71.0024	70.8738	70.3511	70.9124	71.5962	71.
BHR	Bahrain	Very High	AS	35	72.6677	72.7642	72.953	73.375	73.5173	73.7988	74.3554	74.5353	74.6408	74.9801	75.
BHS	Bahamas	Very High	LAC	55	70.1331	69.6157	70.6679	70.6559	70.8507	70.7628	70.7101	71.0541	71.4558	71.7709	72.
BIH	Bosnia and Herzegovina	High	ECA	74	72.3487	72.0887	51.7847	51.5189	67.2094	70.2839	73.4305	73.6865	73.9007	75.0764	74.
BLR	Belarus	Very High	ECA	60	69.9347	69.7159	69.3996	68.6412	69.2323	68.7966	68.9011	68.6524	68.6405	68.375	69.
BLZ	Belize	Medium	LAC	123	70.7437	70.2484	69.9793	69.9882	70.3164	69.8874	69.8058	69.2118	68.5988	68.7734	68.
BOL	Bolivia (Plurinational State of)	Medium	LAC	118	56.4221	57.0973	57.6287	58.4179	58.9886	59.5337	60.0837	60.7521	61.3317	61.8711	62

Figure 5: Life Expectancy By Birth



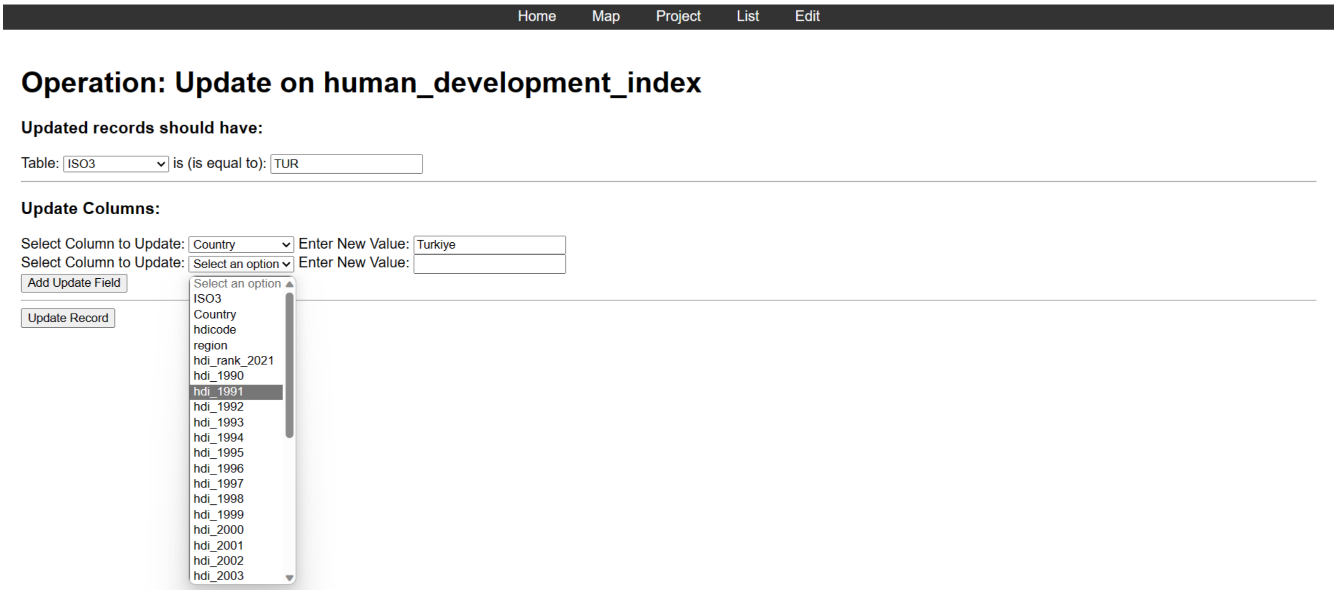
Another world map selection

Figure 6: 2002 human development index map



Operation: Insert on gross_national_income_per_capital

Figure 7: Insert Page



Operation: Update on human_development_index

Updated records should have:

Table: ISO3 is (is equal to): TUR

Update Columns:

Select Column to Update: Country Enter New Value: Turkiye

Select Column to Update: Select an option Enter New Value:

Figure 8: Update Page

3 Implementation

3.1 Database CRUD operations and SQL queries

The following SQL queries were used:

Code Listing 1: Inserting values that exists on other tables but doesn't exist on this table

```sql
INSERT INTO new.co2_production (ISO3, Country, hdicode, region,
   hdi_rank_2021)
VALUES ('ERI', 'Eritrea', 'Low', 'SSA', '176'),
/* There are a lot between. */
('TLS', 'Timor-Leste', 'Medium', 'EAP', '140');

INSERT INTO new.gross_national_income_per_capital (ISO3, Country,
    hdicode, region, hdi_rank_2021)
VALUES ('BIH', 'Bosnia and Herzegovina', 'High', 'ECA', '74'),
/* There are a lot between. */
('WSM', 'Samoa', 'High', 'EAP', '111');

INSERT INTO new.human_development_index (ISO3, Country, hdicode,
   region, hdi_rank_2021)
VALUES ('ATG', 'Antigua and Barbuda', 'High', 'LAC', '71'),
/* There are a lot between. */
('VUT', 'Vanuatu', 'Medium', 'EAP', '140');
```

Code Listing 2: Setting the empty values as null

```sql
UPDATE `new`.`life_expectancy_by_birth` SET `region` = NULL WHERE
    (`ISO3` = 'AND');
UPDATE `new`.`life_expectancy_by_birth` SET `region` = NULL WHERE
    (`ISO3` = 'AUS');
/* There are a lot between. */
UPDATE `new`.`human_development_index` SET `region` = NULL WHERE
   (`ISO3` = 'BEL');
```

```
/* There are a lot between. */
UPDATE `new`.`gross_national_income_per_capital` SET `region` =
   NULL WHERE (`ISO3` = 'CYP');
/* There are a lot between. */
UPDATE `new`.`country_data` SET `region` = NULL WHERE (`ISO3` = '
   AND');
/* There are a lot between. */
UPDATE `new`.`country_data` SET `region` = NULL WHERE (`ISO3` = '
   USA');
```

Code Listing 3: Fixing country names

```
UPDATE `new`.`country_data` SET `Country` = 'Ivory Coast' WHERE
   (`ISO3` = 'CIV');
UPDATE `new`.`country_data` SET `Country` = 'Turkey' WHERE (`ISO3
   ` = 'TUR');
```

Code Listing 4: The function that gets tables from database is using these queries

```
SELECT Country, co2_prod_{date} FROM {table}
SELECT Country, gnipc_{date} FROM {table}
SELECT Country, hdi_{date} FROM {table}
SELECT Country, le_{date} FROM {table}
```

Code Listing 5: insert.py query as return value

```
/*
def insert(table, request):
    form_keys = request.form.keys()
    columns = list(form_keys)
    columns = [col for col in form_keys if request.form.get(col,
       '').strip()]  # Filter out empty or None values

    # Construct the SQL query dynamically based on the available
       data
    columns_values = [(col, request.form.get(col, "NULL")) for
       col in columns if col in form_keys]

    # Surround string values with single quotes in the VALUES
       clause
    formatted_values = [f"'{val}'" if isinstance(val, str) else
       val for _, val in columns_values]
*/
INSERT INTO {table} ({', '.join(col for col, _ in columns_values)
   }) VALUES ({', '.join(str(val) for val in formatted_values)}
```

Code Listing 6: update.py query as return value

```
/*
def update(table, request):
    # Get WHERE condition
    where_column = request.form.get('where_column')
    where_value = request.form.get('where_value')

    # Construct WHERE clause
    where_clause = f"{where_column} = '{where_value}'"

    # Get update columns and values
    update_columns = [key for key in request.form.keys() if key.
       startswith('update_column_')]
    new_columns = [request.form.get(key) for key in
       update_columns]
    new_values = [request.form.get(f'new_value_{index}') for
       index in range(1, len(update_columns) + 1)]

    # Construct SET clause
    set_clause = ', '.join([f"{column} = '{value}'" for column,
        value in zip(new_columns, new_values)])
*/
UPDATE {table} SET {set_clause} WHERE {where_clause}
```

Code Listing 7: delete.py query as return value

```
/*
def delete(table, request):
    column = request.form['column']
    txtvalue = request.form['txtvalue']
*/
DELETE FROM {table} WHERE {column}='{txtvalue}';
```

Code Listing 8: Search function creates query

```
/*
query = "SELECT * FROM "
query += f"`{schema}`"
query += f" WHERE country LIKE '%{country}%'"
query += f" ORDER BY `{column}` {'ASC' if order == 'asc' else '
   DESC'}"
*/

SELECT * FROM `{schema}` WHERE country LIKE '%{country}%' ORDER
   BY `{column}` {'ASC' if order == 'asc' else 'DESC'}
```

3.2 Source Code Outlines

3.2.1 app.py

```
#Imports
from flask import Flask, render_template, request, send_file,
                                    redirect
import mysql.connector
import os
import plotly.express as px
import pandas as pd
from insert import insert
from delete import delete
from update import update
```

```
# MySQL configuration
db_config = {
    'host': 'localhost',
    'user': 'root',
    'password': 'Fdfgbdhgb1.',
    'database': 'dbproj',
}

# Create a connection to MySQL
conn = mysql.connector.connect(**db_config)

# Create a cursor object to interact with the database
cursor = conn.cursor()
```

```
# World Map Page
@app.route('/world_map')
def world_map():
    conn = mysql.connector.connect(**db_config)
    cursor = conn.cursor()
    table = request.args.get('table')
    date = request.args.get('date')
    df = get_data_from_database(table, date)
    cursor.close()
    conn.close()



    if df.empty:
        return 'No data available.'

    # Adjust necessary values for map creation
    fig = px.choropleth(df, locations='Country', locationmode='
                                         country names', color='value
                                         ', template='plotly',
                                         color_continuous_scale='
                                         Viridis')
    fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
    return render_template('world_map.html', plot=fig.to_html(
                                         full_html=False))
```

```
# Project Page
@app.route('/project')
def project():
    pdf_directory = os.path.join(os.path.dirname(__file__), 'docs')
    pdf_path = os.path.join(pdf_directory, 'Project.pdf')
    return send_file(pdf_path, as_attachment=False)
```

```
# Ordering the lists
@app.route('/list/<schema>/search', methods=['GET', 'POST'])
@app.route('/list/<schema>/search/<country>')
@app.route('/list/<schema>/search/<country>/<column>/<order>')
def search(schema=None, column=None, order=None, country=None):
    conn = mysql.connector.connect(**db_config)
    cursor = conn.cursor()

    next_order = 'asc'
    query = "SELECT * FROM "
    columns = None
    results = None
    if request.method == 'POST':
        country = request.form.get('country')
        url = f'/list/{schema}/search/{country}'
        return redirect(url)

    if schema:
        query += f"`{schema}`"

    if country:
        query += f" WHERE country LIKE '%{country}%'"
        print(query)

    if column and order:
        column = column.strip()
        query += f" ORDER BY `{column}` {'ASC' if order == 'asc'
                                                    else 'DESC'}"
        next_order = 'desc' if order == 'asc' else 'asc'

    if schema:
        cursor.execute(query)
        results = cursor.fetchall()
        columns = [i[0] for i in cursor.description]

    cursor.close()
    conn.close()

    return render_template('list.html', schema=schema , columns=
                                        columns, results=results ,
                                        sorted_column=column,
                                        sorted_order=order,
                                        next_order=next_order,
                                        country=country)
```

```
# Inserting
@app.route('/edit/<table>/insert', methods=['GET', 'POST'])
```

```python
def insertf(table=None):
    post = None # This variable is for insert.html to show related content
    if request.method == 'GET':
        return render_template('insert.html', table=table, post=post)
    if request.method == 'POST':
        conn = mysql.connector.connect(**db_config)
        cursor = conn.cursor()
        post = 1
        query = insert(table, request) # SQL query is getting from insert.py
        print(query)
        cursor.execute(query)
        conn.commit()
        cursor.close()
        conn.close()
        return render_template('insert.html', table=table, post=post)
```

```python
# Updating
@app.route('/edit/<table>/update', methods=['GET', 'POST'])
def updatef(table=None):
    post = None
    if request.method == 'GET':
        return render_template('update.html', table=table, post=post)
    if request.method == 'POST':
        conn = mysql.connector.connect(**db_config)
        cursor = conn.cursor()
        post = 1
        query = update(table, request) # SQL query is getting from update.py
        print(query)
        cursor.execute(query)
        conn.commit()
        cursor.close()
        conn.close()
        return render_template('update.html', table=table, post=post
                                    )
```

```
# Deleting
@app.route('/edit/<table>/delete', methods=['GET', 'POST'])
def deletef(table=None):
    post = None
    if request.method == 'GET':
        return render_template('delete.html', table=table, post=post
                                    )
    if request.method == 'POST':
        conn = mysql.connector.connect(**db_config)
        cursor = conn.cursor()
        post = 1
        query = delete(table, request) # SQL query is getting from
                                    delete.py
        print(query)
        cursor.execute(query)
        conn.commit()
        cursor.close()
        conn.close()
        return render_template('delete.html', table=table, post=post
                                    )
```

3.2.2 delete.py

```
def delete(table, request):
    column = request.form['column']
    txtvalue = request.form['txtvalue']
    query = f"DELETE FROM {table} WHERE {column}='{txtvalue}';"
    return query
```

3.2.3 insert.py

```
def insert(table, request):
    form_keys = request.form.keys()
    columns = list(form_keys)
    columns = [col for col in form_keys if request.form.get(col, '')
                                        .strip()]  # Filter out
                                        empty or None values

    # Construct the SQL query dynamically based on the available
                                        data
    columns_values = [(col, request.form.get(col, "NULL")) for col
                                        in columns if col in
                                        form_keys]

    # Surround string values with single quotes in the VALUES clause
    formatted_values = [f"'{val}'" if isinstance(val, str) else val
                                        for _, val in columns_values
                                        ]

    # Construct the SQL query
    query = f"INSERT INTO {table} ({', '.join(col for col, _ in
                                        columns_values)}) VALUES ({
                                        ', '.join(str(val) for val
                                        in formatted_values)})"

    return query
```

3.2.4 update.py

```
def update(table, request):
    # Get WHERE condition
    where_column = request.form.get('where_column')
    where_value = request.form.get('where_value')

    # Construct WHERE clause
```

```
        where_clause = f"{where_column} = '{where_value}'"

        # Get update columns and values
        update_columns = [key for key in request.form.keys() if key.
                                            startswith('update_column_')
                                            ]
        new_columns = [request.form.get(key) for key in update_columns]
        new_values = [request.form.get(f'new_value_{index}') for index
                                            in range(1, len(
                                            update_columns) + 1)]

        # Construct SET clause
        set_clause = ', '.join([f"{column} = '{value}'" for column,
                                            value in zip(new_columns,
                                            new_values)])

        # Construct the SQL query
        query = f"UPDATE {table} SET {set_clause} WHERE {where_clause}"

        return query
```

3.2.5 layout.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
       initial-scale=1.0">
    <title>{% block title %}Default Title{% endblock %}</title>
    <link rel="stylesheet" href="/static/css/style.css">
    {% block head %}{% endblock %}
</head>
<body>
```

```
    <header>
        <nav>
            <ul>
                <li><a href="/">Home</a></li>
                <li><a href="/world">Map</a></li>
                <li><a href="/project">Project</a></li>
                <li><a href="/list">List</a></li>
                <li><a href="/edit">Edit</a></li>
            </ul>
        </nav>
    </header>

    <main>
        {% block content %}

        {% endblock %}
    </main>

    {% block scripts %}

    {% endblock %}
</body>
</html>
```

3.2.6 home.html

```
<div class="home-container">
    <div class="team-section">
        <h1>DATABASE PROJECT</h1>
        <h2>Global Socio-Economic & Environmental Indicators</h2>
        <ul class="team-list">
            <li>Hakan Duran 150200091</li>
            <li>Kerem Er 820210313</li>
            <li>Rengin Helin Yalcin 150220767</li>
            <li>Oguz Eren Kacar 150200018</li>
        </ul>
    </div>
    <div class="table-section">
        <ul class="table-list">
            <li><a href="/list/co2_production">co2_production</a></
               li>
            <li><a href="/list/gross_national_income_per_capital">
               gross_national_income_per_capital</a></li>
            <li><a href="/list/human_development_index">
               human_development_index</a></li>
            <li><a href="/list/life_expectancy_by_birth">
               life_expectancy_by_birth</a></li>
        </ul>
    </div>
</div>
```

3.2.7 project.html

```
{% extends "layout.html" %}
{% block title %}PROJECT PROPOSAL{% endblock %}
{% block content %}
<iframe src="/project" width="100%" height="600px"></iframe>
{% endblock %}
```

3.2.8 world_map.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
       initial-scale=1.0">
    <title>World Map</title>
    {{ plot|safe }}
```

```
</head>
<body>
            <h1><a href="/world">Another world map selection</a></h1
               >
</body>
</html>
```

3.2.9 world.html

```
<h2> Select world map and date: </h2>
<form action="/world_map" method="get">
    <label for="table">Table:</label>
    <select id="table" name="table">
        <option value="" selected disabled>Select an option</option>
        <option value="co2_production">co2_production</option>
        <option value="gross_national_income_per_capita">
           gross_national_income_per_capita</option>
        <option value="human_development_index">
           human_development_index</option>
        <option value="life_expectancy_by_birth">
           life_expectancy_by_birth</option>
    </select>

    <label for="date">Date:</label>
    <select id="date" name="date">
        <option value="" selected disabled>Select an option</option>
        <option value="1990">1990</option>
        <option value="1991">1991</option>
        <option value="1992">1992</option>
        .
        .
        .
        <option value="2020">2020</option>
        <option value="2021">2021</option>
    </select>

    <button type="submit">Submit</button>
</form>
```

3.2.10 edit.html

```
<h1> SQL CUD - Insert, Update, Delete Operations </h1>

{% if table == None %}
Please select a table and an operation:
<form action="/edit" method="post">
    <label for="table">Table:</label>
    <select id="table" name="table">
        <option value="" selected disabled>Select an option</option>
        <option value="co2_production">co2_production</option>
        <option value="gross_national_income_per_capital">gross_national_income_per_capital</option>
        <option value="human_development_index">human_development_index</option>
        <option value="life_expectancy_by_birth">life_expectancy_by_birth</option>
    </select>
    <label for="cud">Operation:</label>
    <select id="cud" name="cud">
        <option value="" selected disabled>Select an option</option>
        <option value="insert">insert</option>
        <option value="update">update</option>
        <option value="delete">delete</option>
    </select>
    <input type="submit" value="Submit">
</form>

{% else %}
Please select  an operation:
<form action="/edit/{{ table }}" method="post">
```

```
    <label for="cud">Operation:</label>
    <select id="cud" name="cud">
        <option value="" selected disabled>Select an option</option>
        <option value="insert">insert</option>
        <option value="update">update</option>
        <option value="delete">delete</option>
    </select>
    <input type="submit" value="Submit">
</form>
```

3.2.11 update.html

```
{% extends "layout.html" %}

{% block title %} Update on {{table}} {% endblock %}

{% block content %}
<h1> Operation: Update on {{table}} </h1>

{% if post == None %}

    {% if table == "co2_production" %}
        <!-- Combined form for WHERE condition and UPDATE columns -->
        <form action="/edit/co2_production/update" method="POST">

            <!-- WHERE condition section -->
            <h3>Updated records should have:</h3>
            <label for="where_column">Table:</label>
            <select id="where_column" name="where_column">
                <!-- Include options for columns -->
                <option value="" selected disabled>Select an option</option>
                <option value="ISO3">ISO3</option>
                <option value="Country">Country</option>
                <option value="hdicode">hdicode</option>
                <option value="region">region</option>
                <option value="hdi_rank_2021">hdi_rank_2021</option>
                <option value="co2_prod_1990">co2_prod_1990</option>
                <option value="co2_prod_1991">co2_prod_1991</option>
                .
                .
                <option value="co2_prod_2020">co2_prod_2020</option>
                <option value="co2_prod_2021">co2_prod_2021</option>
            </select>
            <label for="where_value">is (is equal to):</label>
            <input type="text" id="where_value" name="where_value">

            <hr>

            <!-- UPDATE columns section -->
            <h3>Update Columns:</h3>
            <div id="update_columns_container">
                <div class="update_column_row">
                    <label for="update_column_1">Select Column to Update:</label>
                    <select name="update_column_1">
                        <!-- Include options for columns -->
                        <option value="" selected disabled>Select an option</option>
                        <option value="ISO3">ISO3</option>
                        <option value="Country">Country</option>
                        <option value="hdicode">hdicode</option>
                        <option value="region">region</option>
                        <option value="hdi_rank_2021">hdi_rank_2021</option>
                        <option value="co2_prod_1990">co2_prod_1990</option>
                        <option value="co2_prod_1991">co2_prod_1991</option>
                        .
                        .
                        <option value="co2_prod_2020">co2_prod_2020</option>
                        <option value="co2_prod_2021">co2_prod_2021</option>
                    </select>
                    <label for="new_value_1">Enter New Value:</label>
                    <input type="text" name="new_value_1">
                </div>
            </div>

            <button type="button" onclick="addUpdateField()">Add Update Field</button>

            <hr>

            <!-- Button to submit the combined form -->
            <input type="submit" value="Update Record">

        </form>

        <script>
            function addUpdateField() {
                var container = document.getElementById('update_columns_container');
                var rowCount = container.getElementsByClassName('update_column_row').length + 1;

                var newRow = document.createElement('div');
                newRow.className = 'update_column_row';

                newRow.innerHTML = `
                    <label for="update_column_${rowCount}">Select
```

```
                       Column to Update:</label>
                    <select name="update_column_${rowCount}">
                        <!-- Include options for columns -->
                        <option value="" selected disabled>Select an option</option>
                        <option value="ISO3">ISO3</option>
                        <option value="Country">Country</option>
                        <option value="hdicode">hdicode</option>
                        <option value="region">region</option>
                        <option value="hdi_rank_2021">hdi_rank_2021</option>
                        <option value="co2_prod_1990">co2_prod_1990</option>
                        <option value="co2_prod_1991">co2_prod_1991</option>
                        .
                        .
                        <option value="co2_prod_2020">co2_prod_2020</option>
                        <option value="co2_prod_2021">co2_prod_2021</option>
                    </select>

                    <label for="new_value_${rowCount}">Enter New Value:</label>
                    <input type="text" name="new_value_${rowCount}">
                `;

                container.appendChild(newRow);
            }
        </script>

    {% elif table == "gross_national_income_per_capital" %}
        <!-- Combined form for WHERE condition and UPDATE columns -->
        <form action="/edit/gross_national_income_per_capital/update" method="POST">

            <!-- WHERE condition section -->
            <h3>Updated records should have:</h3>
            <label for="where_column">Table:</label>
            <select id="where_column" name="where_column">
                <!-- Include options for columns -->
                <option value="" selected disabled>Select an option</option>
                <option value="ISO3">ISO3</option>
                <option value="Country">Country</option>
                <option value="hdicode">hdicode</option>
                <option value="region">region</option>
                <option value="hdi_rank_2021">hdi_rank_2021</option>
                <option value="gnipc_1990">gnipc_1990</option>
                <option value="gnipc_1991">gnipc_1991</option>
                .
                .
                <option value="gnipc_2020">gnipc_2020</option>
                <option value="gnipc_2021">gnipc_2021</option>
            </select>
            <label for="where_value">is (is equal to):</label>
            <input type="text" id="where_value" name="where_value">

            <hr>

            <!-- UPDATE columns section -->
            <h3>Update Columns:</h3>
            <div id="update_columns_container">
                <div class="update_column_row">
                    <label for="update_column_1">Select Column to Update:</label>
                    <select name="update_column_1">
                        <!-- Include options for columns -->
                        <option value="" selected disabled>Select an option</option>
                        .
                        .
                    </select>
                    <label for="new_value_1">Enter New Value:</label>
                    <input type="text" name="new_value_1">
                </div>
            </div>

            <button type="button" onclick="addUpdateField()">Add Update Field</button>

            <hr>

            <!-- Button to submit the combined form -->
            <input type="submit" value="Update Record">

        </form>

        <script>
            function addUpdateField() {
                var container = document.getElementById('update_columns_container');
                var rowCount = container.getElementsByClassName('update_column_row').length + 1;

                var newRow = document.createElement('div');
                newRow.className = 'update_column_row';

                newRow.innerHTML = `
                    <label for="update_column_${rowCount}">Select Column to Update:</label>
                    <select name="update_column_${rowCount}">
```

```
                        <!-- Include options for columns -->
                        <option value="" selected disabled>Select an option</option>
                        <option value="ISO3">ISO3</option>
                        <option value="Country">Country</option>
                        <option value="hdicode">hdicode</option>
                        <option value="region">region</option>
                        <option value="hdi_rank_2021">hdi_rank_2021</option>
                        <option value="gnipc_1990">gnipc_1990</option>
                        <option value="gnipc_1991">gnipc_1991</option>
                        .
                        .
                        <option value="gnipc_2020">gnipc_2020</option>
                        <option value="gnipc_2021">gnipc_2021</option>
                    </select>

                    <label for="new_value_${rowCount}">Enter New Value:</label>
                    <input type="text" name="new_value_${rowCount}">
                `;

                container.appendChild(newRow);
            }
        </script>
    {% elif table == "human_development_index" %}
        .
        .
    {% elif table == "life_expectancy_by_birth" %}
        .
        .
    {% else %}
    <h1> ERROR: TABLE IS NOT EXIST. SELECTED TABLE IS: {{ table }} <
       /h1>
    {% endif %}


{% else %}

<h1> SUCCESSFUL OPERATION ON {{table}} </h1>

{% endif %}

{% endblock %}
```

3.2.12 insert.html

```
{% extends "layout.html" %}

{% block title %} Insert on {{table}} {% endblock %}

{% block content %}
<h1> Operation: Insert on {{table}} </h1>

{% if post == None %}
    {% if table == "co2_production" %}
        <form action="/edit/co2_production/insert" method="POST">
            <label for="ISO3">ISO3:</label><br>
            <input type="text" id="ISO3" name="ISO3"><br>

            <label for="Country">Country:</label><br>
            <input type="text" id="Country" name="Country"><br>

            <label for="hdicode">hdicode:</label><br>
            <input type="text" id="hdicode" name="hdicode"><br>

            <label for="region">region:</label><br>
            <input type="text" id="region" name="region"><br>

            <label for="hdi_rank_2021">hdi_rank_2021:</label><br>
            <input type="text" id="hdi_rank_2021" name="hdi_rank_2021"><br>

            <label for="co2_prod_1990">co2_prod_1990:</label><br>
            <input type="text" id="co2_prod_1990" name="co2_prod_1990"><br>

            <label for="co2_prod_1991">co2_prod_1991:</label><br>
            <input type="text" id="co2_prod_1991" name="co2_prod_1991"><br>

            .
            .
            .

            <label for="co2_prod_2020">co2_prod_2020:</label><br>
            <input type="text" id="co2_prod_2020" name="co2_prod_2020"><br>

            <label for="co2_prod_2021">co2_prod_2021:</label><br>
            <input type="text" id="co2_prod_2021" name="co2_prod_2021"><br>

            <input type="submit" value="Submit">
        </form>
    {% elif table == "gross_national_income_per_capital" %}
        <form action="/edit/gross_national_income_per_capital/insert" method="POST">
            <label for="ISO3">ISO3:</label><br>
            <input type="text" id="ISO3" name="ISO3"><br>

```

```html
            <label for="Country">Country:</label><br>
            <input type="text" id="Country" name="Country"><br>

            <label for="hdicode">hdicode:</label><br>
            <input type="text" id="hdicode" name="hdicode"><br>

            <label for="region">region:</label><br>
            <input type="text" id="region" name="region"><br>

            <label for="hdi_rank_2021">hdi_rank_2021:</label><br>
            <input type="text" id="hdi_rank_2021" name="hdi_rank_2021"><br>

            <label for="gnipc_1990">gnipc_1990:</label><br>
            <input type="text" id="gnipc_1990" name="gnipc_1990"><br>

            <label for="gnipc_1991">gnipc_1991:</label><br>
            <input type="text" id="gnipc_1991" name="gnipc_1991"><br>

            .
            .
            .

            <label for="gnipc_2020">gnipc_2020:</label><br>
            <input type="text" id="gnipc_2020" name="gnipc_2020"><br>

            <label for="gnipc_2021">gnipc_2021:</label><br>
            <input type="text" id="gnipc_2021" name="gnipc_2021"><br>

            <input type="submit" value="Submit">
        </form>
```

```

    {% elif table == "human_development_index" %}
        .
        .

    {% elif table == "life_expectancy_by_birth" %}
        .
        .

    {% else %}
    <h1> ERROR: TABLE IS NOT EXIST. SELECTED TABLE IS: {{ table }} </h1>
    {% endif %}

{% else %}

<h1> SUCCESSFUL OPERATION ON {{table}} </h1>

{% endif %}

{% endblock %}
```

3.2.13 delete.html

```
{% extends "layout.html" %}

{% block title %} Delete on {{table}} {% endblock %}

{% block content %}
<h1> Operation: Delete on {{table}} </h1>

{% if post == None %}

    Deleted records should have:

```

```
    {% if table == "co2_production" %}
        <form action="/edit/co2_production/delete" method="post" class="delete-f">
            <label for="column">Table:</label>
            <select id="column" name="column">
                <option value="" selected disabled>Select an option</option>
                <option value="ISO3">ISO3</option>
                <option value="Country">Country</option>
                <option value="hdicode">hdicode</option>
                <option value="region">region</option>
                <option value="hdi_rank_2021">hdi_rank_2021</option>
                <option value="co2_prod_1990">co2_prod_1990</option>
                <option value="co2_prod_1991">co2_prod_1991</option>
                .
                .
                <option value="co2_prod_2020">co2_prod_2020</option>
                <option value="co2_prod_2021">co2_prod_2021</option>
            </select>
            <label for="txtvalue">is (is equal to)</label><br>
            <input type="text" id="txtvalue" name="txtvalue"><br>
            <input type="submit" value="Submit">
        </form>
    {% elif table == "gross_national_income_per_capital" %}
        <form action="/edit/gross_national_income_per_capital/delete" method="post" class="delete-f">
            <label for="column">Table:</label>
            <select id="column" name="column">
                <option value="" selected disabled>Select an option</option>
                <option value="ISO3">ISO3</option>
                <option value="Country">Country</option>
                <option value="hdicode">hdicode</option>
                <option value="region">region</option>
                <option value="hdi_rank_2021">hdi_rank_2021</option>
                <option value="gnipc_1990">gnipc_1990</option>
                <option value="gnipc_1991">gnipc_1991</option>
                .
                .
                <option value="gnipc_2020">gnipc_2020</option>
                <option value="gnipc_2021">gnipc_2021</option>
            </select>
            <label for="txtvalue">is (is equal to)</label><br>
            <input type="text" id="txtvalue" name="txtvalue"><br>
            <input type="submit" value="Submit">
        </form>
    {% elif table == "human_development_index" %}
        .
        .
    {% elif table == "life_expectancy_by_birth" %}
        .
        .
    {% else %}
    <h1> ERROR: TABLE IS NOT EXIST. SELECTED TABLE IS: {{ table }} </h1>
    {% endif %}


{% else %}

<h1> SUCCESSFUL OPERATION ON {{table}} </h1>

{% endif %}

{% endblock %}
```

3.2.14 style.css

```
/* static/css/style.css */

```

```css
body {
    font-family: 'Arial', sans-serif;
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}
header {
    background-color: #333;
    color: #fff;
    padding: 5px 0;
}
nav ul {
    list-style: none;
    padding: 0;
    margin: 0;
    display: flex;
    justify-content: center;
}
nav ul li {
    padding: 0 20px;
}
nav ul li a {
    color: #fff;
    text-decoration: none;
    transition: color 0.3s ease;
}
nav ul li a:hover {
    color: #04AA6D;
}
main {
    padding: 20px;
}
table {
    width: 100%;
    border-collapse: collapse;
```

```
}
th, td {
    padding: 8px;
    text-align: left;
    border-bottom: 1px solid #ddd;
    white-space: nowrap;
}
th {
    background-color: #04AA6D;
    color: white;
}

.content-container {
    width: 80%;
    margin: auto;
    display: flex;
    flex-direction: column;
    align-items: center;
}

.table-container {
    width: 100%;
    margin-top: 20px;
    overflow-x: auto;
    border: 1px solid #ccc;
    padding: 10px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.2);
}

.search-container {
    margin-bottom: 15px;
    text-align: center;
}

.search-form {
```

```css
    display: inline-block;
    margin-bottom: 15px;
    margin-right: 10px;
}

.search-input {
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 5px;
    font-size: 16px;
}

.search-button {
    padding: 10px 20px;
    background-color: #04AA6D;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.search-button:hover {
    background-color: #037a4d;
}

.search-input, .search-button {
    vertical-align: middle;
}

.crud-container {
    display: inline-block;
    margin-bottom: 15px;
    margin-right: 10px;
}
```

```

.insert-form, .delete-form, .update-form {
    display: inline-block;
}

.insert-button {
    display: inline-block;
    padding: 10px 20px;
    background-color: #e8dc01;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.insert-button:hover {
    background-color: #596b13 ;
}

.delete-button {
    display: inline-block;
    padding: 10px 20px;
    background-color: #e80101;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.delete-button:hover {
    background-color: #752323 ;
}

```

```css
.update-button {
    display: inline-block;
    padding: 10px 20px;
    background-color: #109eea;
    color: white;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.update-button:hover {
    background-color: #10188d ;
}

.insert-button, .delete-button, .update-button {
    vertical-align: middle;
}

.header-link {
    color: inherit;
    text-decoration: none;
    display: flex;
    align-items: center;
}

.header-title {
    display: inline-block;
    margin-right: 10px;
}

.ascending::after, .descending::after {
    content: '';
    display: inline-block;
    width: 0;
    height: 0;
    border-left: 5px solid transparent;
    border-right: 5px solid transparent;
    border-bottom: 5px solid white;

}

.descending::after {
    border-bottom: none;
    border-top: 5px solid white;
}

.home-container {
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 50px;
    background-color: #4CAF50;
    color: white;
    height: calc(100vh - 60px);
}

.team-section h1 {
    margin-bottom: 20px;
}

.team-list,
.table-list {
    list-style: none;
    padding: 0;
}

.team-list li,
.table-list li {
    margin-bottom: 10px;
    font-size: 1.2em;
}

.table-list li a {
    color: white;
    text-decoration: none;
    background-color: #333;
    padding: 10px;
    border-radius: 5px;
    transition: background-color 0.3s;
}

.table-list li a:hover {
    background-color: #ddd;
    color: #333;
}

@media (max-width: 768px) {
    .home-container {
        flex-direction: column;
        text-align: center;
    }

    .team-section, .table-section {
        margin-bottom: 20px;
    }
}

.delete-f {
    display: flex;
    flex-direction: row;
    align-items: center;
}

.delete-f label,
.delete-f select,
.delete-f input {
    margin-right: 10px;
}
```