DOCUMENTATION

Carbon footprint calculation page

carbon_footprint_id	user_email	electricity	gas	oil	car	flights	recycle_news	recycle_tin
Filter	Filter	Filter	Fil...	Fil...	Fil...	Filter	Filter	Filter
1	dia@gma...	123	1...	1...	1...	123	yes	no

This was the initial database whose purpose was to gain the information that the user inputed in the form and save it.

However, the issue was that this wouldn't be able to calculate and save the carbon footprint scores.

This query is executed to calculate and update the carbon score of the user this query follows a specific formula and involves input from different factors that impact carbon footprint.

SQL Query :

```sql
ALTER TABLE carbon_score
ADD COLUMN score NUMBER
```

This query alters the table 'carbon_score' by adding the column 'score' where the user's carbon score will be stored.

```sql
UPDATE carbon_score
SET score =
    (electricity * 105) + (gas * 105) + (oil * 113) +(car * 0.79) +
    (CASE WHEN flights <= 4 THEN flights * 1100 ELSE 0 END) +
    (CASE WHEN flights > 4 THEN flights * 4400 ELSE 0 END) +
    (CASE WHEN recycle_news = 'no' THEN 184 ELSE 0 END) +
    (CASE WHEN recycle_tin = 'no' THEN 166 ELSE 0 END);
```

This equation will output how much the user contributes to the carbon footprint(e.g. electricity = £100 * 105 = 10500 this will mean the user contributes 10500 lb of CO_2 via their electricity usage per year). I used a case statement for the condition of the user input for recycling.

Here is an exemplary user score :

carbon_footprint_id	user_email	score
1	dia@gmail.com	581192.17

To show in the actual front end I planned to create individual pages however it will be much easier for users to go to what they need in less clicks as possible so I created an option in the navigation bar where the user can see their carbon scores.

```python
@app.route("/dashboard")
def dashboard():
    user_email = get_user_email_from_session()
    carbon_scores = list_carbon_score_by_user( filename: "customer_info.db", user_email)
    bookings = list_booking( filename: "customer_info.db", user_email)
    return render_template( template_name_or_list: "dashboard.html", carbon_scores=carbon_scores, bookings=bookings)
```

So i created a page called dashboard