```
from flask import Flask, render_template_string, request

app = Flask(__name__)

# HTML template
HTML_TEMPLATE = """
<!DOCTYPE html>
<html lang="zh">
<head>
  <meta charset="UTF-8">
  <title>LCC STUDENT personal carbon footprint calculator</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 40px;
      background-color: #f9f9f9;
    }
    h1 {
      color: #2a7ae2;
      text-align: center;
    }
    h3 {
      color: #444;
    }
    label {
      display: block;
      margin-top: 10px;
      font-size: 14px;
    }
    input[type="number"], input[type="checkbox"], select {
      padding: 5px;
    }
    input[type="number"] {
      width: 100px;
      border: 1px solid #ccc;
      border-radius: 4px;
    }
    button {
      margin-top: 20px;
      padding: 10px 20px;
      background-color: #2a7ae2;
      color: white;
      border: none;
      border-radius: 4px;
      cursor: pointer;
    }
    button:hover {
      background-color: #1a5bb2;
    }
    .result {
      margin-top: 30px;
      font-weight: bold;
      background-color: #fff;
      padding: 15px;
      border: 1px solid #ddd;
      border-radius: 4px;
    }
```

```
    .section {
      margin-bottom: 20px;
      background-color: #fff;
      padding: 15px;
      border: 1px solid #ddd;
      border-radius: 4px;
    }
    .transport-type {
      margin-top: 15px;
      padding-top: 10px;
      border-top: 1px dashed #eee;
    }
    .month-selector {
      margin-left: 10px;
      font-size: 12px;
    }
    .footer {
      margin-top: 40px;
      font-size: 12px;
      color: #666;
      line-height: 1.5;
      border-top: 1px solid #eee;
      padding-top: 15px;
    }
  </style>
</head>
<body>
  <h1>LCC STUDENT personal carbon footprint  calculatorper  month</h1>
  <form method="POST">
    <div class="section">
      <h3 ></h3>
      <label      >
        <input type="number" name="numClothes" value="{{ numClothes }}"
             min="0">
      </label>
      <label      >
        <input type="checkbox" name="includeWashing" {{ 'checked' if
            includeWashing else '' }}>
      </label>
      <label       >
        <input type="number" name="washesPerCloth" value="{{
            washesPerCloth }}" min="0" {{ 'disabled' if not
            includeWashing else '' }}>
      </label>
    </div>

    <div class="section">
      <h3 ></h3>
      <label      >
        <input type="number" name="mealsVeggie" value="{{ mealsVeggie
            }}" min="0">
      </label>
      <label      >
        <input type="number" name="mealsMeat" value="{{ mealsMeat }}"
            min="0">
      </label>
      <label     >
```

```
        <input type="number" name="takeoutMeals" value="{{ takeoutMeals }}" min="0">
      </label>
    </div>

    <div class="section">
      <h3 ></h3>
      <label    > kWh
        <input type="number" name="electricityKwh" value="{{ electricityKwh }}" min="0">
      </label>
      <label       >
        <input type="number" name="dormArea" value="{{ dormArea }}" min="0">
      </label>
      <label       >
        <input type="number" name="computerHours" value="{{ computerHours }}" min="0">
      </label>
      <label  >  W60W
        <input type="number" name="computerWattage" value="{{ computerWattage }}" min="0">
      </label>
    </div>

    <div class="section">
      <h3 ></h3>

      <div class="transport-type">
        <h4  ></h4>
        <label      >/
          <input type="number" name="busTrips" value="{{ busTrips }}" min="0">
        </label>
        <label       >/
          <input type="number" name="busMetroAvgDistance" value="{{ busMetroAvgDistance }}" min="0">
        </label>
        <label      >
          <input type="number" name="bikeTrips" value="{{ bikeTrips }}" min="0">
        </label>
        <label       >
          <input type="number" name="bikeAvgDistance" value="{{ bikeAvgDistance }}" min="0">
        </label>
      </div>

      <div class="transport-type">
        <h4  >/</h4>
        <label     >
          <input type="number" name="taxiTrips" value="{{ taxiTrips }}" min="0">
        </label>
        <label       >
          <input type="number" name="taxiAvgDistance" value="{{ taxiAvgDistance }}" min="0">
```

```
        </label>
        <label  >
          <select name="taxiType">
            <option value="gasoline" {{ 'selected' if taxiType == '
                gasoline' else ''   }}>1.6 L</option>
            <option value="diesel" {{ 'selected' if taxiType == 'diesel
                ' else ''   }}></option>
            <option value="ev" {{ 'selected' if taxiType == 'ev' else
                ''   }}></option>
          </select>
        </label>
      </div>

      <div class="transport-type">
        <h4  >/</h4>
        <label     >
          <input type="number" name="shuttleTrips" value="{{
              shuttleTrips }}" min="0">
        </label>
        <label      >
          <input type="number" name="ebikeTrips" value="{{ ebikeTrips
              }}" min="0">
        </label>
        <label       >
          <input type="number" name="ebikeAvgDistance" value="{{
              ebikeAvgDistance }}" min="0">
        </label>
      </div>

      <div class="transport-type">
        <h4  ></h4>
        <label   >
          <select name="hsrMonth" class="month-selector">
            <option value="0" {{ 'selected' if hsrMonth == 0 else ''
                  }}></option>
            <option value="1" {{ 'selected' if hsrMonth == 1 else ''
                }}>1</option>
            <option value="2" {{ 'selected' if hsrMonth == 2 else ''
                }}>2</option>
            <option value="3" {{ 'selected' if hsrMonth == 3 else ''
                }}>3</option>
            <option value="4" {{ 'selected' if hsrMonth == 4 else ''
                }}>4</option>
            <option value="5" {{ 'selected' if hsrMonth == 5 else ''
                }}>5</option>
            <option value="6" {{ 'selected' if hsrMonth == 6 else ''
                }}>6</option>
            <option value="7" {{ 'selected' if hsrMonth == 7 else ''
                }}>7</option>
            <option value="8" {{ 'selected' if hsrMonth == 8 else ''
                }}>8</option>
            <option value="9" {{ 'selected' if hsrMonth == 9 else ''
                }}>9</option>
            <option value="10" {{ 'selected' if hsrMonth == 10 else ''
                }}>10</option>
            <option value="11" {{ 'selected' if hsrMonth == 11 else ''
                }}>11</option>
```

```
            <option value="12" {{ 'selected' if hsrMonth == 12 else '' }}>12</option>
          </select>
        </label>
        <label      >
          <input type="number" name="hsrTrips" value="{{ hsrTrips }}" min="0">
        </label>
        <label       >
          <input type="number" name="hsrAvgDistance" value="{{ hsrAvgDistance }}" min="0">
        </label>
      </div>

      <div class="transport-type">
        <h4  ></h4>
        <label      >
          <input type="number" name="domesticFlights" value="{{ domesticFlights }}" min="0">
        </label>
        <label      >
          <input type="number" name="domesticFlightDistance" value="{{ domesticFlightDistance }}" min="0">
        </label>
        <label   >
          <select name="domesticCabin">
            <option value="economy" {{ 'selected' if domesticCabin == 'economy' else ''  }}></option>
            <option value="business" {{ 'selected' if domesticCabin == 'business' else ''    }}>/</option>
          </select>
        </label>

        <label      >
          <input type="number" name="internationalFlights" value="{{ internationalFlights }}" min="0">
        </label>
        <label      >
          <input type="number" name="internationalFlightDistance" value="{{ internationalFlightDistance }}" min="0">
        </label>
        <label   >
          <select name="internationalCabin">
            <option value="economy" {{ 'selected' if internationalCabin == 'economy' else ''   }}></option>
            <option value="business" {{ 'selected' if internationalCabin == 'business' else ''    }}>/</option>
          </select>
        </label>
      </div>
    </div>

    <div class="section">
      <h3  ></h3>
      <label     > kg
        <input type="number" name="recycledPlastic" value="{{
```

```
            recycledPlastic }}" min="0">
      </label>
      <label    > kg
        <input type="number" name="recycledPaper" value="{{
            recycledPaper }}" min="0">
      </label>
      <label     > kg
        <input type="number" name="recycledMetal" value="{{
            recycledMetal }}" min="0">
      </label>
      <label    > kg
        <input type="number" name="recycledCardboard" value="{{
            recycledCardboard }}" min="0">
      </label>
    </div>

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

  {% if result %}
  <div class="result">
    <h3   ></h3>
    <strong   ></strong>{{ "%.2f"|format(clothing) }} kg  COe<br>
    <strong   ></strong>{{ "%.2f"|format(food) }} kg  COe<br>
    <strong   ></strong>{{ "%.2f"|format(housing) }} kg  COe<br>
    <h4    ></h4>•
     {{ "%.2f"|format(publicTransport) }} kg  COe<br>•
     /{{ "%.2f"|format(taxiTransport) }} kg  COe<br>•
     {{ "%.2f"|format(shuttleTransport) }} kg  COe<br>•
     {{ "%.2f"|format(ebikeTransport) }} kg  COe<br>•
     {{ "%.2f"|format(hsrTransport) }} kg  COe<br>•
     {{ "%.2f"|format(airTransport) }} kg  COe<br>
    <strong    ></strong>{{ "%.2f"|format(transport) }} kg  COe<br>
    <strong    ></strong>{{ "%.2f"|format(recycling) }} kg  COe<br>
    <br>
    <h3   >{{ "%.2f"|format(total) }} kg  COe</h3>
    <p    >{{ computerHours  }} × {{ computerWattage }}W = {{ "%.2f"|
        format(computerElectricity) }} kWh</p>
    <p    >{{ shuttleTrips  }} ×  12 = {{ shuttleKm  }}</p>
  </div>
  {% endif %}

  <div class="footer">
    <h3  ></h3>
    <p                                  ></p>
    <p                           ></p>
    <p           >               CPCD</p>
  </div>
</body>
</html>
"""

# Emission factors
EMISSION_FACTORS = {
    # Clothing
    "clothing_production": 7.0,  # kgCO2e per piece
    "clothing_transport": 0.5,   # kgCO2e per piece
```

```
    "clothing_washing_electricity": 0.2,  # kgCO2e per wash
    "clothing_washing_detergent": 0.1,    # kgCO2e per wash

    # Food
    "food_production_veggie": 0.6,  # kgCO2e per veggie meal
    "food_production_meat": 1.0,    # kgCO2e per meat meal
    "food_transport": 0.1,          # kgCO2e per meal
    "food_cooking": 0.1167,         # kgCO2e per meal
    "food_waste": 0.05,             # kgCO2e per meal
    "takeout_packaging": 1.6758,    # kgCO2e per takeout meal

    # Housing
    "housing_electricity": 0.5834,  # kgCO2e per kWh
    "housing_heating": 0.0,         # kgCO2e per m²

    # Transportation - Public
    "transport_bus_metro": 0.05,    # kgCO2e per km
    "transport_bike": 0.0,         # kgCO2e per km
    "transport_shuttle": 0.08,     # kgCO2e per km
    "transport_ebike": 0.02,       # kgCO2e per km
    "transport_hsr_electricity": 0.0188,  # kgCO2e per km

    # Transportation - Taxi
    "transport_taxi_gasoline": 0.25,   # kgCO2e per km
    "transport_taxi_diesel": 0.29,     # kgCO2e per km
    "transport_taxi_ev": 0.085,        # kgCO2e per km

    # Transportation - Flight
    "transport_flight_domestic_economy": 0.14,  # kgCO2e per km
    "transport_flight_domestic_business": 0.28, # kgCO2e per km
    "transport_flight_international_economy": 0.21,  # kgCO2e per km
    "transport_flight_international_business": 0.42,  # kgCO2e per km

    # Recycling
    "recycling_plastic": -1.5,    # kgCO2e per kg
    "recycling_paper": -0.8,      # kgCO2e per kg
    "recycling_metal": -2.0,      # kgCO2e per kg
    "recycling_cardboard": -0.992  # kgCO2e per kg
}

@app.route("/", methods=["GET", "POST"])
def carbon_footprint_calculator():
    # Default values
    defaults = {
        "numClothes": 0,
        "includeWashing": False,
        "washesPerCloth": 0,
        "mealsVeggie": 0,
        "mealsMeat": 0,
        "takeoutMeals": 0,
        "electricityKwh": 0,
        "dormArea": 0,
        "computerHours": 0,
        "computerWattage": 60,
        "busTrips": 0,
        "busMetroAvgDistance": 0,
        "bikeTrips": 0,
```

```
        "bikeAvgDistance": 0,
        "taxiTrips": 0,
        "taxiAvgDistance": 0,
        "taxiType": "gasoline",
        "shuttleTrips": 0,
        "ebikeTrips": 0,
        "ebikeAvgDistance": 0,
        "hsrMonth": 0,
        "hsrTrips": 0,
        "hsrAvgDistance": 0,
        "domesticFlights": 0,
        "domesticFlightDistance": 0,
        "domesticCabin": "economy",
        "internationalFlights": 0,
        "internationalFlightDistance": 0,
        "internationalCabin": "economy",
        "recycledPlastic": 0,
        "recycledPaper": 0,
        "recycledMetal": 0,
        "recycledCardboard": 0,
    }

    result = None
    calculation = {}

    if request.method == "POST":
        # Get form data with defaults
        form_data = {}
        for key in defaults:
            if key == "includeWashing":
                form_data[key] = key in request.form
            else:
                try:
                    form_data[key] = float(request.form.get(key,
                        defaults[key]))
                except ValueError:
                    form_data[key] = request.form.get(key, defaults[key
                        ])

        # Calculate emissions
        calculation = calculate_emissions(form_data)
        result = True

    # Merge defaults with form data and calculation results
    context = {**defaults, **(form_data if request.method == "POST"
        else {}), **(calculation if result else {})}
    context["result"] = result

    return render_template_string(HTML_TEMPLATE, **context)

def calculate_emissions(data):
    # Clothing
    clothes = float(data["numClothes"])
    include_washing = data["includeWashing"]
    washes_per_cloth = float(data["washesPerCloth"])

    clothing = clothes * (EMISSION_FACTORS["clothing_production"] +
```

```
        EMISSION_FACTORS["clothing_transport"])
    washing_electricity = 0

    if include_washing:
        total_washes = clothes * washes_per_cloth
        washing_electricity = total_washes * EMISSION_FACTORS["
            clothing_washing_electricity"]
        washing_detergent = total_washes * EMISSION_FACTORS["
            clothing_washing_detergent"]
        clothing += washing_electricity + washing_detergent

    # Food
    meals_veggie = float(data["mealsVeggie"])
    meals_meat = float(data["mealsMeat"])
    takeout_meals = float(data["takeoutMeals"])
    total_meals = meals_veggie + meals_meat

    food = (meals_veggie * EMISSION_FACTORS["food_production_veggie"] +
            meals_meat * EMISSION_FACTORS["food_production_meat"] +
            total_meals * EMISSION_FACTORS["food_transport"] +
            total_meals * EMISSION_FACTORS["food_cooking"] +
            total_meals * EMISSION_FACTORS["food_waste"] +
            takeout_meals * EMISSION_FACTORS["takeout_packaging"])

    # Housing
    electricity = float(data["electricityKwh"])
    dorm = float(data["dormArea"])
    computer_hours = float(data["computerHours"])
    computer_wattage = float(data["computerWattage"])
    computer_electricity = computer_hours * computer_wattage / 1000  #
        Convert to kWh

    housing_electricity = max(0, (electricity * EMISSION_FACTORS["
        housing_electricity"]) - washing_electricity)
    housing = (housing_electricity +
              dorm * EMISSION_FACTORS["housing_heating"] +
              computer_electricity * EMISSION_FACTORS["
                  housing_electricity"])

    # Transportation - Public
    bus_trips = float(data["busTrips"])
    bus_metro_avg_distance = float(data["busMetroAvgDistance"])
    bike_trips = float(data["bikeTrips"])
    bike_avg_distance = float(data["bikeAvgDistance"])

    bus_metro_km = bus_trips * bus_metro_avg_distance
    bike_km = bike_trips * bike_avg_distance

    public_transport = (bus_metro_km * EMISSION_FACTORS["
        transport_bus_metro"] +
                        bike_km * EMISSION_FACTORS["transport_bike"])

    # Transportation - Taxi
    taxi_trips = float(data["taxiTrips"])
    taxi_avg_distance = float(data["taxiAvgDistance"])
    taxi_type = data["taxiType"]
    taxi_factor = EMISSION_FACTORS[f"transport_taxi_{taxi_type}"]
```

```
    taxi_km = taxi_trips * taxi_avg_distance
    taxi_transport = taxi_km * taxi_factor

    # Transportation - Shuttle/E-bike
    shuttle_trips = float(data["shuttleTrips"])
    shuttle_km = shuttle_trips * 12  # Fixed 12 km per trip
    shuttle_transport = shuttle_km * EMISSION_FACTORS["
        transport_shuttle"]

    ebike_trips = float(data["ebikeTrips"])
    ebike_avg_distance = float(data["ebikeAvgDistance"])
    ebike_km = ebike_trips * ebike_avg_distance
    ebike_transport = ebike_km * EMISSION_FACTORS["transport_ebike"]

    # Transportation - HSR
    hsr_month = int(data["hsrMonth"])
    hsr_trips = float(data["hsrTrips"])
    hsr_avg_distance = float(data["hsrAvgDistance"])
    hsr_transport = 0

    if hsr_month > 0:  # Only calculate if month is selected
        hsr_km = hsr_trips * hsr_avg_distance
        hsr_transport = hsr_km * EMISSION_FACTORS["
            transport_hsr_electricity"]

    # Transportation - Flight
    domestic_flights = float(data["domesticFlights"])
    domestic_flight_distance = float(data["domesticFlightDistance"])
    domestic_cabin = data["domesticCabin"]
    domestic_factor = EMISSION_FACTORS[f"transport_flight_domestic_{
        domestic_cabin}"]
    domestic_flight_km = domestic_flights * domestic_flight_distance
    domestic_flight_emission = domestic_flight_km * domestic_factor

    international_flights = float(data["internationalFlights"])
    international_flight_distance = float(data["
        internationalFlightDistance"])
    international_cabin = data["internationalCabin"]
    international_factor = EMISSION_FACTORS[f"
        transport_flight_international_{international_cabin}"]
    international_flight_km = international_flights *
        international_flight_distance
    international_flight_emission = international_flight_km *
        international_factor

    air_transport = domestic_flight_emission +
        international_flight_emission

    # Total transportation
    transport = (public_transport + taxi_transport + shuttle_transport
        +
                 ebike_transport + hsr_transport + air_transport)

    # Recycling
    recycled_plastic = float(data["recycledPlastic"])
    recycled_paper = float(data["recycledPaper"])
    recycled_metal = float(data["recycledMetal"])
```

```
    recycled_cardboard = float(data["recycledCardboard"])

    recycling = (recycled_plastic * EMISSION_FACTORS["recycling_plastic
        "] +
                 recycled_paper * EMISSION_FACTORS["recycling_paper"] +
                 recycled_metal * EMISSION_FACTORS["recycling_metal"] +
                 recycled_cardboard * EMISSION_FACTORS["
                     recycling_cardboard"])

    # Total emissions
    total = clothing + food + housing + transport + recycling

    return {
        "clothing": clothing,
        "food": food,
        "housing": housing,
        "publicTransport": public_transport,
        "taxiTransport": taxi_transport,
        "shuttleTransport": shuttle_transport,
        "ebikeTransport": ebike_transport,
        "hsrTransport": hsr_transport,
        "airTransport": air_transport,
        "transport": transport,
        "recycling": recycling,
        "total": total,
        "computerElectricity": computer_electricity,
        "shuttleKm": shuttle_km,
    }

if __name__ == "__main__":
    app.run(debug=True)
```

Listing 1: Carbon calculator