

Kodların Açıklaması

→ HTML Seablonunu göstermek için

```
from flask import Flask, render_template, request
import pandas as pd
import pickle
import os
```

→ form verilerini almak için

→ dosya işlemleri için

12:52

```
app = Flask(__name__)
```

→ Python'da web arayüz için gerekli bir değişken

```
# Veri setini yükle ve modeli oluştur
data = pd.read_csv('tips_data.csv')
features = ['total_bill', 'sex', 'smoker', 'time', 'size', 'day_Sat', 'day_Sun', 'day_Thur']
target = 'tip'
```

→ Veri dosyasını pandas ile yükle

→ Modelin kullanacağı
sütunlar

```
X = data[features]
y = data[target]
```

```
from sklearn.ensemble import RandomForestRegressor
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X, y)
```

→ Random forest algoritması ile regresyon modeli oluşturuyoruz

modeli veri ile eğitiyoruz.

100 kadar
örnekle
kullanılarak
sabit seed (tohum)

```
@app.route('/', methods=['GET', 'POST'])
def index():
```

→ Flask'in

→ Her iki için route tanımlıyoruz.

Her get - her post isteklerini kabul eder

```
if request.method == 'POST':
    total_bill = float(request.form['total_bill'])
    sex = int(request.form['sex'])
    smoker = int(request.form['smoker'])
    time = int(request.form['time'])
    size = int(request.form['size'])
    day = request.form['day']
```

Formdaki bilgileri alıyor ve
uygun tutlara dönüştürüyor.

```
    day_Sat = 1 if day == 'Cumartesi' else 0
    day_Sun = 1 if day == 'Pazar' else 0
    day_Thur = 1 if day == 'Perşembe' else 0
```

→ Bütün bilgilerini "one-hot encoding"
bilgiye çeviriyoruz.

```
    tahmin = model.predict([[total_bill, sex, smoker, time, size, day_Sat, day_Sun, day_Thur]])
    tahmin = round(tahmin, 2)
    yuzde = round((tahmin / total_bill) * 100, 2)
```

→ Modelden tahmin yapıyoruz

→ yüzde kesesi

```
    return render_template('index.html', tahmin=tahmin, yuzde=yuzde, total_bill=total_bill)
```

→ Sonuçları HTML Seablonuna
gönderiyoruz.

```
    return render_template('index.html')
```

→ Sayfa ilk açıldığında form

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

gönderilmezse boş sayfa gösteriyoruz.

→ Uygulamaya besleniliyor

Hata ayıklama modu aktif

Flask Sayfesinde

α Web arayüzü oldu

α Form verilerini işleyebiliriz.

α Model tahminlerini kontrol
gösterebiliriz.

