Energy2.py

```python
import pandas as pd
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score

# Load the dataset
file_path = 'C:/Users/2273581/Downloads/Energy_dataset.csv'
df = pd.read_csv(file_path)

# Define the features (X) and target (y)
X = df[['energy_consumption_kwh', 'peak_hours_usage', 'off_peak_usage', 'renewable_energy_pct', 'household_size', 't
y = df['billing_amount']

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features
scaler = StandardScaler()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Best parameters for Decision Tree: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 10}
Best parameters for Decision Tree: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 10}
Best cross-validation score for Decision Tree: -0.075061928680156
Decision Tree - Mean Squared Error: 17702.50311737769
Decision Tree - Mean Squared Error: 17702.50311737769
Decision Tree - R-squared: -0.02891041950058204
```

Python Python Python

Energy2.py

```python
import pandas as pd
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score

# Load the dataset
file_path = 'C:/Users/2273581/Downloads/Energy_dataset.csv'
df = pd.read_csv(file_path)

# Define the features (X) and target (y)
X = df[['energy_consumption_kwh', 'peak_hours_usage', 'off_peak_usage', 'renewable_energy_pct', 'household_size', 't
y = df['billing_amount']

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features
scaler = StandardScaler()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
[CV] END max_depth=30, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=  27.4s
[CV] END max_depth=30, min_samples_leaf=4, min_samples_split=10, n_estimators=200; total time=  27.3s
Best parameters for Random Forest: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 10, 'n_estimators': 200}
Best cross-validation score for Random Forest: -0.006132373396719126
Random Forest - Mean Squared Error: 17343.91378899142
Random Forest - R-squared: -0.008068378471019155
Fitting 3 folds for each of 36 candidates, totalling 108 fits
```

Python Python Python