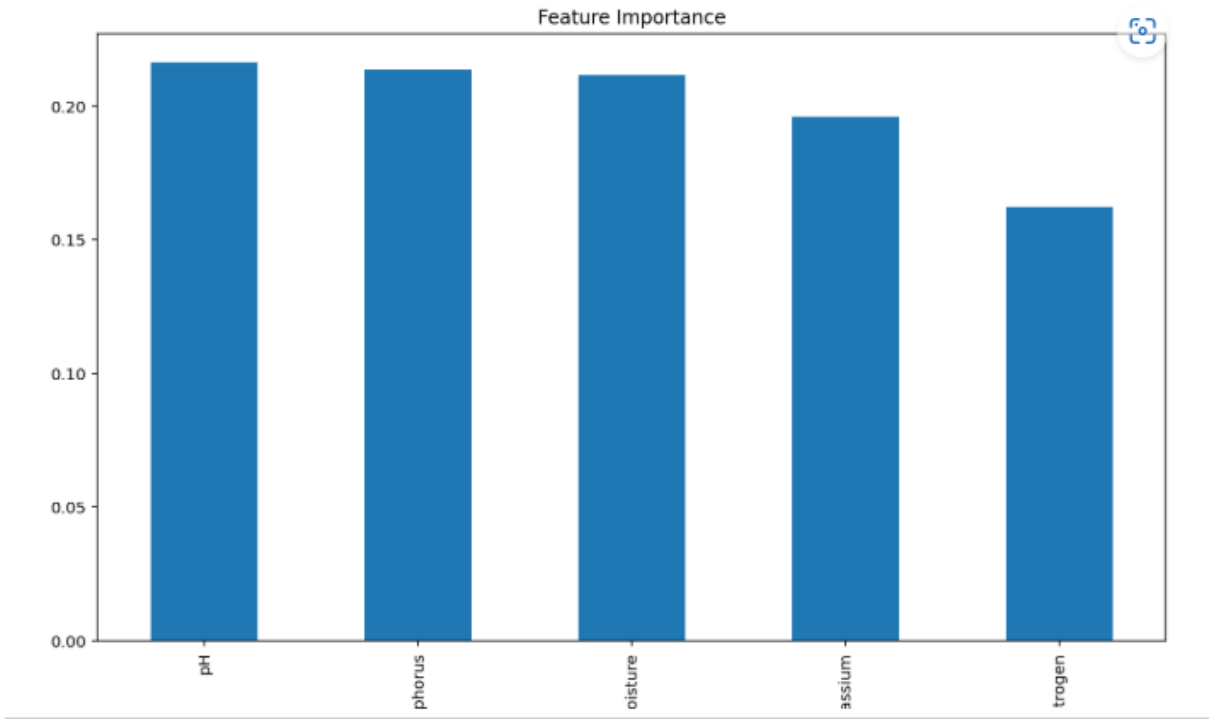
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Load or simulate soil sensor data
data = {
    'Moisture': np.random.uniform(10, 50, 100),
    'pH': np.random.uniform(4.5, 8, 100),
    'Nitrogen': np.random.uniform(10, 100, 100),
    'Phosphorus': np.random.uniform(5, 50, 100),
    'Potassium': np.random.uniform(5, 50, 100),
    'SoilQuality': np.random.choice(['Poor', 'Moderate', 'Good'], 100)
}
df = pd.DataFrame(data)

# Encode labels
df['SoilQuality'] = df['SoilQuality'].astype('category').cat.codes

# Split data
X = df.drop('SoilQuality', axis=1)
y = df['SoilQuality']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train ML model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

```
# Predictions
y_pred = model.predict(X_test)

# Evaluation
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy:.2f}')
print(classification_report(y_test, y_pred))

# Feature Importance
feature_importances = pd.Series(model.feature_importances_, index=X.columns)
feature_importances.sort_values(ascending=False).plot(kind='bar')
plt.title('Feature Importance')
plt.show()
```

```
Accuracy: 0.50
              precision    recall  f1-score   support

           0       0.33      0.40      0.36         5
           1       0.60      0.60      0.60         5
           2       0.56      0.50      0.53        10

    accuracy                           0.50        20
   macro avg       0.50      0.50      0.50        20
weighted avg       0.51      0.50      0.50        20
```


Feature Importance
0.20
0.15
0.10
0.05
0.00
pH
phorus
oisture
assium
trogen