```python
from sklearn.tree import DecisionTreeClassifier, plot_tree, export_text
from sklearn.preprocessing import OneHotEncoder
import matplotlib.pyplot as plt
X = [
    ['Sunny', 'Hot', 'High', 'Weak'],
    ['Sunny', 'Hot', 'High', 'Strong'],
    ['Overcast', 'Hot', 'High', 'Weak'],
    ['Rain', 'Mild', 'High', 'Weak'],
    ['Rain', 'Cool', 'Normal', 'Weak'],
    ['Rain', 'Cool', 'Normal', 'Strong'],
    ['Overcast', 'Cool', 'Normal', 'Strong'],
    ['Sunny', 'Mild', 'High', 'Weak'],
    ['Sunny', 'Cool', 'Normal', 'Weak'],
    ['Rain', 'Mild', 'Normal', 'Weak'],
    ['Sunny', 'Mild', 'Normal', 'Strong'],
    ['Overcast', 'Mild', 'High', 'Strong'],
    ['Overcast', 'Hot', 'Normal', 'Weak'],
    ['Rain', 'Mild', 'High', 'Strong']
]
y = ['No', 'No', 'Yes', 'Yes', 'Yes', 'No', 'Yes', 'No', 'Yes', 'Yes', 'Yes', 'Yes', 'Yes', 'No']
```

```
encoder = OneHotEncoder()
X_encoded = encoder.fit_transform(X).toarray()
clf = DecisionTreeClassifier(criterion='entropy', max_depth=3)
clf.fit(X_encoded, y)
tree_rules = export_text(clf, feature_names=encoder.get_feature_names_out())
print("Decision Tree Rules:\n")
print(tree_rules)
plt.figure(figsize=(12,8))
plot_tree(
    clf,
    feature_names=encoder.get_feature_names_out(),
    class_names=clf.classes_,
    filled=True,
    rounded=True,
    fontsize=10
)
plt.title("Decision Tree - Play Tennis")
plt.show()
example = [['Sunny', 'Cool', 'High', 'Strong']]
example_encoded = encoder.transform(example).toarray()
prediction = clf.predict(example_encoded)
print(f"\nPrediction for {example[0]}: {prediction[0]}")
```

```
Decision Tree Rules:

|--- x0_Overcast <= 0.50
|   |--- x2_High <= 0.50
|   |   |--- x3_Strong <= 0.50
|   |   |   |--- class: Yes
|   |   |--- x3_Strong >  0.50
|   |   |   |--- class: No
|   |--- x2_High >  0.50
|   |   |--- x0_Sunny <= 0.50
|   |   |   |--- class: No
|   |   |--- x0_Sunny >  0.50
|   |   |   |--- class: No
|--- x0_Overcast >  0.50
|   |--- class: Yes
```

Decision Tree - Play Tennis