```
In [1]: ▶ import numpy as np
            import pandas as pd
            import matplotlib.pyplot as plt
            from sklearn.datasets import load_iris
In [2]:

    df = load iris()

In [3]:
         ⋈ df
   Out[3]: {'data': array([[5.1, 3.5, 1.4, 0.2],
                     [4.9, 3., 1.4, 0.2],
                     [4.7, 3.2, 1.3, 0.2],
                     [4.6, 3.1, 1.5, 0.2],
                     [5., 3.6, 1.4, 0.2],
                     [5.4, 3.9, 1.7, 0.4],
                     [4.6, 3.4, 1.4, 0.3],
                     [5., 3.4, 1.5, 0.2],
                     [4.4, 2.9, 1.4, 0.2],
                     [4.9, 3.1, 1.5, 0.1],
                     [5.4, 3.7, 1.5, 0.2],
                     [4.8, 3.4, 1.6, 0.2],
                     [4.8, 3., 1.4, 0.1],
                     [4.3, 3., 1.1, 0.1],
                     [5.8, 4., 1.2, 0.2],
                     [5.7, 4.4, 1.5, 0.4],
                     [5.4, 3.9, 1.3, 0.4],
                     [5.1, 3.5, 1.4, 0.3],
                     [5.7, 3.8, 1.7, 0.3],
                          20 1 5 2 2
In [4]: \mathbf{M} \times \mathbf{X} = \mathbf{df.data}
            y = df.target
         ▶ from sklearn.model_selection import train_test_split
In [5]:
In [6]:
         | X_train,X_test,y_train,y_test = train_test_split(X,y,test_size = .2 , random_state = 42
        Classification
In [7]:
         ▶ from sklearn.tree import DecisionTreeClassifier
            from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
         M model = DecisionTreeClassifier(max_depth=2, min_samples_split=3, min_samples_leaf=2)
In [8]:
            model.fit(X train,y train)
            y_pred = model.predict(X_test)
In [9]:
         ▶ | from sklearn.metrics import accuracy score
            accuracy_score(y_test,y_pred)
   Out[9]: 0.9666666666666667
```

```
In [10]: ▶ import matplotlib.pyplot as plt
         from sklearn import tree
         plt.figure()
         tree.plot_tree(model,filled = True)
  Out[10]: [Text(0.4, 0.83333333333333334, 'x[2] <= 2.45\ngini = 0.667\nsamples = 120\nvalue = [4
         0, 41, 39]'),
          Text(0.2, 0.5, 'gini = 0.0\nsamples = 40\nvalue = [40, 0, 0]'),
          Text(0.6, 0.5, 'x[2] \le 4.75 = 0.5 = 80 = 80 = [0, 41, 39]'),
          x[2] \le 2.45
                       gini = 0.667
                      samples = 120
                    value = [40, 41, 39]
                                x[2] <= 4.75
               gini = 0.0
                                  gini = 0.5
             samples = 40
                                samples = 80
            value = [40, 0, 0]
                              value = [0, 41, 39]
                       gini = 0.053
                                          gini = 0.206
                       samples = 37
                                         samples = 43
                     value = [0, 36, 1]
                                        value = [0, 5, 38]
      Regression
Model1 = DecisionTreeRegressor(ccp alpha=0.01,max depth=2, min samples split=3, min sam
         model1.fit(X_train,y_train)
```

Out[15]: 0.9553798693111485

