Step 1: Project Structure:

water_quality_project/

```
water_quality_project/
│
├── main.py
├── data_preprocessing.py
├── model_training.py
├── model_evaluation.py
├── prediction.py
└── water_potability.csv
```

Step 2: Created main.py

Code:

```python
from data_preprocessing import load_and_preprocess_data
from model_training import train_model
from model_evaluation import evaluate_model
from prediction import predict_water_quality


if __name__ == "__main__":
    # Load and preprocess data
    X_train, X_test, y_train, y_test = load_and_preprocess_data('water_potability.csv')

    # Train the model
    model = train_model(X_train, y_train)

    # Evaluate the model
    evaluate_model(model, X_test, y_test)

    # Make prediction (example)
    sample_data = [7.0, 204.9, 20791.32, 368.52, 564.3, 10.38, 4.5, 396.4, 10.0]
    result = predict_water_quality(model, sample_data)
    print("\nPrediction for sample input:", result)
```

Step 3: Created data_preprocessing.py file

Code:

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.impute import SimpleImputer


def load_and_preprocess_data(filepath):

    data = pd.read_csv(filepath)


    # Handling missing values

    imputer = SimpleImputer(strategy='mean')

    data_imputed = pd.DataFrame(imputer.fit_transform(data), columns=data.columns)


    # Splitting features and target

    X = data_imputed.drop('Potability', axis=1)

    y = data_imputed['Potability']


    # Train-test split

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


    return X_train, X_test, y_train, y_test
```

Step 4: Created model_training.py file

Code:

```
from sklearn.ensemble import RandomForestClassifier


def train_model(X_train, y_train):

    model = RandomForestClassifier(n_estimators=100, random_state=42)

    model.fit(X_train, y_train)

    return model
```

Step 5: Created model_evaluation.py file

Code:

```python
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix


def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    acc = accuracy_score(y_test, y_pred)
    print("Accuracy:", acc)
    print("\nClassification Report:\n", classification_report(y_test, y_pred))
    print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

Step 6: Created prediction.py file

Code:

```python
import numpy as np


def predict_water_quality(model, input_data):
    prediction = model.predict([input_data])
    return "Safe to Drink" if prediction[0] == 1 else "Not Safe to Drink"
```

Step 7: paste water_potability.csv file in same folder

Output:

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS                      Python Debug Console  + ∨  ⊓ 🗑 ⋯ ∨ ×

PS C:\Users\Admin\OneDrive\Desktop\Skills and Self Development with AI\AI and ML\Artificial Intelligence & Machine Learning\Projects\water_quality_
project>  & 'c:\Users\Admin\AppData\Local\Programs\Python\Python313\python.exe' 'c:\Users\Admin\.vscode\extensions\ms-python.debugpy-2025.6.0-win32
-x64\bundled\libs\debugpy\launcher' '64093' '--' 'C:\Users\Admin\OneDrive\Desktop\Skills and Self Development with AI\AI and ML\Artificial Intellig
ence & Machine Learning\Projects\water_quality_project\main.py'
Accuracy: 0.6783536585365854

Classification Report:
              precision    recall  f1-score   support

         0.0       0.70      0.86      0.77       412
         1.0       0.61      0.38      0.47       244

    accuracy                           0.68       656
   macro avg       0.65      0.62      0.62       656
weighted avg       0.67      0.68      0.66       656


Confusion Matrix:
 [[353  59]
 [152  92]]
c:\Users\Admin\AppData\Local\Programs\Python\Python313\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but
  RandomForestClassifier was fitted with feature names
  warnings.warn(

Prediction for sample input: Not Safe to Drink
PS C:\Users\Admin\OneDrive\Desktop\Skills and Self Development with AI\AI and ML\Artificial Intelligence & Machine Learning\Projects\water_quality_
project>
```