

```

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
import pandas as pd
from sklearn.preprocessing import LabelEncoder

# Load dataset
file_path = (r"C:\Users\nimes\OneDrive\Documents\datasets\Play_tennis - Play_tennis.csv") # Change to your actual file path
data = pd.read_csv(file_path)

# Handle missing or incorrect values
data = data.replace("hingh", "high") # Fixing the typo in 'humidity'

# Initialize LabelEncoders
label_encoders = {}
for column in data.columns:
    if data[column].dtype == 'object': # Encode only categorical columns
        label_encoders[column] = LabelEncoder()
        data[column] = label_encoders[column].fit_transform(data[column])

# Convert to float
data = data.astype(float)

print(data.head())

# Save the transformed dataset
data.to_csv("encoded_dataset.csv", index=False)

# Load dataset
file_path = (r"C:\Users\nimes\PycharmProjects\college\encoded_dataset.csv") # Change this to your actual dataset path
data = pd.read_csv(file_path)

# Assuming the last column is the target variable
y = data.iloc[:, -1]
X = data.iloc[:, :-1]

# Split data 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)

# Create a Gaussian Naïve Bayes classifier
classifier = GaussianNB()

# Train the model
classifier.fit(X_train, y_train)

```

```
# Predict on test data
y_pred = classifier.predict(X_test)

# Compute accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of Naïve Bayes classifier: {accuracy * 100:.2f}%")
```

	outlook	temperature	humidity	windy	play
0	0.0	1.0	0.0	0.0	1.0
1	0.0	0.0	1.0	1.0	1.0
2	0.0	2.0	0.0	1.0	1.0
3	0.0	1.0	1.0	0.0	1.0
4	1.0	2.0	0.0	0.0	1.0

Accuracy of Naïve Bayes classifier: 33.33%