```
import pandas as pd
import numpy as np
import random


#dane - informacje o pogodzie
outlook=['Sunny','Sunny','Overcast','Rainy','Rainy','Rainy','Overcast','Sunny','Sunny','Rainy','Sunny','Overcast','Overcast','Rainy']
temp=['Hot','Hot','Hot','Mild','Cool','Cool','Cool','Mild','Cool','Mild','Mild','Mild','Hot','Mild']
humid=['High','High','High','High', 'Normal','Normal','Normal','High', 'Normal','Normal','Normal', 'High', 'Normal', 'High']
wind=['Weak', 'Strong', 'Weak','Weak','Weak','Strong', 'Strong','Weak','Weak','Weak','Strong', 'Strong','Weak', 'Strong']

#etykiety - czy pogoda jest odpowiednia na grę w tenisa?
play=['No','No','Yes','Yes','Yes','No','Yes','No','Yes','Yes','Yes','Yes','Yes','No']
```

Konwersja **danych kategorycznych** do **danych numerycznych**:

```
from sklearn import preprocessing
le = preprocessing.LabelEncoder()

outlook_encoded=le.fit_transform(outlook)
temp_encoded=le.fit_transform(temp)
humid_encoded=le.fit_transform(humid)
wind_encoded=le.fit_transform(wind)

print(outlook_encoded)
print(temp_encoded)
print(humid_encoded)
print(wind_encoded)
```

```
[2 2 0 1 1 1 0 2 2 1 2 0 0 1]
[1 1 1 2 0 0 0 2 0 2 2 2 1 2]
[0 0 0 0 1 1 1 0 1 1 1 0 1 0]
[1 0 1 1 1 0 0 1 1 1 0 0 1 0]
```

Konersja etykiet do **danych numerycznych**:

```
label=le.fit_transform(play)
print(label)
```

```
[0 0 1 1 1 0 1 0 1 1 1 1 1 0]
```

Outlook + temp

```
data=list(zip(outlook_encoded,temp_encoded))
print(data)
```

```
[(2, 1), (2, 1), (0, 1), (1, 2), (1, 0), (1, 0), (0, 0), (2, 2), (2, 0), (1, 2), (2, 2), (0, 2), (0, 1), (1, 2)]
```

```
data=list(zip(outlook_encoded,temp_encoded,humid_encoded,wind_encoded))
data
```

```
[(2, 1, 0, 1),
 (2, 1, 0, 0),
 (0, 1, 0, 1),
 (1, 2, 0, 1),
 (1, 0, 1, 1),
 (1, 0, 1, 0),
 (0, 0, 1, 0),
 (2, 2, 0, 1),
 (2, 0, 1, 1),
 (1, 2, 1, 1),
 (2, 2, 1, 0),
 (0, 2, 0, 0),
 (0, 1, 1, 1),
 (1, 2, 0, 0)]
```

```
from sklearn.neighbors import KNeighborsClassifier

neigh = KNeighborsClassifier(n_neighbors=6)
neigh.fit(data, label)

# case: Sunny, Cool, High, Strong
case = np.array([[2, 0, 0, 0]])

# "0" is No for Play Tennis
print(neigh.predict(case))
```

```
[0]
```