5-dars uchun Report:

Demak birlamchi holatda encoding haqida so'z yuritildi, bunda Handling missing values (mean, mode, median, fixed va drop), undan so'ng esa encoding haqida so'z bordi. Unga tarif berildi quyidagicha; Birorta so'z ko'rinishida berilgan datasetimiz ya'ni umumiy yana ham aniqroq aytsak categorical (string, object) ni numberical (int, float) holatga o'tkazish jarayoni encoding hisoblanadi.

Uning turlari: One-hot encoding, Label encoding, Frequency encoding, Target encoding va Ordinal encodingdir.

1) One-hot encodingda categoricaldan numericalga aylantirishda 0 va 1 dan iborat bo'ladi va bitta ustundagi qiymatlarni har biriga alohida ustun yaratib unga 0 va 1 dan iborat qiymatlar berib chiqadi.
Bu yerda classlarni tushunib olish ham muhimdri. Ya'ni dublicated bo'lmagan qiymatlardan bittadan vakil oladi. Classlar sonini topish uchun bizga

```
df.nunique()
```

qaysi ustunda qaysi classlar borligini va sonini aks ettiradi.
So'ngra manashu kodlardan fodalangan holda bir ustunni (object, string) ni numberical(int yoki float) ko'rinishiga o'tkazish jarayonidagi kodlar jamlanmasidir:

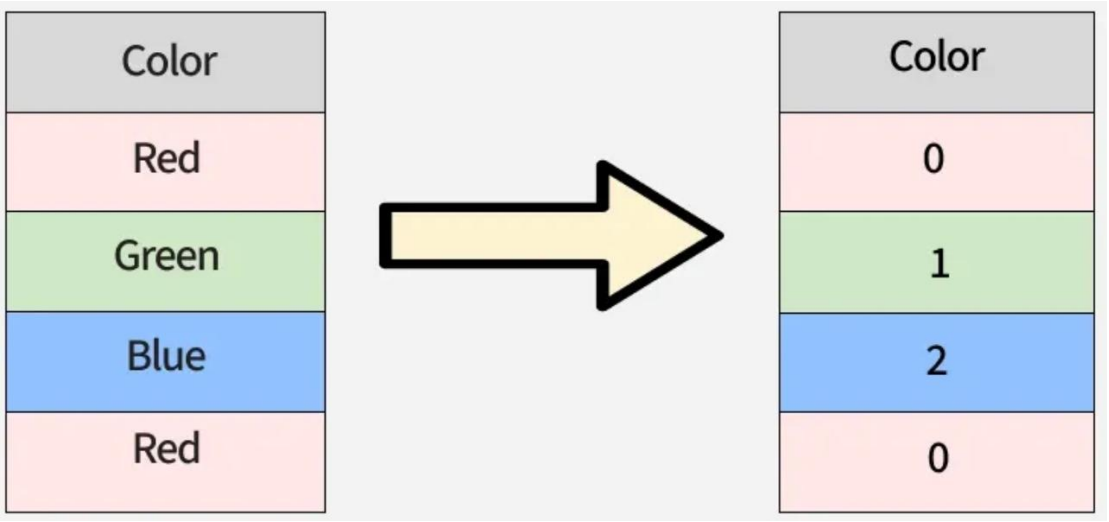
```
dummies = pd.get_dummies(df['Patient Gender'],
prefix='tasnifli ustun', dtype=int)

dummies

df = pd.concat([df.drop(columns=['Patient
Gender']),dummies], axis=1)

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9216 entries, 0 to 9215
Data columns (total 12 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Patient Id                  9216 non-null   object
 1   Patient Admission Date      9216 non-null   object
 2   Patient Admission Time      9216 non-null   object
 3   Merged                      9216 non-null   object
 4   Patient Age                 9216 non-null   int64
 5   Patient Race                9216 non-null   object
 6   Department Referral         3816 non-null   object
 7   Patient Admission Flag      9216 non-null   object
 8   Patient Waittime            9216 non-null   int64
 9   tasnifli ustun_Female       9216 non-null   int64
```

```
 10  tasnifli ustun_Femaleemale  9216 non-null   int64
 11  tasnifli ustun_Male         9216 non-null   int64
dtypes: int64(5), object(7)
memory usage: 864.1+ KB
```

shunday ko’rinishda bo’ladi.

O’ng tarafdagi datasetimizga **dummies** datasetchamizga qo’shimimiz kerakligi sababli bizga **concat()** funksiyasidan foydalanamiz. Va original ustunni tashlab yuborib one-hot encoding qilingan ustunlarni qo’shish uchun **df.drop** dan foydalanamiz. Bu yerda **axis=1 yangi dummies** ustunlar yoniga qo’shadi. Va e’tibor berishimiz bo’lgan jabha bor, har doim one-code encoding ishlatishdan oldin **df.nunique()** qilib biroz mushohada qilishimiz kerak. Sababi bazida one-hot encoding qlishda datasetimiz collapse holatida bo'lishi mumkin unga sabab ustunlar soni qatorlar sonidan ortib ketkanidir.

2) Label Encoing (tamg’a yoki belgilik) - Categorical qiymatlarni alifbo tartibidagi ketma-ketlikda 0 dan boshlab z gacha nechta qiymat bo’lsa o’shangacha almashtiradi.
Masalan:



uni chaqirishda esa biz sklearn kutubxonasidagi preprocessing moduleni chqariib undan LabelEncode classini import qilayapmiz.

```
from sklearn.preprocessing import LabelEncoder
```

Bu yerda encoder degan o’zgaruvchi va uni chaqrib olayapmiz.

```
encoder = LabelEncoder()
encoder
```

So’ngida esa undan encoding qilishda quyidagi kod orqali foydalanmiz:

```
df['Patient Race'] = encoder.fit_transform(df['Patient Race'])
```

bu yerda biz tanlagan ustun va uni encoder orqali fit_transform bilan bir holatdan boshqa holatda transfor qilib olamiz. Va shu bilan birga keyincahlik biz bir chegara yoki shablon ya’ni datasetmizdan kelib chiqib o’zimiz proporsiya qilib olishmiz uchun bizga **Threshold** kerak bo’ladi. Bu bizga chegarani belgilashda kerak bo’ladi asosan one-hot encodingda.