```
########################################
# Abdurrahman Bulut
########################################
# Task 1
import seaborn as sns

df = sns.load_dataset("titanic")
########################################
########################################
# Task 2
females = df['sex'].value_counts()['female']
males = df['sex'].value_counts()['male']
########################################
########################################
# Task 3
df.nunique()
########################################
########################################
# Task 4
df['pclass'].nunique()  # unique deÄŸer sayÄ±sÄ±
df['pclass'].unique()  # unique deÄŸerler
########################################
########################################
# Task 5
df[['pclass', 'parch']].nunique()
########################################
########################################
# Task 6
# baÅŸta tipi object, sonradan category yapÄ±yoruz bu satÄ±rla
df['embarked'] = df['embarked'].astype('category')
########################################
########################################
# Task 7
df[df['embarked'] == 'C'].reset_index()
########################################
########################################
# Task 8
df[df['embarked'] != 'S'].reset_index()
########################################
########################################
# Task 9
df[(df['age'] < 30) & (df['sex'] == 'female')].reset_index()
########################################
########################################
# Task 10
var = df[(df['fare'] > 500) | (df['age'] > 70)].reset_index()
########################################
########################################
# Task 11
df.isnull().sum()
########################################
########################################
# Task 12
df.drop(columns=['who'], inplace=True)
########################################
########################################
# Task 13
mode = df['deck'].mode()[0]
df['deck'].fillna(mode, inplace=True)
########################################
########################################
# Task 14
median = df['age'].median()
df['age'].fillna(median, inplace=True)
########################################
########################################
# Task 15
df.groupby(['pclass', 'sex'])['survived'].agg(['sum', 'count', 'mean'])
########################################
```

```
######################################
# Task 16
df['age_flag'] = df['age'].apply(lambda x: 1 if x < 30 else 0)
######################################
######################################
# Task 17
df = sns.load_dataset("tips")
######################################
######################################
# Task 18
df.groupby('time')['total_bill'].agg(['sum', 'min', 'max', 'mean'])
######################################
######################################
# Task 19
df.groupby(['day', 'time'])['total_bill'].agg(['sum', 'min', 'max', 'mean'])
######################################
######################################
# Task 20
df[(df['time'] == 'Lunch') & (df['sex'] == 'Female')].groupby('day')[['total_bill', 'tip']].agg(
    ['sum', 'min', 'max', 'mean'])
######################################
######################################
# Task 21
df.loc[(df['size'] < 3) & (df['total_bill'] > 10), 'total_bill'].mean()
######################################
######################################
# Task 22
df['total_bill_tip_sum'] = df['total_bill'] + df['tip']
######################################
######################################
# Task 23
df.sort_values(by='total_bill_tip_sum', ascending=False).head(30)
######################################
```