In [ ]:
```
import pandas as pd
import numpy as np
```

In [ ]:
```
# Creating from a csv file
mcEnrollmentDf = pd.read_csv ('Montgomery_College_Enrollment_Data.csv')
```

In [ ]:
```
# View data
mcEnrollmentDf.head ()
```

In [ ]:
```
# View data
mcEnrollmentDf.tail ()
```

In [ ]:
```
# View data
mcEnrollmentDf
```

In [ ]:
```
# <TAB> completion
mcEnrollmentDf.
```

In [ ]:
```
# Access column labels
mcEnrollmentDf.columns
```

In [ ]:
```
# Count of values in each column
mcEnrollmentDf.count ()
```

In [ ]:
```
# Statistical information on numeric columns
mcEnrollmentDf.describe ()
```

In [ ]:
```
# Data types
mcEnrollmentDf.dtypes
```

In [ ]:
```
# Size
mcEnrollmentDf.shape
```

In [ ]:
```
# Counts of values in one column
mcEnrollmentDf ['Age Group'].value_counts ()
```

In [ ]:
```
# Selection
mcEnrollmentDf ['Age Group']
```

In [ ]:
```
# Add a column
mcEnrollmentDf ['ZIPDuplicated'] = mcEnrollmentDf ['ZIP']
```

In [ ]:
```
mcEnrollmentDf.head ()
```

In [ ]:
```
# Compare columns
mcEnrollmentDf ['ZIPDuplicated'] == mcEnrollmentDf ['ZIP']
```

In [ ]:
```
# Add 1 to each value in a column
mcEnrollmentDf ['ZIPDuplicated'] += 1
```

In [ ]:
```
# Compare columns
mcEnrollmentDf ['ZIPDuplicated'] == mcEnrollmentDf ['ZIP']
```

In [ ]:
```
mcEnrollmentDf.columns
```

In [ ]:
```
# Delete a column
del mcEnrollmentDf ['ZIPDuplicated']
```

In [ ]:
```
mcEnrollmentDf.columns
```

In [ ]:
```
# Transpose
mcEnrollmentDf.T
```

In [ ]:
```
# Transpose
mcEnrollmentDf
```

In [ ]:
```
# Sort
mcEnrollmentDf.sort_values (by = 'Student Type')
```

In [ ]:
```
# Count various types of students
mcEnrollmentDf ['Student Type'].value_counts ()
```

In [ ]:
```
# Example of logical comparison
mcEnrollmentDf [mcEnrollmentDf ['Student Type'] == 'Continuing']
```

In [ ]:
```
# Add a column and set value
mcEnrollmentDf ['Working'] = 'X'
mcEnrollmentDf ['ZIPDuplicated'] = mcEnrollmentDf ['ZIP']
```

In [ ]:
```
mcEnrollmentDf
```

In [ ]:
```
# Dropping any rows with missing values
mcEnrollmentDf.dropna (how = 'any')
```

In [ ]:
```
# Grouping
tDf = mcEnrollmentDf.groupby (['Student Type'])
tDf.get_group ('Continuing')
```

In [ ]:
```
mcEnrollmentDf.head ()
```

In [ ]:
```
# Convert a column to String type
mcEnrollmentDf ['ZIPDuplicated'] = mcEnrollmentDf ['ZIPDuplicated'].fillna (0.0).as
type (int)
mcEnrollmentDf ['ZIPDuplicated'] = mcEnrollmentDf ['ZIPDuplicated'].astype (str)
```

In [ ]:
```
mcEnrollmentDf.dtypes
```

In [ ]:
```
# Find counts of various student types based on age groups (like a pivot table)
mcEnrollmentDf.groupby (['Age Group', 'Student Type']).size ()
```

In [ ]:
```
# Find counts of vstudents attneind Germantwon campus
mcEnrollmentDf ['Attending Germantown'].value_counts ()
```

In [ ]:
```
# Find count of students attneind Rockville campus
mcEnrollmentDf ['Attending Rockville'].value_counts ()
```

In [ ]:
```
# Find count of students attneind Takoma Park campus
mcEnrollmentDf ['Attending Takoma Park/SS'].value_counts ()
```

In [ ]:
```
# Find counts of various student statuses attending Germantown campus
mcEnrollmentDf.groupby (['Attending Germantown', 'Student Status']).size ()
```

In [ ]:
```
# Find counts of various student statuses attending Rockville campus
mcEnrollmentDf.groupby (['Attending Rockville', 'Student Status']).size ()
```

In [ ]:
```
# Find counts of various student statuses attending Takoma Park campus
mcEnrollmentDf.groupby (['Attending Takoma Park/SS', 'Student Status']).size ()
```

In [ ]:
```
# Add a new column converting Yes/No to 1/0 for Germantown campus
mcEnrollmentDf ['AGN'] = np.where (mcEnrollmentDf ['Attending Germantown'] == 'Yes
', 1, 0)
```

In [ ]:
```
mcEnrollmentDf.head ()
```

In [ ]:
```
# Add a new column which counts how many campuses a student is attending
mcEnrollmentDf ['No. of Campuses'] = 0
mcEnrollmentDf ['No. of Campuses'] = np.where (mcEnrollmentDf ['Attending Germantow
n'] == 'Yes', mcEnrollmentDf ['No. of Campuses'] + 1, mcEnrollmentDf ['No. of Campu
ses'])
mcEnrollmentDf ['No. of Campuses'] = np.where (mcEnrollmentDf ['Attending Rockville
'] == 'Yes', mcEnrollmentDf ['No. of Campuses'] + 1, mcEnrollmentDf ['No. of Campus
es'])
mcEnrollmentDf ['No. of Campuses'] = np.where (mcEnrollmentDf ['Attending Takoma Pa
rk/SS'] == 'Yes', mcEnrollmentDf ['No. of Campuses'] + 1, mcEnrollmentDf ['No. of C
ampuses'])
```

In [ ]:
```
mcEnrollmentDf.head ()
```

In [ ]:
```
mcEnrollmentDf
```

In [ ]:
```
mcEnrollmentDf ['No. of Campuses'].value_counts ()
```

In [ ]:
```
mcEnrollmentDf.to_csv ('Mont College Enrollment - Working.csv')
```

In [ ]:
```
```