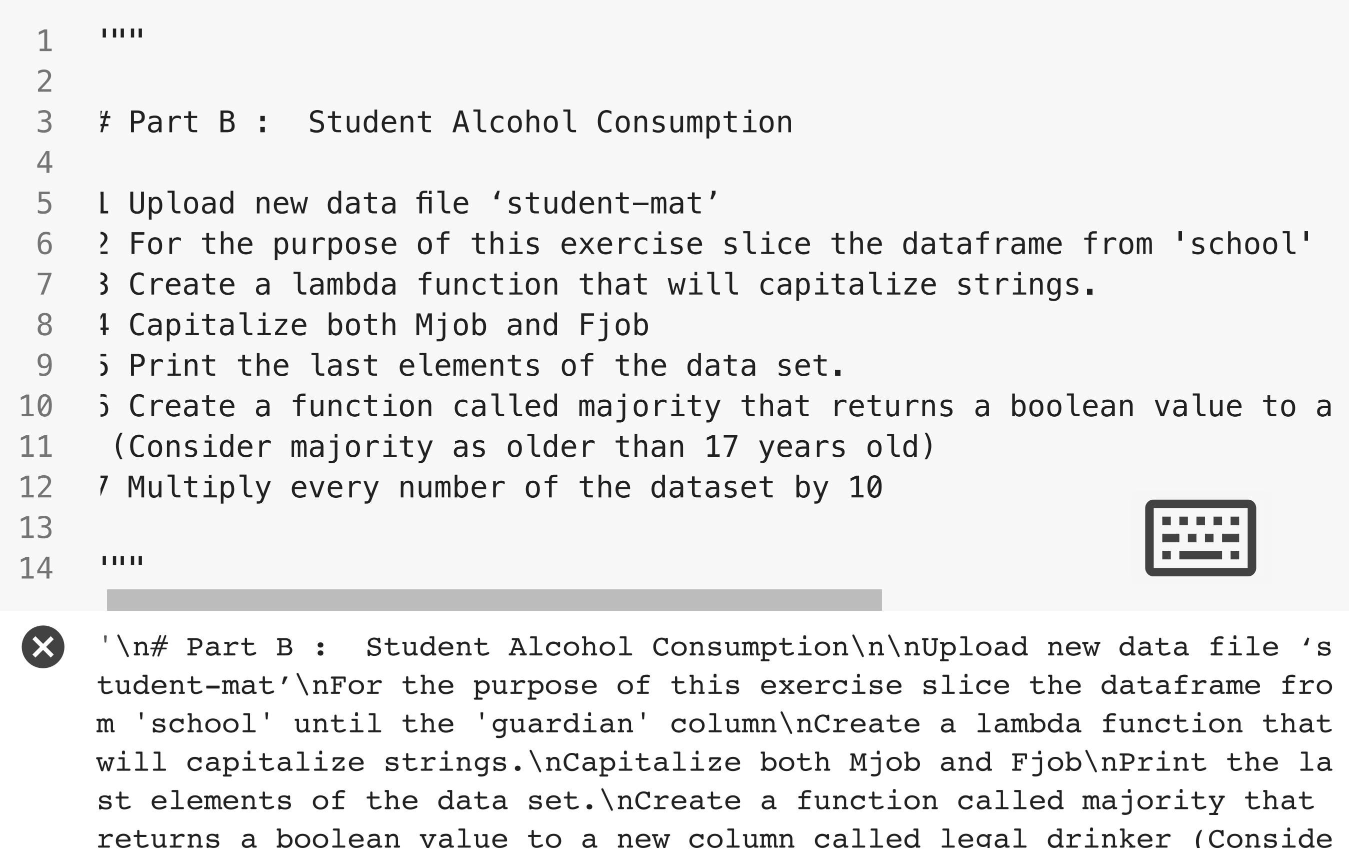

```
"""

# Part B :  Student Alcohol Consumption

1 Upload new data file ‘student-mat’
2 For the purpose of this exercise slice the dataframe from 'school'
3 Create a lambda function that will capitalize strings.
4 Capitalize both Mjob and Fjob
5 Print the last elements of the data set.
6 Create a function called majority that returns a boolean value to a
  (Consider majority as older than 17 years old)
7 Multiply every number of the dataset by 10

"""
```

```
'\n# Part B :  Student Alcohol Consumption\n\nUpload new data file ‘student-mat’\nFor the purpose of this exercise slice the dataframe from 'school' until the 'guardian' column\nCreate a lambda function that will capitalize strings.\nCapitalize both Mjob and Fjob\nPrint the last elements of the data set.\nCreate a function called majority that returns a boolean value to a new column called legal drinker (Conside
```

1 Upload new data file ‘student-mat’

```
import numpy as np
import pandas as pd
sal = pd.read_csv("student-mat.csv")
sal
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	18	U	GT3	A	4	4	at_home	te
1	GP	F	17	U	GT3	T	1	1	at_home	
2	GP	F	15	U	LE3	T	1	1	at_home	
3	GP	F	15	U	GT3	T	4	2	health	se
4	GP	F	16	U	GT3	T	3	3	other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	20	U	LE3	A	2	2	services	se
391	MS	M	17	U	LE3	T	3	1	services	se
392	MS	M	21	R	GT3	T	1	1	other	
393	MS	M	18	R	LE3	T	3	2	services	
394	MS	M	19	U	LE3	T	1	1	other	at_

395 rows × 33 columns

```
sal.columns
```

```
Index(['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu'
       'Mjob', 'Fjob', 'reason', 'guardian', 'traveltime', 'studytime'
       'failures', 'schoolsup', 'famsup', 'paid', 'activities', 'nurse
       'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout'
       'Walc', 'health', 'absences', 'G1', 'G2', 'G3'],
      dtype='object')
```

2 For the purpose of this exercise slice the dataframe from 'school' until the 'guardian' column

```
sal = sal.loc[:, 'school':'guardian']
sal
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	18	U	GT3	A	4	4	at_home	te
1	GP	F	17	U	GT3	T	1	1	at_home	
2	GP	F	15	U	LE3	T	1	1	at_home	
3	GP	F	15	U	GT3	T	4	2	health	se
4	GP	F	16	U	GT3	T	3	3	other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	20	U	LE3	A	2	2	services	se
391	MS	M	17	U	LE3	T	3	1	services	se
392	MS	M	21	R	GT3	T	1	1	other	
393	MS	M	18	R	LE3	T	3	2	services	
394	MS	M	19	U	LE3	T	1	1	other	at_

395 rows × 12 columns

3 Create a lambda function that will capitalize strings.

```
#A capitalized string is a string with the first character in each wor
# to its corresponding uppercase value,
# and all remaining characters set to their corresponding lowercase va
```

```
capitalizer = lambda x: x.capitalize()
capitalizer
```

```
<function __main__.<lambda>>
```

4 Capitalize both Mjob and Fjob

```
sal['Mjob']=sal['Mjob'].apply(capitalizer)
sal
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	18	U	GT3	A	4	4	At_home	te
1	GP	F	17	U	GT3	T	1	1	At_home	
2	GP	F	15	U	LE3	T	1	1	At_home	
3	GP	F	15	U	GT3	T	4	2	Health	se
4	GP	F	16	U	GT3	T	3	3	Other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	20	U	LE3	A	2	2	Services	se
391	MS	M	17	U	LE3	T	3	1	Services	se
392	MS	M	21	R	GT3	T	1	1	Other	
393	MS	M	18	R	LE3	T	3	2	Services	
394	MS	M	19	U	LE3	T	1	1	Other	at_

395 rows × 12 columns

```
sal['Fjob']=sal['Fjob'].apply(capitalizer)
sal
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	18	U	GT3	A	4	4	At_home	Te
1	GP	F	17	U	GT3	T	1	1	At_home	
2	GP	F	15	U	LE3	T	1	1	At_home	
3	GP	F	15	U	GT3	T	4	2	Health	Se
4	GP	F	16	U	GT3	T	3	3	Other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	20	U	LE3	A	2	2	Services	Se
391	MS	M	17	U	LE3	T	3	1	Services	Se
392	MS	M	21	R	GT3	T	1	1	Other	
393	MS	M	18	R	LE3	T	3	2	Services	
394	MS	M	19	U	LE3	T	1	1	Other	At_

395 rows × 12 columns

5 Print the last elements of the data set.

```
sal.tail()
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
390	MS	M	20	U	LE3	A	2	2	Services	Se
391	MS	M	17	U	LE3	T	3	1	Services	Se
392	MS	M	21	R	GT3	T	1	1	Other	
393	MS	M	18	R	LE3	T	3	2	Services	
394	MS	M	19	U	LE3	T	1	1	Other	At_

6 Create a function called majority that returns a boolean value to a new column called legal_drinker (Consider majority as older than 17 years old)

```python
def majority(a) :
    if a>17 :
        return True
    else :
        return False
```

```python
sal['legal_drinker']=sal['age'].apply(majority)
sal.head()
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	F
0	GP	F	18	U	GT3	A	4	4	At_home	Teac
1	GP	F	17	U	GT3	T	1	1	At_home	O
2	GP	F	15	U	LE3	T	1	1	At_home	O
3	GP	F	15	U	GT3	T	4	2	Health	Servi
4	GP	F	16	U	GT3	T	3	3	Other	O

7 Multiply every number of the dataset by 10

```python
def mul10(x) :
    if type(x) == int :
        return 10*x
    return x
```

```
sal.applymap(mul10)
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	180	U	GT3	A	40	40	At_home	Te
1	GP	F	170	U	GT3	T	10	10	At_home	
2	GP	F	150	U	LE3	T	10	10	At_home	
3	GP	F	150	U	GT3	T	40	20	Health	Se
4	GP	F	160	U	GT3	T	30	30	Other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	200	U	LE3	A	20	20	Services	Se
391	MS	M	170	U	LE3	T	30	10	Services	Se
392	MS	M	210	R	GT3	T	10	10	Other	
393	MS	M	180	R	LE3	T	30	20	Services	
394	MS	M	190	U	LE3	T	10	10	Other	At_

395 rows × 13 columns

```
sal
```

```
     school  sex  age  address  famsize  Pstatus  Medu  Fedu      Mjob
  0      GP    F   18        U      GT3        A     4     4   At_home   Te
  1      GP    F   17        U      GT3        T     1     1   At_home
  2      GP    F   15        U      LE3        T     1     1   At_home
  3      GP    F   15        U      GT3        T     4     2    Health   Se
  4      GP    F   16        U      GT3        T     3     3     Other
...     ...  ...  ...      ...      ...      ...   ...   ...       ...
390      MS    M   20        U      LE3        A     2     2  Services   Se
391      MS    M   17        U      LE3        T     3     1  Services   Se
392      MS    M   21        R      GT3        T     1     1     Other
393      MS    M   18        R      LE3        T     3     2  Services
394      MS    M   19        U      LE3        T     1     1     Other  At_

395 rows × 13 columns
```

```
salmul10=sal.applymap(mul10)
salmul10
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	
0	GP	F	180	U	GT3	A	40	40	At_home	Te
1	GP	F	170	U	GT3	T	10	10	At_home	
2	GP	F	150	U	LE3	T	10	10	At_home	
3	GP	F	150	U	GT3	T	40	20	Health	Se
4	GP	F	160	U	GT3	T	30	30	Other	
...	...	...	...	...	...	...	...	...	...	
390	MS	M	200	U	LE3	A	20	20	Services	Se
391	MS	M	170	U	LE3	T	30	10	Services	Se
392	MS	M	210	R	GT3	T	10	10	Other	
393	MS	M	180	R	LE3	T	30	20	Services	
394	MS	M	190	U	LE3	T	10	10	Other	At_

395 rows × 13 columns