Started on	Wednesday, 10 September 2025, 9:36 AM
State	Finished
Completed on	Wednesday, 10 September 2025, 11:59 AM
Time taken	2 hours 22 mins
Overdue	22 mins 50 secs
Grade	80.00 out of 100.00

```
Question 1
Correct
Mark 20.00 out of 20.00
```

Create a Pandas program to join two dataframes using keys from right dataframe only.

For example:

```
Input
                                                                                   Result
{'key1': ['K0', 'K0', 'K1', 'K2'], 'key2': ['K0', 'K1', 'K0', 'K1'], 'P': ['P0', 'P1', 'P2',
                                                                                   Original DataFrames:
'P3'],'Q': ['Q0', 'Q1', 'Q2', 'Q3']}
                                                                                    key1 key2
                                                                                              Р
{'key1': ['K0', 'K1', 'K1', 'K2'],'key2': ['K0', 'K0', 'K0', 'K0'],'R': ['R0', 'R1', 'R2',
                                                                                   0 K0
                                                                                          K0 P0 Q0
'R3'],'S': ['S0', 'S1', 'S2', 'S3']}
                                                                                      Κ0
                                                                                          K1 P1 Q1
                                                                                          K0 P2 Q2
                                                                                   2 K1
                                                                                   3 K2 K1 P3 Q3
                                                                                     key1 key2 R S
                                                                                      K0 K0 R0 S0
                                                                                   0
                                                                                   1
                                                                                      K1
                                                                                           K0 R1 S1
                                                                                      K1
                                                                                          K0 R2 S2
                                                                                   2
                                                                                      K2
                                                                                          K0 R3 S3
                                                                                   Merged Data (keys from
                                                                                   data2):
                                                                                     key1 key2
                                                                                   R S
                                                                                   0 K0 K0 P0
                                                                                                   00
                                                                                   RØ SØ
                                                                                          K0 P2
                                                                                   1
                                                                                      K1
                                                                                                   02
                                                                                   R1 S1
                                                                                   2
                                                                                      K1
                                                                                          K0 P2
                                                                                                   Q2
                                                                                   R2 S2
                                                                                   3 K2 K0 NaN NaN
                                                                                   R3 S3
                                                                                   Merged Data (keys from
                                                                                   data1):
                                                                                     key1 key2 R
                                                                                                   S
                                                                                      Q
                                                                                      Κ0
                                                                                          K0 R0 S0
                                                                                   P0 Q0
                                                                                          K1 NaN NaN
                                                                                   1 K0
                                                                                   P1 Q1
                                                                                   2
                                                                                      K1 K0 R1
                                                                                   P2 Q2
                                                                                   3 K1 K0 R2 S2
                                                                                   P2 Q2
                                                                                   4
                                                                                      Κ2
                                                                                          K1 NaN NaN
                                                                                   P3 Q3
```

Answer: (penalty regime: 0 %)

```
import pandas as pd
    a=eval(input())
3
   b=eval(input())
    df1=pd.DataFrame(a)
   df2=pd.DataFrame(b)
5
   print("Original DataFrames:")
    print(df1)
7
8
    print("----")
    print(df2)
9
10
    print("Merged Data (keys from data2):")
11
    df_merge=pd.merge(df1,df2,on=["key1","key2"],how='right')
12
    print(df_merge)
13
   print()
    print("Merged Data (keys from data1):")
15
16
    df_merge2=pd.merge(df2,df1,on=["key1","key2"],how='right')
17
   print(df_merge2)
```

	Input	Expected Got	
•	{'key1': ['K0', 'K0', 'K1', 'K2'],'key2': ['K0', 'K1', 'K0',	Original Original	
	'K1'],'P': ['P0', 'P1', 'P2', 'P3'],'Q': ['Q0', 'Q1', 'Q2',	DataFrames: DataFrames:	
	'03']}	key1 key2 P Q key1 key2 P ()
	{'key1': ['K0', 'K1', 'K1', 'K2'], 'key2': ['K0', 'K0', 'K0',	0 K0 K0 P0 Q0 0 K0 K0 P0 Q0	-
	'KO'],'R': ['RO', 'R1', 'R2', 'R3'],'S': ['SO', 'S1', 'S2',	1 K0 K1 P1 Q1 1 K0 K1 P1 Q	
	'S3']}	2 K1 K0 P2 Q2 2 K1 K0 P2 Q2	
	33]]	3 K2 K1 P3 Q3 3 K2 K1 P3 Q3	
		3 K2 K1 13 Q3 3 K2 K1 13 Q.	
		key1 key2 R S key1 key2 R S	:
		0 K0 K0 R0 S0 0 K0 K0 R0 S0	
		1 K1 K0 R1 S1 1 K1 K0 R1 S2	
		2 K1 K0 R2 S2 2 K1 K0 R2 S2	
		3 K2 K0 R3 S3 3 K2 K0 R3 S3	
		Merged Data (keys Merged Data (keys	
		from data2): from data2):	
		key1 key2 P key1 key2 P	
		0 R S 0 R S	
		0 K0 K0 P0 0 K0 K0 P0	
		Q0 R0 S0 Q0 R0 S0	
		1 K1 K0 P2 1 K1 K0 P2	
		02 R1 S1	
		2 K1 K0 P2 2 K1 K0 P2	
		02 R2 S2	
		3 K2 K0 NaN 3 K2 K0 NaN	
		NaN R3 S3 NaN R3 S3	
		Merged Data (keys Merged Data (keys	
		from data1): from data1):	
		key1 key2 R key1 key2 R	
		S P Q S P Q	
		0 K0 K0 R0 0 K0 K0 R0	
		S0 P0 Q0 S0 P0 Q0	
		1 K0 K1 NaN 1 K0 K1 NaN	
		NaN P1 Q1 NaN P1 Q1	
		2 K1 K0 R1 2 K1 K0 R1	
		S1 P2 Q2 S1 P2 Q2	
		3 K1 K0 R2 3 K1 K0 R2	
		S2 P2 Q2 S2 P2 Q2	
		4 K2 K1 NaN 4 K2 K1 NaN	
		NaN P3 Q3 NaN P3 Q3	

Passed all tests! 🗸

Correct

Question ${\bf 2}$

Correct

Mark 20.00 out of 20.00

Write a Numpy program to Convert a 1-D array into a 2-D array with 3 rows

For example:

Input	Result
[0, 1, 2, 3, 4, 5, 6, 7, 8] The original array: [0 1 2 3 4 5 6 7 8]	The original array: [0 1 2 3 4 5 6 7 8]
	3 x 3 Array: [[0 1 2] [3 4 5] [6 7 8]]

Answer: (penalty regime: 0 %)

```
import numpy as np
a=eval(input())
b=np.array(a)
print("The original array:\n",b)

print()
print("3 x 3 Array:\n",np.reshape(b,(3,3)))
```

	Input	Expected	Got	
~	[0, 1, 2, 3, 4, 5, 6, 7, 8]	The original array:	The original array:	~
	The original array: [0 1 2 3 4 5 6 7 8]	[0 1 2 3 4 5 6 7 8]	[0 1 2 3 4 5 6 7 8]	
		3 x 3 Array:	3 x 3 Array:	
		[[0 1 2]	[[0 1 2]	
		[3 4 5]	[3 4 5]	
		[6 7 8]]	[6 7 8]]	
~	[11, 12, 13, 14, 15, 16, 17, 18, 19]	The original array:	The original array:	~
		[11 12 13 14 15 16 17 18 19]	[11 12 13 14 15 16 17 18 19]	
		3 x 3 Array:	3 x 3 Array:	
		[[11 12 13]	[[11 12 13]	
		[14 15 16]	[14 15 16]	
		[17 18 19]]	[17 18 19]]	

Passed all tests! 🗸

Correct

Question **3**Correct

Mark 20.00 out of 20.00

Write a Python Pandas program first to carry out the Multiplication and Division mathematics operations for the following two series a1 and a2.

For example:

Input	Result
[6, 6, 6, 6, 6]	Multiplication of two Series:
[1, 2, 5, 7, 9]	0 6
	1 12
	2 30
	3 42
	4 54
	dtype: int64
	Division of Series1 by Series2:
	0 6.000000
	1 3.000000
	2 1.200000
	3 0.857143
	4 0.666667
	dtype: float64

Answer: (penalty regime: 0 %)

```
import pandas as pd
a = eval(input())
b = eval(input())
c = pd. Series(a)
d = pd. Series(b)
print("Multiplication of two Series:")
print(c*d)
print("Division of Series1 by Series2:")
print(c/d)
```

	Input	Expected	Got	
~	[6, 6, 6, 6, 6]	Multiplication of two Series:	Multiplication of two Series:	~
	[1, 2, 5, 7, 9]	0 6	0 6	
		1 12	1 12	
		2 30	2 30	
		3 42	3 42	
		4 54	4 54	
		dtype: int64	dtype: int64	
		Division of Series1 by Series2:	Division of Series1 by Series2:	
		0 6.000000	0 6.000000	
		1 3.000000	1 3.000000	
		2 1.200000	2 1.200000	
		3 0.857143	3 0.857143	
		4 0.666667	4 0.666667	
		dtype: float64	dtype: float64	

	Input	Expected	Got			
~	[1,3,5,7,9]	Multiplication of two Series:	~			
	[2,4,6,8,10]	0 2	0 2			
		1 12	1 12			
		2 30	2 30			
		3 56	3 56			
		4 90	4 90			
		dtype: int64	dtype: int64			
		Division of Series1 by Series2:	Division of Series1 by Series2:			
		0 0.500000	0 0.500000			
		1 0.750000	1 0.750000			
		2 0.833333	2 0.833333			
		3 0.875000	3 0.875000			
		4 0.900000	4 0.900000			
		dtype: float64	dtype: float64			

Passed all tests! 🗸

Correct

Question ${\bf 4}$

Correct

Mark 20.00 out of 20.00

Create a numpy program to find the sum of first column in a given numpy array.

For example:

Inp	Input [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]								Re	su	lt				
[1,	2,	3,	4,	5,	6,	7,	8,	9,	10,	11,	12]	[[1	2	3]
												[4	5	6] 9]
												[7	8	9]
												[:	10	11	12]]
												22			

Answer: (penalty regime: 0 %)

```
import numpy as np
a=eval(input())
b=np.array(a)
c=np.reshape(b,(4,3))
print(c)
d=np.sum(c[:,0])
print(d)
```

	Input	Expected	Got	
~	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]	[4 5 6] [7 8 9]	[[1 2 3] [4 5 6] [7 8 9] [10 11 12]] 22	~
~	[2,4,6,8,10,12,14,16,18,20,22,24]	[8 10 12]	[[2 4 6] [8 10 12] [14 16 18] [20 22 24]] 44	~

Passed all tests! ✓

Correct

Question **5**Incorrect Mark 0.00 out of 20.00

Create a Python program to write the user given employee details into a csv file and read the content from the newly created file.

note:fieldnames = ['emp_name', 'dept', 'birth_month']

For example:

Test	Input	Result
<pre>with open('employee_file2.csv', 'r') as file: csvreader = csv.reader(file) header = next(csvreader) for row in csvreader: rows.append(row) print(header) print(rows)</pre>	<pre>{'emp_name': 'John Smith', 'dept': 'Accounting', 'birth_month': 'November'} {'emp_name': 'Erica Meyers', 'dept': 'IT', 'birth_month': 'March'}</pre>	<pre>['emp_name', 'dept', 'birth_month'] [['John Smith', 'Accounting', 'November'], ['Erica Meyers', 'IT', 'March']]</pre>

Answer: (penalty regime: 0 %)

	Test	Input	Expected	Got	
×	with	{'emp_name': 'John	['emp_name', 'dept',	***Run error***	×
	open('employee_file2.csv',	Smith', 'dept':	'birth_month']	Traceback (most recent	
	'r') as file:	'Accounting',	[['John Smith',	call last):	
	csvreader =	'birth_month':	'Accounting',	File	
	csv.reader(file)	'November'}	'November'], ['Erica	"testerpython3",	
	header = next(csvreader)	{'emp_name': 'Erica	Meyers', 'IT', 'March']]	line 2, in <module></module>	
	for row in csvreader:	Meyers', 'dept': 'IT',		csvreader =	
	rows.append(row)	'birth_month': 'March'}		csv.reader(file)	
	print(header)			NameError: name 'csv' is	
	print(rows)			not defined	

Testing was aborted due to error.

Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect