**ADTA-5340 Mid Term Practical**

**Q1)**

**Output:**

Enter an integer value:500

The user has entered this value: 500

**Description:**

1. User Input is taken and stored in anIntValue variable using input() function. I have entered 500 for this question.
2. Printing the user entered value in the output console. The result will be 500 with some text.

**Q2)**

**Output: NameError**

**Error:**

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

Cell In[2], line 3

**1** x = 10.5

**2** y = 5

----> 3 Print("Data type of x: ",type(x),'**\n**')

**4** print("Data type of y: ",type(y),'**\n**')

NameError: name 'Print' is not defined

**Correction:**

Error is in the name of print. All the keywords in python are in lowercase letters except True, False and None. Change in the code at line #3 Print is having uppercase letter ‘P’ instead of ‘p’. So, correct code is print().

**Output after correction:**

Data type of x: <class 'float'>

Data type of y: <class 'int'>

**Description:**

1. Creating new variables x, y with values 10.5, 5
2. Printing the datatype of above variables in the output

**Q3)**

**Output: NameError**

**Error:**

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

Cell In[4], line 2

**1** aDaystring="6/1/2021"

----> 2 aList=aDayString.split('/')

**3** print(aList)

NameError: name 'aDayString' is not defined

**Correction:**

We need to make correction at line #2 since aDaystring which is defined in line #1 is not used in the line #2 correctly. So correct variable name aDaystring to aDayString

**Output after Correction:**

['6', '1', '2021']

**Description:**

1. New variable, aDaystring created with value “6/1/2021”
2. aList variable is created and assigned with aDaystring data after splitted using ‘/’. So the list of each values will be stored in aList.
3. Printing the aList value.

**Q4)**

**Output: SyntaxError**

**Error:**

Cell In[6], line 1

list(range(1,21)]

^

SyntaxError: closing parenthesis ']' does not match opening parenthesis '('

**Correction:**

There is an incorrect bracket in the code.Which resulted in an error. So, change ‘]’ to ‘)’

**Output after Correction:**

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

**Description**:

A range of values from 1 to 20 is created​ and converted to list.

Q5)

**Error: NameError**

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

Cell In[8], line 1

----> 1 list(Range(0,-20,-3))

NameError: name 'Range' is not defined

**Correction:**

All the keywords in Python are in lowercase letters except True, False and None. In line #1, range function is written as Range with capital ‘r’(‘R’). So, we need to change it to lowercase ‘r’.

**Output after Correction:**

[0, -3, -6, -9, -12, -15, -18]

**Description:**

A range of values from 0 to -20 with step size of -3 is created​ and converted to list.

Q6)

**Error: SyntaxError**

Cell In[10], line 1

Import pandas as pd

^

SyntaxError: invalid syntax

**Correction:**

All the keywords in Python are in lowercase letters. Here, import is written in title case letters. So, correct it with lowercase “import”.

**Output after Correction**

0

0 5

1 10

2 33

3 14

4 245

**Description:**

1. Import pandas library and aliased as ‘pd’
2. New list, aList is created with values.
3. New dataframe, df is created using above list data
4. Printing the variable df.

Q7)

**Output: NameError**

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

Cell In[12], line 4

**2** aDict= {"Name":["Sally", "Joe", "Jose", "Susan", "Saanvi"], 'Age':[20,30,40, 25,23], 'Time':[2.0,2.2,2.5,2.55,3]}

**3** df=pd.DataFrame(aDict,index=['1st Place','2nd Place','3rd Place','4th Place', '5th Place'])

----> 4 print(DF)

NameError: name 'DF' is not defined

**Correction:**

In this code, incorrect variable DF is used in line #4. Since Python is case-sensitive language, df and DF is treated differently.

**Output after Correction:**

Name Age Time

1st Place Sally 20 2.00

2nd Place Joe 30 2.20

3rd Place Jose 40 2.50

4th Place Susan 25 2.55

5th Place Saanvi 23 3.00

**Description**

1. Importing pandas library and aliased as pd
2. New dictionary variable created, aDict is created with some values of Name, Age and Time.
3. New dataframe, df is created using above values of dictionary and added index to it
4. Printing the dataframe, df.

Q8)

**Output: KeyError**

**Error:**

---------------------------------------------------------------------------

KeyError Traceback (most recent call last)

File ~/anaconda3/lib/python3.10/site-packages/pandas/core/indexes/base.py:3802, in Index.get\_loc(self, key, method, tolerance)

**3801** **try**:

-> 3802 **return** self.\_engine.get\_loc(casted\_key)

**3803** **except** **KeyError** **as** err:

File ~/anaconda3/lib/python3.10/site-packages/pandas/\_libs/index.pyx:138, in pandas.\_libs.index.IndexEngine.get\_loc()

File ~/anaconda3/lib/python3.10/site-packages/pandas/\_libs/index.pyx:165, in pandas.\_libs.index.IndexEngine.get\_loc()

File pandas/\_libs/hashtable\_class\_helper.pxi:5745, in pandas.\_libs.hashtable.PyObjectHashTable.get\_item()

File pandas/\_libs/hashtable\_class\_helper.pxi:5753, in pandas.\_libs.hashtable.PyObjectHashTable.get\_item()

KeyError: 'five'

The above exception was the direct cause of the following exception:

KeyError Traceback (most recent call last)

Cell In[14], line 7

**5** df['three']=pd.Series([1,20,30], index=['z', 'v','y'])

**6** print(df)

----> 7 df['four'] = df['one'] + df['five']

**8** print('**\n**')

**9** print(df)

File ~/anaconda3/lib/python3.10/site-packages/pandas/core/frame.py:3807, in DataFrame.\_\_getitem\_\_(self, key)

**3805** **if** self.columns.nlevels > 1:

**3806** **return** self.\_getitem\_multilevel(key)

-> 3807 indexer = self.columns.get\_loc(key)

**3808** **if** is\_integer(indexer):

**3809** indexer = [indexer]

File ~/anaconda3/lib/python3.10/site-packages/pandas/core/indexes/base.py:3804, in Index.get\_loc(self, key, method, tolerance)

**3802** **return** self.\_engine.get\_loc(casted\_key)

**3803** **except** **KeyError** **as** err:

-> 3804 **raise** **KeyError**(key) **from** **err**

**3805** **except** **TypeError**:

**3806** # If we have a listlike key, \_check\_indexing\_error will raise

**3807** # InvalidIndexError. Otherwise we fall through and re-raise

**3808** # the TypeError.

**3809** self.\_check\_indexing\_error(key)

KeyError: 'five'

**Correction:**

Calling the column name, ‘five’ which does not exist in the dataframe and resulted in an error. The correct column name used in this context is colum ‘three.’

**Output after Correction:**

one two three

t NaN 10.0 NaN

v 5.0 NaN 20.0

w 4.0 5.0 NaN

x 3.0 NaN NaN

y 2.0 2.0 30.0

z 1.0 1.0 1.0

one two three four

t NaN 10.0 NaN NaN

v 5.0 NaN 20.0 25.0

w 4.0 5.0 NaN NaN

x 3.0 NaN NaN NaN

y 2.0 2.0 30.0 32.0

z 1.0 1.0 1.0 2.0

**Description:**

1. Importing pandas with alias name as pd
2. A dictionary, aDictOfSeries created with values of ‘one’ and ‘two’.
3. New dataframe, df is created using above created dictionary.
4. New column in dataframe, df is created and named as ‘three’ with values of series.
5. Printing the dataframe, df.
6. New column, ‘four’ is created, with addition of columns one and three. Some number with NaN with result in NaN
7. Printing the dataframe, df with the update above.

Q9)

**Output: AttributeError**

**Error:**

---------------------------------------------------------------------------

AttributeError Traceback (most recent call last)

Cell In[16], line 3

**1** **import** **numpy** **as** **np**

**2** x= np.arange(15).reshape((3,5))

----> 3 y.sum(axis=0)

AttributeError: 'int' object has no attribute 'sum'

**Correction:**

Incorrect use of function on variable. Sum function is available for numpy arrays but in this code, it is applied on variable y which is declared as int variable in previous code. So Correction is change y to variable x, which is numpy array. Now, the code executes smoothly.

**Output after Correction:**

array([15, 18, 21, 24, 27])

**Description:**

1. Importing numpy library aliased as np
2. Numpy array, x is created using range from 0 to 14 and reshaped as 3 rows and 5 columns
3. Sum function is applied on above array. Since axis is 0 is given, values in the array are summed column-wise.

Q10)

**Output:**

Resulted in TypeError

**Error:**

TypeError Traceback (most recent call last)

Cell In[18], line 2

**1** aTuple = ("O'Reily", "Wiley","Google","Azure", "AWS")

----> 2 aTuple[2]= 'Pearson'

TypeError: 'tuple' object does not support item assignment

**Description:**

1. New tuple, aTuple variable created with values
2. Tuples in python are immutable. We cannot re-assign the values in the tuple.

Q11)

**Output: NameError**

**Error:**

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

Cell In[19], line 2

**1** sampl\_string= "It's a Wonderful Life."

----> 2 print (sample\_string[2:-4])

NameError: name 'sample\_string' is not defined

**Correction:**

There is a spelling mistake in the variable at line #2. So, sampl\_string must be at line #2 in place of sample\_string

**Output after Correction:**

's a Wonderful L

**Description:**

1. Sampl\_string variable created with string
2. Above variable is sliced to print sub-string from index 2 to -4(index 4 from right end)

Q12)

**Output:**

**Code 1:**

Hello and welcome to my interactive tutorial.

What is your name? Sri Charan

What is your age? 25

What city do you reside in? Denton, Texas

What is your email? sri@gmail.com

Thank you very much Sri Charan, you will be contacted at sri@gmail.com.

**Code 2:**

Create a password: 1234

Welcome to the portal Sri Charan

Please enter your password: 1234

Successful! Welcome back Sri Charan

**Description:**

**Code1:**

1. Printing message with welcome
2. Reading Name input from user and storing variable, name
3. Reading Age input from user and storing variable, age
4. Reading City input from user and storing variable, city
5. Reading Email input from user and storing variable, email
6. Printing message with thank you.

**Code2:**

1. Reading Password input from user and storing variable, password
2. Printing welcome message with name
3. Asking user to enter password again and storing it in variable password\_check
4. Using if condition, checking whether password and password\_check have same values. If yes, Success message is printed with name, else, “Sorry, no Entrance. Wrong password” is printed in the console.