Functions in Python

A function is set of statements that take input in the form of parameters, performs computation on the input and returns a result in the form of return statement

Syntax –

def `function-name` (`parameters if any`):

    # function code

    `return` statement

Note : It is a best practice to avoid usage of input() and print() functions in a function definition

WAF to calculate factorial of a number

In [5]:
```python
# Function Definition
def factorial(num):
    fact = 1
    for i in range(num, 1, -1):
        fact *= i
    return fact
```

In [3]:
```python
factorial(5)   # Function call
```

Out[3]: 120

Note - Function attributes

Function Definition

1. def - keyword
2. factorial() - name of function
3. num - parameter defined in function, used to take input required for algorithm
4. function implementation - algorithm/code
5. return statement - the value that has to returned back

Function Definition

1. function name ()
2. 5 - argument - object passed to the parameters

In [9]:
```python
# Function Definition
def factorial(num):
    if type(num) == int :
        fact = 1
        for i in range(num, 1, -1):
```

```
        fact *= i
    return fact
```

In [10]:
```python
print(factorial("abcd"))
```

None

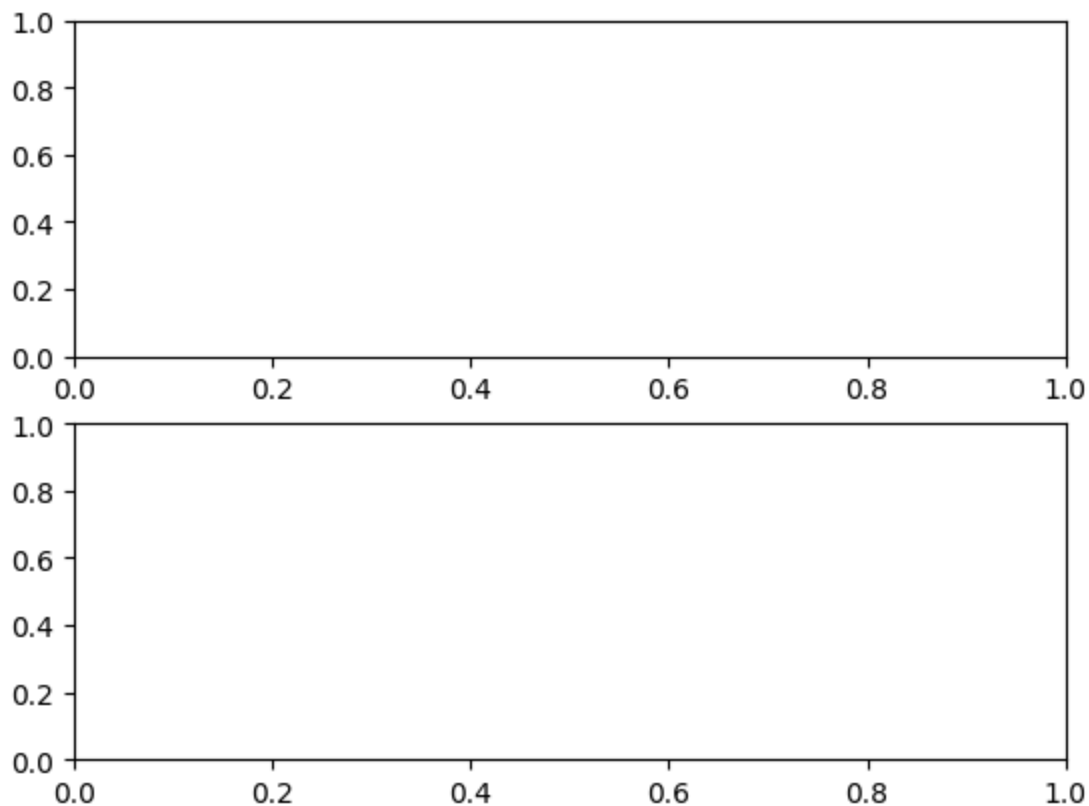**Note - If a function has no return statement it will return None as default**

In [7]:
```python
# Function returning multiple values
def func(num) :
    return num**2, num**3   # returns multiple values in a single tuple object

square, cube = func(5)   # storing the returned values into variable - unpacking of
print(square, cube)
```

25 125

In [13]:
```python
import matplotlib.pyplot as plt

fig, ax = plt.subplots(nrows=2)   # examples of unpacking of tuples
```


Function Arguments

- Required Positional Arguments
- Default Arguments
- Variable length Arguments
- Key-word Arguments
- Variable-length Keyword Arguments

Required Positional Argument

```python
In [12]: def demo(name, age):
             print(f"Name - {name} | Age - {age}")

         demo("Jane", 30)
         demo(30, "Jane")  # Position of arguments must be same as parameters
         demo("Jane")  # No of arguments = No of Parameters
```

```
Name - Jane | Age - 30
Name - 30 | Age - Jane
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[12], line 6
      4 demo("Jane", 30)
      5 demo(30, "Jane")
----> 6 demo("Jane")

TypeError: demo() missing 1 required positional argument: 'age'
```

Examples

```python
In [14]: lst = [1, 2, 3, 4]
         lst.insert(5, 2)  # 2 is inserted at index 5 - positional argument
         lst.insert(5)  # Required Argument
         lst
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[14], line 3
      1 lst = [1, 2, 3, 4]
      2 lst.insert(5, 2)  # 2 is inserted at index 5 - positional argument
----> 3 lst.insert(5)  # 2 is inserted at index 5
      5 lst

TypeError: insert expected 2 arguments, got 1
```

```python
In [15]: help(lst.insert)
```

```
Help on built-in function insert:

insert(index, object, /) method of builtins.list instance
    Insert object before index.
```

Example - 2

```python
In [18]: strg = "mississippi"
         strg.replace("i", "*")
         strg.replace("*", "i")  # Positional argumnent
         strg.replace("i")  # Missing argument
```

```
----------------------------------------------------------------------------
TypeError                                    Traceback (most recent call last)
Cell In[18], line 4
      2 strg.replace("i", "*")
      3 strg.replace("*", "i")   # Positional argumnent
----> 4 strg.replace("i")

TypeError: replace expected at least 2 arguments, got 1
```

In [19]: `help(strg.replace)`

```
Help on built-in function replace:

replace(old, new, count=-1, /) method of builtins.str instance
    Return a copy with all occurrences of substring old replaced by new.

      count
        Maximum number of occurrences to replace.
        -1 (the default value) means replace all occurrences.

    If the optional argument count is given, only the first count occurrences are
    replaced.
```

Default Argument

In [20]:
```
strg = "mississippi"
strg.replace("i", "*")   # count = -1 default, replacing all occurences
strg.replace("i", "*", 2)   # count = 2, replacing first 2 occurences
```

Out[20]: `'m*ss*ssippi'`

Example

In [21]: `list(enumerate(["Jack", "Jane", "Rosie"])) # default start = 0`

Out[21]: `[(0, 'Jack'), (1, 'Jane'), (2, 'Rosie')]`

In [22]: `list(enumerate(["Jack", "Jane", "Rosie"], start = 101)) # start = 101`

Out[22]: `[(101, 'Jack'), (102, 'Jane'), (103, 'Rosie')]`

In [24]:
```
result = enumerate(["Jack", "Jane", "Rosie"], start = 101)
print(type(result))
```

```
<class 'enumerate'>
```

In [23]: `help(enumerate)`

```
Help on class enumerate in module builtins:

class enumerate(object)
 |  enumerate(iterable, start=0)
 |
 |  Return an enumerate object.
 |
 |    iterable
 |       an object supporting iteration
 |
 |  The enumerate object yields pairs containing a count (from start, which
 |  defaults to zero) and a value yielded by the iterable argument.
 |
 |  enumerate is useful for obtaining an indexed list:
 |       (0, seq[0]), (1, seq[1]), (2, seq[2]), ...
 |
 |  Methods defined here:
 |
 |  __getattribute__(self, name, /)
 |      Return getattr(self, name).
 |
 |  __iter__(self, /)
 |      Implement iter(self).
 |
 |  __next__(self, /)
 |      Implement next(self).
 |
 |  __reduce__(...)
 |      Return state information for pickling.
 |
 |  ----------------------------------------------------------------------
 |  Class methods defined here:
 |
 |  __class_getitem__(...)
 |      See PEP 585
 |
 |  ----------------------------------------------------------------------
 |  Static methods defined here:
 |
 |  __new__(*args, **kwargs)
 |      Create and return a new object.  See help(type) for accurate signature.
```

Variable-length Argument

```python
In [29]: def demo(name, *args, age = 18):
             print(f"Name - {name} | Age - {age} | ARGS - {args}")

         demo("Jane", "abcd", 70, 80, 90, 19)
```

```
Name - Jane | Age - 18 | ARGS - ('abcd', 70, 80, 90, 19)
```

Example

```python
In [30]: help(print)
```

```
Help on built-in function print in module builtins:

print(*args, sep=' ', end='\n', file=None, flush=False)
    Prints the values to a stream, or to sys.stdout by default.

    sep
      string inserted between values, default a space.
    end
      string appended after the last value, default a newline.
    file
      a file-like object (stream); defaults to the current sys.stdout.
    flush
      whether to forcibly flush the stream.
```

Key-word Argument

In [27]:
```python
demo("Jane", 60, 70, 80, 90, age = 19)
```

Name - Jane | Age - 19 | ARGS - (60, 70, 80, 90)

In [32]:
```python
name = "Anna"
print("Welcome", name, ". How are you?", sep = "_")
```

Welcome_Anna_. How are you?

Variable-length Keyword Argument

In [45]:
```python
def demo(name, *args, age = 18, **kwargs):
    print(f"Name - {name} | Age - {age} | ARGS - {args} | KWARGS - {kwargs} | Gende

demo("Jane", "abcd", 70, 80, 90, age = 19, mob = 9876543, gender = "F")
```
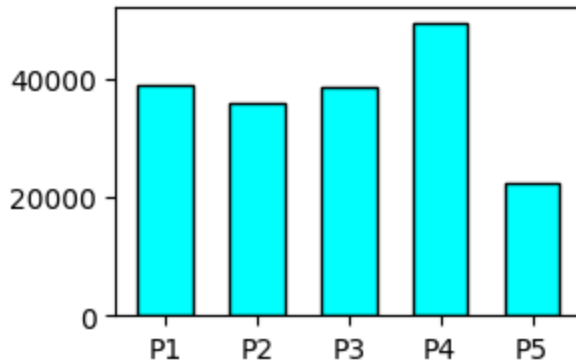
Name - Jane | Age - 19 | ARGS - ('abcd', 70, 80, 90) | KWARGS - {'mob': 9876543, 'ge
nder': 'F'} | Gender - F

Example

In [38]:
```python
import matplotlib.pyplot as plt
import random as r

products = ["P1", "P2", "P3", "P4", "P5"]
sales = [r.randint(1000, 50000) for i in range(5)]
plt.figure(figsize=(3, 2))
plt.bar(products, sales, width=0.6, color = "cyan", edgecolor = "black")
```

Out[38]:   <BarContainer object of 5 artists>

Application of packing and unpacking of tuple wrt functions

Ex. packing and unpacking wrt return statement

```
In [48]:  def func(num) :
              return num**2, num**3  # packing
          sq, cu = func(5)  # unpacking
```

Ex. packing of tuples wrt function arguments

```
In [46]:  def demo(*args):
              print(args)  # args - tuple object

          demo(10, 20, 30, 40, 50)  # packing of tuples
```

```
(10, 20, 30, 40, 50)
```

```
In [56]:  # Function to write data to a file
          def write_to_file(name, age, gender):
              with open("data.txt", "a") as file :
                  file.write(f"Name - {name} | Age - {age} | Gender - {gender}\n")
                  print(f"Data for {name} added to the file.")
```

Ex. unpacking of tuples wrt function arguments

```
In [54]:  data = [["Jane", 30, "F"], ["George", 25, "M",], ["Jack", 35, "M"]]
          for lst in data :
              write_to_file(*lst)  # *lst is using unpacking of tuples to pass the values in
```

```
Data for Jane added to the file.
Data for George added to the file.
Data for Jack added to the file.
```

Ex. packing of key-value pairs to a dict

```
In [47]:  def demo(**kwargs):
              print(kwargs)  # kwargs - dict object

          demo(name = "Jane", age = "30")  # packing of dict
```

```
{'name': 'Jane', 'age': '30'}
```

Ex. unpacking of elements in a dict

```
In [59]:  data = [{"name" : "Jane", "gender" : "F", "age" : 30}, {"age" :25, "gender" : "M",
          for i in data :
              write_to_file(**i)
```

Data for Jane added to the file.
Data for Jack added to the file.

Note - Brackets

1. `()` - tuples, parenthesis, function call
2. `[]` - lists, indexing in lists/tuples/dict
3. `{}` - sets, dict, inserting variables in formatted string

Purpose of `/` and `*` in a function definition

- `*` - Arguments defined `after` * must be keyword only
- `/` - Arguments defined `before` / must be positional only

```
In [63]:  def demo(name, age):
              print(f"Name - {name} | Age - {age}")

          demo("Jane", 30) # Positional only
          demo (age = 30, name = "Jane") # keyword-only
```

Name - Jane | Age - 30
Name - Jane | Age - 30

```
In [64]:  def demo(name, /, age):
              print(f"Name - {name} | Age - {age}")

          demo("Jane", 30)
          demo("Jane", age = 30)
          demo(name = "Jane", age = 30)   # Error - name must be positional only
```

Name - Jane | Age - 30
Name - Jane | Age - 30

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[64], line 6
      4 demo("Jane", 30)
      5 demo("Jane", age = 30)
----> 6 demo(name = "Jane", age = 30)

TypeError: demo() got some positional-only arguments passed as keyword arguments: 'n
ame'
```

Example - strg.replace()

```
In [65]:  strg = "mississippi"
          strg.replace("i", "*", 2)
```

Out[65]: 'm*ss*ssippi'
```

```
In [67]:   strg = "mississippi"
           strg.replace("i", "*", count = 2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[67], line 2
      1 strg = "mississippi"
----> 2 strg.replace("i", "*", count = 2)

TypeError: str.replace() takes no keyword arguments
```

```
In [66]:   help(strg.replace)
```

```
Help on built-in function replace:

replace(old, new, count=-1, /) method of builtins.str instance
    Return a copy with all occurrences of substring old replaced by new.

      count
          Maximum number of occurrences to replace.
          -1 (the default value) means replace all occurrences.

    If the optional argument count is given, only the first count occurrences are
    replaced.
```

```
In [69]:   def demo(name, *, age):
               print(f"Name - {name} | Age - {age}")

           demo("Jane", age = 30)
           demo(name = "Jane", age = 30)
           demo("Jane", 30)  # Error - age must be keyword only as it is after *
```

```
Name - Jane | Age - 30
Name - Jane | Age - 30
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[69], line 6
      4 demo("Jane", age = 30)
      5 demo(name = "Jane", age = 30)
----> 6 demo("Jane", 30)  # Error - age must be keyword only as it is after *

TypeError: demo() takes 1 positional argument but 2 were given
```

Example - sorted()

```
In [70]:   help(sorted)
```

```
Help on built-in function sorted in module builtins:

sorted(iterable, /, *, key=None, reverse=False)
    Return a new list containing all items from the iterable in ascending order.

    A custom key function can be supplied to customize the sort order, and the
    reverse flag can be set to request the result in descending order.
```

sorted(iterable, /, *, key=None, reverse=False)

- iterable - must be positional only
- key and reverse - must be keyword only

Variable Scope

- A namespace is a Container where names are mapped to variables
- A scope defines the hierarchical order in which the namespaces need to be
- searched in order to obtain the name-to-object mapping
- Scope defined the accessibility and lifetime of a variable

The LEGB rule

The `LEGB` rule decides the order in which the namespaces are to be searched for variable scoping

Variable scope hierarchy:

1. Built-In (B): Reserved names in Python
2. Global Variable (G): Defined at the uppermost level
3. Enclosed (E): Defined inside enclosing or nested functions
4. Local Variable (L): Defined inside a function

In [82]:
```python
from math import pi    # built-in scope
def outer() :
    # pi = 10   # enclosed scope
    def inner():
        # pi = 5    # local scope
        print(pi)
    inner()

# pi = 15    # Global scope
outer()
```

3.141592653589793

Lambda Function

- A lambda function is also called as an anonymous function as it is a function that is defined without a name.

- A lambda function behaves similar to a standard function except it is defined in one-line.

- It is defined using a lambda key-word.

- Lambda functions can have any number of arguments but only one expression. The expression is evaluated and returned.

- Lambda functions can be used wherever function objects are required.

- Syntax of Lambda Function –

lambda *parameters* : *expression*

Write a lambda function to return addition of 2 numbers

```
In [84]:  add = lambda x, y : x + y
          add(2, 3)
```

Out[84]: 5

Write a lambda function to return square of the number

```
In [85]:  sq = lambda x : x** 2
          sq(5)
```

Out[85]: 25

Function Object

- Everything in Python is an object, including functions.
- You can assign them to variables, store them in data structures, and pass or return them to and from other functions
- Functions in Python can be passed as arguments to other functions, assigned to variables or even stored as elements in various data structures.

function definition/implemenation

```
In [86]:  def func(a, b):  # -> function definition
              if a < b :
                  return a
              else:
                  return b
```

function call

```
In [87]:  # function call
          var = func(3, 4)
          var
```

Out[87]: 3

function object

```
In [88]:   # function object
           var = func
           var

Out[88]:   <function __main__.func(a, b)>

In [89]:   type(var)

Out[89]:   function

In [90]:   var(3, 4)

Out[90]:   3

In [91]:   var = len   # var varible can now access the len func
           var("abcd")

Out[91]:   4
```

Applilcations of Function Object

Ex. WAP to sort a list of strings as per the last character.

```
In [94]:   lst = ["flight", "car", "train", "bike"]
           sorted(lst) # alphabetically sort

Out[94]:   ['bike', 'car', 'flight', 'train']

In [96]:   lst = ["flight", "car", "train", "bike"]
           sorted(lst, key = len, reverse = True) # no of characters in each word sort

Out[96]:   ['flight', 'train', 'bike', 'car']

In [97]:   lst = ["flight", "car", "train", "bike"]
           sorted(lst, key = lambda x : x[-1]) # sort by last character of each word

Out[97]:   ['bike', 'train', 'car', 'flight']

In [98]:   max(lst)

Out[98]:   'train'

In [100…   max(lst, key = len)

Out[100…   'flight'

In [102…   max(lst, key = min)

Out[102…   'flight'
```

Note -

- There are many functions like sorted which take function object as an argument.
- Use a built-in function if available, else defined a custom function.
- If the logic for custom function is one-liner use lambda function else use standard user-defined function

Ex. WAP to display the student details in sorted order DESC of their marks.

```
In [103…  students = {"Jane" : 40, "Max" : 50, "Sam" : 45, "Mary" : 70}
          sorted(students)  # list of keys sorted alphabetically
```

```
Out[103…  ['Jane', 'Mary', 'Max', 'Sam']
```

```
In [106…  students = {"Jane" : 40, "Max" : 50, "Sam" : 45, "Mary" : 70}
          dict(sorted(students.items()))  # dict sorted alphabetically wrt keys
```

```
Out[106…  {'Jane': 40, 'Mary': 70, 'Max': 50, 'Sam': 45}
```

```
In [112…  students = {"Jane" : 40, "Max" : 50, "Sam" : 45, "Mary" : 70}
          dict(sorted(students.items(), key = lambda tup : tup[1], reverse = True))
```

```
Out[112…  {'Mary': 70, 'Max': 50, 'Sam': 45, 'Jane': 40}
```

map() - filter() - reduce()

map(func_obj, sequence)

- func_obj : It is a function object to which map passes each element of given sequence.
- sequence : It is a sequence which is to be mapped.
- Returns a sequence of the results after applying the given function to each item of a given iterable.

Ex. WAP to map the given list to list of squares

```
In [114…  lst = [1, 2, 3, 4]
          list(map(lambda x : x**2, lst))
```

```
Out[114…  [1, 4, 9, 16]
```

Ex. WAP to generate a list of square roots of the given list using map

```
In [117…  import math
          lst = [1, 4, 9, 16]
          list(map(math.sqrt, lst))
```

```
Out[117…  [1.0, 2.0, 3.0, 4.0]
```

filter(func_obj, sequence)

- func_obj : function that tests if each element of a sequence true or not. It should always be a Boolean function.

- sequence : It is a sequence which is to be filtered.
- Returns a sequence of filtered elements.

Ex. WAP to filter all the even numbers from the given list of numbers.

```python
In [118...   lst = [1, 2, 3, 4, 5, 6]
             list(filter(lambda x : x % 2 == 0 ,lst))
```

Out[118...   [2, 4, 6]

reduce(func_obj, seq)

- The reduce(func_obj, seq) function is used to apply a particular function passed in its argument to all of the list elements mentioned in the sequence passed along. This function is defined in "functools" module.
  Note: func_obj will always take two parameters

Working : the number just succeeding the second element and the result is again stored.

- At first step, first two elements of sequence are picked and the result is obtained.
- Next step is to apply the same function to the previously attained result and

- This process continues till no more elements are left in the container.
- The final returned result is returned and printed on console.

Ex. Reduce the list to summation of all numbers

```python
In [119...   from functools import reduce
             reduce(lambda x, y : x + y ,range(1, 11))
```

Out[119...   55

```python
In [120...   from functools import reduce
             reduce(lambda x, y : x + y ,range(1, 11), 10)
```

Out[120...   65

Additional Examples

Ex. WAP to extract all digits from a string (using filter)

```python
In [125...   strg = "abcd12345"
             list(filter(lambda st : st.isdigit(), strg))  # method call
```

Out[125...   ['1', '2', '3', '4', '5']

```python
In [129...   strg = "abcd12345"
             list(filter(str.isdigit, strg))  # method call
```

Out[129...   ['1', '2', '3', '4', '5']
```

```
In [123...   strg.isdigit()   # method called on strg object
```

```
Out[123...   False
```

```
In [126...   str.isdigit(strg)     # function called by class name
```

```
Out[126...   False
```

Ex. use reduce() to extract sum of all digits in a string

```
In [130...   from functools import reduce
             strg = "abcd12345"

             reduce(lambda x, y : x + int(y) if y.isdigit() else 0 ,strg, 0)
```

```
Out[130...   15
```

Ex. Combine all elements in the list into a single string object.

Capitalize first character of the string while concatenating

```
In [136...   strg = ["lion", "tiger", "horse"]
             "".join(strg).title()
             # o/p - LionTigerHorse
```

```
Out[136...   'Liontigerhorse'
```

```
In [138...   strg = ["lion", "tiger", "horse"]
             reduce(lambda x, y : x + y.title(), strg, "")
```

```
Out[138...   'LionTigerHorse'
```

Ex. Using map() multiply the two lists

```
In [140...   lst1 = [1, 2, 3, 4]
             lst2 = [5, 6, 7, 8]

             list(map(lambda x1, x2 : x1 * x2, lst1, lst2))
```

```
Out[140...   [5, 12, 21, 32]
```

Exception Handling

It may be convenient to recognize the problems in your python code before you put it to real use. But that does not always happen. Sometimes, problems show up when you run the code; sometimes, midway of that. A Python exception is an error that's detected during execution.

Python does not provide any compile time Exception Handling. Developer has to proactively recognize the need for exception handling.

- What are Errors and Exceptions
- Handling Exceptions
- Defining Clean-up Actions
- Predefined Clean-up Actions
- Raising Exceptions

Syntax Error

```
In [141...   for i in range(5)
                 print(i)
```

```
  Cell In[141], line 1
    for i in range(5)
                     ^
SyntaxError: expected ':'
```

Exception

Even if a statement or expression is syntactically correct, it may cause an error when an attempt is made to execute it. Errors detected during execution are called exceptions and are not unconditionally fatal

ZeroDivisionError

```
In [142...   a, b = 1, 0
             print(a/b)
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
Cell In[142], line 2
      1 a, b = 1, 0
----> 2 print(a/b)

ZeroDivisionError: division by zero
```

ValueError

```
In [143...   int("abcd")
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[143], line 1
----> 1 int("abcd")

ValueError: invalid literal for int() with base 10: 'abcd'
```

NameError

```
In [144...   print(z)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[144], line 1
----> 1 print(z)

NameError: name 'z' is not defined
```

FileNotFoundError

```
In [145...   open("abc.txt")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
Cell In[145], line 1
----> 1 open("abc.txt")

File ~\AppData\Local\Programs\Python\Python312\Lib\site-packages\IPython\core\intera
ctiveshell.py:324, in _modified_open(file, *args, **kwargs)
    317 if file in {0, 1, 2}:
    318     raise ValueError(
    319         f"IPython won't let you open fd={file} by default "
    320         "as it is likely to crash IPython. If you know what you are doing, "
    321         "you can use builtins' open."
    322     )
--> 324 return io_open(file, *args, **kwargs)

FileNotFoundError: [Errno 2] No such file or directory: 'abc.txt'
```

TypeError

```
In [146...   "2" + 2
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[146], line 1
----> 1 "2" + 2

TypeError: can only concatenate str (not "int") to str
```

```
In [147...   def func(a):
                 pass

             func()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[147], line 4
      1 def func(a):
      2     pass
----> 4 func()

TypeError: func() missing 1 required positional argument: 'a'
```

IndexError

```
In [148…    l = [1,2,3]
            l[10]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[148], line 2
      1 l = [1,2,3]
----> 2 l[10]

IndexError: list index out of range
```

KeyError

```
In [149…    d = {1:2, 3:4}

            d["abc"]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[149], line 3
      1 d = {1:2, 3:4}
----> 3 d["abc"]

KeyError: 'abc'
```

ModuleNotFoundError

```
In [150…    import math1
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[150], line 1
----> 1 import math1

ModuleNotFoundError: No module named 'math1'
```

AttributeError

```
In [151…    import math
            math.sq1()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[151], line 2
      1 import math
----> 2 math.sq1()

AttributeError: module 'math' has no attribute 'sq1'
```

```
In [152…    string = "abcd"
```

```
string.UPPER()
```

```
---------------------------------------------------------------------------
AttributeError                              Traceback (most recent call last)
Cell In[152], line 3
      1 string = "abcd"
----> 3 string.UPPER()

AttributeError: 'str' object has no attribute 'UPPER'
```

Handling Exceptions

try :

   #risky code

except :

    #code to handle error

else:

    #executed if everything goes fine

finally:

    #gets executed in either case.

In [153...
```python
try :
    file = open("file.txt")
except FileNotFoundError as e :
    print(e)
else:
    print("No exception thus executed")
finally:
    print("Will execute irrespective of exception")
```

```
[Errno 2] No such file or directory: 'file.txt'
Will execute irrespective of exception
```

In [154...
```python
try :
    file = open("data.txt")
    print("try block complete")
except FileNotFoundError as e :
    print(e)
else:
    print("No exception thus executed")
finally:
    print("Will execute irrespective of exception")
```

```
try block complete
No exception thus executed
Will execute irrespective of exception
```

Rasie an Exception

Ex. Define a function "division" implementing exception handling

In [159...
```python
def division(num, deno) :
    if type(num) == int and type(deno) == int :
        if deno != 0 :
            return num/deno
        else:
            raise ZeroDivisionError("Denominator is zero")
    else:
        raise TypeError("Numerator and Denominator must be integers")
```

In [161...
```python
division(10, "abc")
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[161], line 1
----> 1 division(10, "abc")

Cell In[159], line 8, in division(num, deno)
      6             raise ZeroDivisionError("Denominator is zero")
      7 else:
----> 8     raise TypeError("Numerator and Denominator must be integers")

TypeError: Numerator and Denominator must be integers
```

Ex. Write a function to take and validate int input from user. Value must be in range of 1-10

In [169...
```python
def validate_input() :
    try :
        num = int(input("Enter a number in the range of 1- 10"))
        if num in range(1, 11):
            return num
        else :
            raise Exception("Number must be in the range of 1-10")
    except ValueError as e:
        print("Number must be an integer")
        return validate_input()
    except Exception as e :
        print(e.args[0])
        return validate_input()
```

In [170...
```python
number = validate_input()
print(number)
# some task with number
```

```
Number must be an integer
5
```

Reading and Processing data

To set the file path

```
In [1]: import os
        os.getcwd() # gives path for current working dir
```

Out[1]: 'T:\\Material_general\\Emerson\\Classwork'

```
In [ ]: os.chdir(r"T:\Python\Trainer\Class_Notebooks\Python for Data Analytics\Classwork\Da
```

Read data from `customers.txt` **file**

```
In [2]: file = open("customers.txt")  # default file opens in read mode
        data = file.readlines()
        file.close()
```

```
In [3]: data
```

```
Out[3]:  ['4000001,Kristina,Chung,55,Pilot\n',
          '4000002,Paige,Chen,74,Teacher\n',
          '4000003,Sherri,Melton,34,Firefighter\n',
          '4000004,Gretchen,Hill,66,Computer hardware engineer\n',
          '4000005,Karen,Puckett,74,Lawyer\n',
          '4000006,Patrick,Song,42,Veterinarian\n',
          '4000007,Elsie,Hamilton,43,Pilot\n',
          '4000008,Hazel,Bender,63,Carpenter\n',
          '4000009,Malcolm,Wagner,39,Artist\n',
          '4000010,Dolores,McLaughlin,60,Writer\n',
          '4000011,Francis,McNamara,47,Therapist\n',
          '4000012,Sandy,Raynor,26,Writer\n',
          '4000013,Marion,Moon,41,Carpenter\n',
          '4000014,Beth,Woodard,65,\n',
          '4000015,Julia,Desai,49,Musician\n',
          '4000016,Jerome,Wallace,52,Pharmacist\n',
          '4000017,Neal,Lawrence,72,Computer support specialist\n',
          '4000018,Jean,Griffin,45,Childcare worker\n',
          '4000019,Kristine,Dougherty,63,Financial analyst\n',
          '4000020,Crystal,Powers,67,Engineering technician\n',
          '4000021,Alex,May,39,Environmental scientist\n',
          '4000022,Eric,Steele,66,Doctor\n',
          '4000023,Wesley,Teague,42,Carpenter\n',
          '4000024,Franklin,Vick,28,Dancer\n',
          '4000025,Claire,Gallagher,42,Musician\n',
          '4000026,Marian,Solomon,27,Lawyer\n',
          '4000027,Marcia,Walsh,64,Accountant\n',
          '4000028,Dwight,Monroe,45,Economist\n',
          '4000029,Wayne,Connolly,40,Real estate agent\n',
          '4000030,Stephanie,Hawkins,50,Human resources assistant\n',
          '4000031,Neal,Middleton,59,Civil engineer\n',
          '4000032,Gretchen,Goldstein,24,Engineering technician\n',
          '4000033,Tim,Watts,58,Lawyer\n',
          '4000034,Jerome,Johnston,38,Childcare worker\n',
          '4000035,Shelley,Weeks,25,Reporter\n',
          '4000036,Priscilla,Wilkerson,35,Agricultural and food scientist\n',
          '4000037,Elsie,Barton,27,Childcare worker\n',
          '4000038,Beth,Walton,73,Firefighter\n',
          '4000039,Erica,Hall,33,Police officer\n',
          '4000040,Douglas,Ross,27,Secretary\n',
          '4000041,Donald,Chung,65,Computer hardware engineer\n',
          '4000042,Katherine,Bender,44,Physicist\n',
          '4000043,Paul,Woods,63,Doctor\n',
          '4000044,Patricia,Mangum,67,Civil engineer\n',
          '4000045,Lois,Joseph,44,Musician\n',
          '4000046,Louis,Rosenthal,31,\n',
          '4000047,Christina,Bowden,47,Computer software engineer\n',
          '4000048,Darlene,Barton,54,Doctor\n',
          '4000049,Harvey,Underwood,70,Engineering technician\n',
          '4000050,William,Jones,53,Photographer\n',
          '4000051,Frederick,Baker,52,Writer\n',
          '4000052,Shirley,Merritt,21,Reporter\n',
          '4000053,Jason,Cross,56,Civil engineer\n',
          '4000054,Judith,Cooper,22,Economist\n',
          '4000055,Gretchen,Holmes,39,Childcare worker\n',
          '4000056,Don,Sharpe,53,Social worker\n',
```

```
'4000057,Glenda,Morgan,37,Real estate agent\n',
'4000058,Scott,Hoyle,40,Doctor\n',
'4000059,Pat,Allen,45,Secretary\n',
'4000060,Michelle,Rich,44,Artist\n',
'4000061,Jessica,Rich,37,Actor\n',
'4000062,Evan,Grant,66,Agricultural and food scientist\n',
'4000063,Melinda,Proctor,27,Teacher\n',
'4000064,Calvin,Diaz,65,Athlete\n',
'4000065,Eugene,Graham,52,Police officer\n',
'4000066,Vickie,Watkins,55,Computer support specialist\n',
'4000067,Luis,Hinton,69,Childcare worker\n',
'4000068,Allan,Marsh,67,Athlete\n',
'4000069,Melanie,Hewitt,47,Real estate agent\n',
'4000070,Marianne,Branch,53,Judge\n',
'4000071,Natalie,Walton,24,Recreation and fitness worker\n',
"4000072,Caroline,O'Brien,44,Computer support specialist\n",
'4000073,Arlene,Case,62,Musician\n',
'4000074,Kyle,Watts,39,Engineering technician\n',
'4000075,Calvin,Christensen,54,Architect\n',
'4000076,Gary,Parks,65,Pharmacist\n',
'4000077,Samantha,Hardin,27,Doctor\n',
'4000078,Sara,Lucas,44,Loan officer\n',
'4000079,Stacy,Eason,31,Musician\n',
'4000080,Gladys,Davidson,52,Recreation and fitness worker\n',
'4000081,Mike,Whitehead,26,Politician\n',
'4000082,Lynne,Rose,36,Loan officer\n',
'4000083,Faye,Sparks,61,Civil engineer\n',
'4000084,Diana,Moore,44,Computer support specialist\n',
'4000085,Leon,Pearson,24,Physicist\n',
'4000086,Ethel,Rodgers,30,Librarian\n',
'4000087,Steve,Graves,73,Nurse\n',
'4000088,Alison,Scarborough,66,Designer\n',
'4000089,Sherri,Sutton,75,Social worker\n',
'4000090,Patsy,Sinclair,48,Police officer\n',
'4000091,Kelly,Bowman,69,Childcare worker\n',
'4000092,Stacy,Olsen,25,Veterinarian\n',
'4000093,Curtis,Love,45,Secretary\n',
'4000094,Dana,McLean,61,Artist\n',
'4000095,Jennifer,Christian,54,Human resources assistant\n',
'4000096,Brett,Lamb,39,Engineering technician\n',
'4000097,Brandon,James,29,Musician\n',
'4000098,Keith,Chandler,25,Coach\n',
'4000099,Joann,Stout,32,Real estate agent\n',
'4000100,Ronnie,Cowan,71,Photographer\n',
'4000101,Scott,Golden,27,Teacher\n',
'4000102,Gene,Bowling,73,Recreation and fitness worker\n',
'4000103,Louise,Beasley,54,Loan officer\n',
'4000104,Geoffrey,Clapp,41,Photographer\n',
'4000105,Patricia,Abrams,51,Veterinarian\n',
'4000106,Jennifer,Tilley,35,Agricultural and food scientist\n',
'4000107,Mary,Morse,70,Automotive mechanic\n',
'4000108,Shawn,Boykin,34,Photographer\n',
'4000109,Vincent,Sumner,31,Lawyer\n',
'4000110,Kurt,Cassidy,32,Dancer\n',
'4000111,Danny,Davidson,70,Agricultural and food scientist\n',
'4000112,Charlene,Heath,26,Electrician\n',
```

```
'4000113,Alice,Blanchard,73,Economist\n',
'4000114,Joan,McAllister,73,Engineering technician\n',
'4000115,Betty,McKenzie,45,Computer support specialist\n',
'4000116,Danny,Byrne,51,Dancer\n',
'4000117,Peggy,Schroeder,61,Loan officer\n',
'4000118,Leslie,Griffin,37,Photographer\n',
'4000119,Marshall,Gross,42,Actor\n',
'4000120,Sara,Perkins,67,Actor\n',
'4000121,Martha,Robertson,37,Agricultural and food scientist\n',
'4000122,Jack,Palmer,52,Human resources assistant\n',
'4000123,Gayle,Brady,45,Firefighter\n',
'4000124,Benjamin,Rowe,64,Childcare worker\n',
'4000125,Roberta,Zhang,38,Statistician\n',
'4000126,Patricia,Hodge,59,Artist\n',
'4000127,Clifford,Li,63,Photographer\n',
'4000128,Joanne,Bowling,35,Musician\n',
'4000129,Martin,Justice,58,Electrician\n',
'4000130,Toni,Glass,46,Lawyer\n',
'4000131,Beth,Willis,62,Carpenter\n',
'4000132,Jessica,Hester,72,Civil engineer\n',
'4000133,Samantha,Floyd,72,Childcare worker\n',
'4000134,Jimmy,Graves,69,Nurse\n',
'4000135,Vincent,Fischer,33,Statistician\n',
'4000136,Dianne,Norman,68,Veterinarian\n',
'4000137,Rhonda,Chan,34,Pharmacist\n',
'4000138,Tamara,Hunt,25,Psychologist\n',
'4000139,Mary,Byrd,34,Environmental scientist\n',
'4000140,Sidney,Lane,54,Statistician\n',
'4000141,Jeff,Kaplan,70,Chemist\n',
'4000142,Sandra,Heller,51,Photographer\n',
'4000143,Katie,May,28,Recreation and fitness worker\n',
'4000144,Raymond,Jennings,63,Coach\n',
'4000145,Roger,Hanna,23,Musician\n',
'4000146,Natalie,Locklear,37,Politician\n',
'4000147,Kathy,Holloway,65,Pharmacist\n',
'4000148,Troy,Jones,74,Secretary\n',
'4000149,Neal,Glover,40,Real estate agent\n',
'4000150,Martin,Vick,31,Physicist\n',
"4000151,Jack,O'Donnell,59,Actor\n",
'4000152,Vincent,Goldman,22,Electrician\n',
'4000153,Beth,McKenna,40,Veterinarian\n',
'4000154,Milton,Starr,23,Carpenter\n',
'4000155,Tamara,Stone,73,Firefighter\n',
'4000156,Mitchell,McClure,68,Loan officer\n',
'4000157,Franklin,Watson,64,Coach\n',
'4000158,Leroy,Monroe,51,Computer support specialist\n',
'4000159,Glen,Abbott,29,Loan officer\n',
'4000160,Judith,Singer,73,Actor\n',
'4000161,Alice,Hall,33,Recreation and fitness worker\n',
'4000162,Bruce,Farrell,35,Librarian\n',
'4000163,Kathleen,Lucas,66,Chemist\n',
'4000164,Amy,Norman,62,Automotive mechanic\n',
'4000165,Ronnie,Atkins,67,Dancer\n',
'4000166,Martha,Monroe,31,Judge\n',
'4000167,Lynn,Robertson,45,Lawyer\n',
'4000168,Jose,Sykes,47,Writer\n',
```

```
'4000169,Robert,Reid,39,Carpenter\n',
'4000170,Pauline,Chandler,70,Economist\n',
'4000171,Stephen,Finch,30,Coach\n',
'4000172,Peggy,Hobbs,69,Musician\n',
'4000173,Donna,Adkins,72,Electrical engineer\n',
'4000174,Doris,Kinney,36,Athlete\n',
'4000175,Ben,Whitaker,35,Computer support specialist\n',
'4000176,Kristin,Alexander,21,Coach\n',
'4000177,Ryan,Conner,51,Electrical engineer\n',
'4000178,Tracey,Waters,73,Computer hardware engineer\n',
'4000179,Mark,Becker,39,Computer support specialist\n',
'4000180,Louis,Rollins,35,Economist\n',
'4000181,Janet,Love,39,Politician\n',
'4000182,Leo,Adkins,42,Economist\n',
'4000183,Constance,Black,36,Firefighter\n',
'4000184,Sarah,Fox,73,Psychologist\n',
'4000185,Gladys,Hatcher,62,Musician\n',
'4000186,Hazel,Wu,68,Therapist\n',
'4000187,Hazel,Lloyd,48,Politician\n',
'4000188,Jerome,Joyce,40,Artist\n',
'4000189,Vincent,Welch,54,Psychologist\n',
'4000190,Kim,Matthews,38,Architect\n',
'4000191,Joseph,Chappell,59,Reporter\n',
'4000192,Alison,MacDonald,26,Childcare worker\n',
'4000193,Eric,Kane,61,Childcare worker\n',
'4000194,Heather,Butler,67,Farmer\n',
'4000195,Claire,Pickett,59,Lawyer\n',
'4000196,Michele,Bowman,54,Computer software engineer\n',
'4000197,Terry,Barton,27,Recreation and fitness worker\n',
'4000198,Ken,Kennedy,35,Automotive mechanic\n',
'4000199,Judy,Branch,42,Athlete\n',
'4000200,Clyde,Thornton,70,Pharmacist\n',
'4000201,Timothy,McNeill,70,Librarian\n',
'4000202,Ted,Weinstein,23,Human resources assistant\n',
'4000203,Sandra,Middleton,28,Librarian\n',
'4000204,Billie,Moss,65,Photographer\n',
'4000205,Katharine,Lucas,67,Electrician\n',
'4000206,Lester,Rich,53,Human resources assistant\n',
'4000207,Johnny,Carlton,28,Librarian\n',
'4000208,Ashley,Brady,43,Financial analyst\n',
'4000209,Lauren,Schultz,33,Electrician\n',
'4000210,Carolyn,Nichols,49,Agricultural and food scientist\n',
'4000211,Louis,Harvey,72,Financial analyst\n',
'4000212,Charlene,Stevenson,65,Carpenter\n',
'4000213,Ashley,Houston,24,Judge\n',
'4000214,Sheryl,Dunn,71,Civil engineer\n',
'4000215,Ben,West,64,Electrician\n',
"4000216,Linda,O'Brien,46,Loan officer\n",
'4000217,Sarah,Barr,52,Therapist\n',
'4000218,Brett,Snyder,22,Doctor\n',
'4000219,Dana,Cain,63,Human resources assistant\n',
'4000220,Rebecca,Heath,74,Environmental scientist\n',
'4000221,Glenda,Boswell,28,Civil engineer\n',
'4000222,Glen,Olsen,63,Childcare worker\n',
'4000223,Don,Pittman,70,Designer\n',
'4000224,Gregory,Weiner,58,Accountant\n',
```

```
'4000225,Randy,Petersen,71,Actor\n',
'4000226,Geraldine,Davis,36,Accountant\n',
'4000227,Edna,Coleman,58,Veterinarian\n',
'4000228,Sidney,Terrell,32,Statistician\n',
'4000229,Faye,Norman,64,Pilot\n',
'4000230,Kathy,Burch,28,Pilot\n',
'4000231,Marguerite,Weiner,45,Police officer\n',
'4000232,Marvin,Parrott,67,Doctor\n',
'4000233,Alex,Henry,67,Police officer\n',
'4000234,Alexander,Gray,35,Reporter\n',
'4000235,Marsha,Chang,42,Chemist\n',
'4000236,Karl,McLean,69,Photographer\n',
'4000237,Laura,Eason,31,Loan officer\n',
'4000238,Melinda,Weeks,59,Computer software engineer\n',
'4000239,Eva,Siegel,50,Psychologist\n',
'4000240,Robert,Puckett,56,Physicist\n',
'4000241,Bill,Heath,35,Actor\n',
'4000242,Edna,Hoyle,74,Electrical engineer\n',
'4000243,Clifford,Garrett,72,Veterinarian\n',
'4000244,Penny,Neal,59,Veterinarian\n',
'4000245,Glenda,Baker,52,Economist\n',
'4000246,Arthur,Goldman,58,Childcare worker\n',
'4000247,Wesley,Shaffer,58,Farmer\n',
'4000248,Allison,Choi,23,Computer hardware engineer\n',
'4000249,Florence,Carver,34,Electrical engineer\n',
'4000250,Claire,Shelton,34,Environmental scientist\n',
'4000251,Jeremy,House,61,Pilot\n',
'4000252,Christy,Lyons,31,Physicist\n',
'4000253,Wesley,Moser,64,Doctor\n',
'4000254,Lucy,Dickinson,72,Loan officer\n',
'4000255,Susan,Abbott,65,Actor\n',
'4000256,Nancy,Hobbs,40,Environmental scientist\n',
'4000257,Monica,Dodson,58,\n',
'4000258,Justin,Spencer,67,Economist\n',
'4000259,Margaret,Burgess,74,Social worker\n',
'4000260,Philip,Liu,59,Nurse\n',
'4000261,Miriam,Wong,27,Childcare worker\n',
'4000262,Miriam,Blackburn,52,Athlete\n',
'4000263,Christopher,McKay,67,Judge\n',
'4000264,Joanna,Middleton,37,Firefighter\n',
'4000265,Stacy,Frazier,69,Designer\n',
'4000266,Janice,Reid,40,Therapist\n',
'4000267,Brian,Braswell,64,Chemist\n',
'4000268,Juan,Steele,57,Nurse\n',
'4000269,Leroy,Donovan,74,Photographer\n',
'4000270,Tommy,Barrett,38,Engineering technician\n',
'4000271,Alice,Nance,59,Pilot\n',
'4000272,Adam,Washington,54,Electrical engineer\n',
'4000273,Larry,Rogers,50,Engineering technician\n',
'4000274,Angela,McMahon,66,Secretary\n',
'4000275,Lawrence,Miles,74,Dancer\n',
'4000276,Glen,Kramer,43,Coach\n',
'4000277,Valerie,Jennings,59,Loan officer\n',
'4000278,Meredith,Bowles,62,Accountant\n',
'4000279,Jan,Brown,58,Secretary\n',
'4000280,Lillian,Bolton,32,Photographer\n',
```

```
'4000281,Allen,Craven,64,Therapist\n',
'4000282,Ashley,Hendrix,50,Accountant\n',
'4000283,Ian,Nichols,57,Electrician\n',
'4000284,Gina,Saunders,48,Photographer\n',
'4000285,Donna,Lehman,71,Social worker\n',
'4000286,Catherine,Sherrill,48,Nurse\n',
'4000287,Lester,Cash,30,Firefighter\n',
'4000288,Jacob,Pittman,34,Economist\n',
'4000289,Nancy,Sullivan,70,Electrical engineer\n',
'4000290,Gail,Whitehead,21,Nurse\n',
'4000291,Lloyd,Mack,67,Recreation and fitness worker\n',
'4000292,Donna,Rice,21,Social worker\n',
'4000293,Kay,Ayers,44,Childcare worker\n',
'4000294,Wendy,Cherry,70,Loan officer\n',
'4000295,Lucille,Richmond,70,Photographer\n',
'4000296,Virginia,York,67,Politician\n',
'4000297,Bruce,Wiley,43,Artist\n',
'4000298,Marc,Harrington,57,Physicist\n',
'4000299,Geoffrey,Reed,53,Reporter\n',
'4000300,Tiffany,Nash,41,Coach\n',
'4000301,Todd,Wilkerson,55,Librarian\n',
'4000302,Jill,Kent,27,Electrical engineer\n',
'4000303,Erin,Finch,54,Police officer\n',
'4000304,Melinda,Starr,57,Statistician\n',
'4000305,Julie,Holland,61,Architect\n',
'4000306,Stanley,Glover,44,Nurse\n',
'4000307,Karen,Clements,74,Farmer\n',
'4000308,Judy,Schultz,42,Statistician\n',
'4000309,Raymond,Hawley,66,Recreation and fitness worker\n',
'4000310,Julie,Skinner,56,Electrical engineer\n',
'4000311,Sue,Hamrick,43,Reporter\n',
'4000312,Colleen,Winters,36,Designer\n',
'4000313,Judith,Dolan,44,Civil engineer\n',
'4000314,Luis,Turner,69,Computer software engineer\n',
'4000315,Bradley,Beatty,36,Human resources assistant\n',
'4000316,Glenda,Douglas,28,Loan officer\n',
'4000317,Brent,Byrne,62,Judge\n',
'4000318,Clyde,Hendricks,41,Loan officer\n',
'4000319,Sharon,Mayer,70,Statistician\n',
'4000320,Mary,Cochran,25,Computer support specialist\n',
'4000321,Sue,Reilly,33,Physicist\n',
'4000322,Geraldine,Jensen,50,Pilot\n',
'4000323,Rita,Yates,34,Librarian\n',
'4000324,Annie,Haynes,51,Social worker\n',
'4000325,Ruth,Harmon,75,Artist\n',
'4000326,Michelle,Matthews,67,Computer software engineer\n',
'4000327,Heather,Dawson,43,\n',
'4000328,Jane,Barefoot,53,Accountant\n',
'4000329,Wallace,Kaplan,35,Secretary\n',
'4000330,Kimberly,Gross,38,Designer\n',
'4000331,Jon,Richmond,37,Agricultural and food scientist\n',
'4000332,Lee,Pope,55,Statistician\n',
'4000333,Kenneth,Pickett,28,Police officer\n',
'4000334,Keith,Schwartz,65,Childcare worker\n',
'4000335,Jim,Singleton,55,Accountant\n',
'4000336,Steven,Ballard,62,Pilot\n',
```

```
'4000337,Pauline,Spivey,55,Computer hardware engineer\n',
'4000338,Marc,Denton,71,Musician\n',
'4000339,Kerry,Huff,38,Writer\n',
'4000340,Lynne,Mangum,37,Chemist\n',
'4000341,Claude,Berger,51,Electrical engineer\n',
'4000342,Rose,McCall,29,Farmer\n',
'4000343,Kathy,Pollard,43,Politician\n',
'4000344,Catherine,Garcia,69,Athlete\n',
'4000345,Frances,Wagner,28,Veterinarian\n',
'4000346,Diana,Crane,26,Lawyer\n',
'4000347,Leslie,Wolf,50,Loan officer\n',
'4000348,Marian,Crane,37,Automotive mechanic\n',
'4000349,Russell,Dalton,21,Civil engineer\n',
'4000350,Sheryl,Diaz,30,Pharmacist\n',
'4000351,Lewis,Currin,73,Farmer\n',
'4000352,Ernest,Stanton,51,Lawyer\n',
'4000353,Vivian,Carey,64,Statistician\n',
'4000354,Harold,Li,49,Doctor\n',
'4000355,Malcolm,Chan,27,Environmental scientist\n',
'4000356,Kay,Hess,56,Childcare worker\n',
'4000357,Max,Robinson,32,Dancer\n',
'4000358,Vicki,Mills,56,Computer support specialist\n',
'4000359,Leon,Bender,22,Firefighter\n',
'4000360,Alvin,McDonald,73,Musician\n',
'4000361,Jacob,Moore,64,Environmental scientist\n',
'4000362,Holly,Fox,74,Physicist\n',
'4000363,Audrey,Lanier,54,Human resources assistant\n',
'4000364,Christina,Harris,39,Doctor\n',
'4000365,Joseph,Underwood,57,Reporter\n',
'4000366,Evelyn,Parsons,37,Human resources assistant\n',
'4000367,Nicholas,Vaughn,66,Carpenter\n',
'4000368,Ben,Banks,60,Accountant\n',
'4000369,Dorothy,Sherrill,28,Agricultural and food scientist\n',
'4000370,Theresa,Oakley,35,Nurse\n',
'4000371,Harriet,Rubin,71,Real estate agent\n',
'4000372,Barry,Maynard,53,Pharmacist\n',
'4000373,Jerome,Hill,66,Computer hardware engineer\n',
'4000374,Jason,Livingston,67,Civil engineer\n',
'4000375,Norman,Lam,63,Teacher\n',
'4000376,Glen,Thompson,40,Coach\n',
'4000377,Ellen,Creech,36,Human resources assistant\n',
'4000378,Dianne,Dillon,34,Recreation and fitness worker\n',
'4000379,Harry,Foster,50,Coach\n',
'4000380,Tim,Starr,59,Photographer\n',
'4000381,Kent,Roy,31,Electrical engineer\n',
'4000382,Eddie,Barbour,61,Athlete\n',
'4000383,Arnold,Burke,70,Reporter\n',
'4000384,Melanie,Ritchie,36,Financial analyst\n',
'4000385,Ronald,Odom,49,Psychologist\n',
'4000386,Gene,Pearce,30,Architect\n',
'4000387,Peter,Rosenberg,60,Politician\n',
'4000388,Jean,Garrett,73,Doctor\n',
"4000389,Neal,O'Connor,26,Therapist\n",
'4000390,Tracy,Cates,54,Dancer\n',
'4000391,Holly,McIntosh,73,Computer software engineer\n',
'4000392,Emma,Olson,74,Pilot\n',
```

```
'4000393,Molly,Cox,55,Social worker\n',
'4000394,Carlos,Erickson,63,Statistician\n',
'4000395,Betsy,Chang,54,Designer\n',
'4000396,Andrew,Briggs,45,Farmer\n',
'4000397,Bill,Klein,26,Teacher\n',
'4000398,Eileen,Goldberg,57,Therapist\n',
'4000399,Randall,Hinson,51,Judge\n',
'4000400,Kelly,Weiss,72,Librarian\n',
'4000401,Elaine,Pritchard,48,Automotive mechanic\n',
'4000402,Rhonda,Goldman,38,Computer support specialist\n',
'4000403,Leo,Lassiter,65,Pilot\n',
'4000404,Clarence,Massey,74,Politician\n',
'4000405,Grace,Stark,41,Pharmacist\n',
'4000406,Jordan,Dunlap,59,Agricultural and food scientist\n',
'4000407,Glenda,Humphrey,32,Carpenter\n',
'4000408,Jacob,Singleton,33,Veterinarian\n',
'4000409,Jordan,Horowitz,58,Actor\n',
'4000410,Dawn,Lutz,66,Therapist\n',
'4000411,Joanna,Hoover,50,\n',
'4000412,Sandra,Kang,47,Designer\n',
'4000413,Stephen,Melton,33,Athlete\n',
'4000414,Jessica,Teague,72,Firefighter\n',
'4000415,Regina,Ellington,52,Social worker\n',
'4000416,Paige,Cherry,45,Environmental scientist\n',
'4000417,Joyce,Jennings,61,Social worker\n',
'4000418,Tamara,Creech,46,Social worker\n',
'4000419,Toni,Lynn,67,Civil engineer\n',
'4000420,Betty,Albright,64,Engineering technician\n',
'4000421,Jerry,Alston,31,Lawyer\n',
'4000422,Tommy,Burnette,72,Doctor\n',
"4000423,Alan,O'Neal,59,Pilot\n",
'4000424,Christina,Morris,56,Dancer\n',
'4000425,Dean,Lutz,26,Pilot\n',
'4000426,Veronica,Callahan,31,Veterinarian\n',
'4000427,Kelly,Conway,36,Designer\n',
'4000428,Marcus,Harvey,62,Computer support specialist\n',
'4000429,Carrie,Watson,52,Musician\n',
'4000430,George,Glover,41,Nurse\n',
'4000431,Nina,Savage,35,Politician\n',
'4000432,Paige,Henson,61,Dancer\n',
'4000433,Jay,Wang,64,Carpenter\n',
'4000434,Sue,Ellis,59,Environmental scientist\n',
'4000435,Brenda,Barbour,36,Architect\n',
'4000436,Joshua,Sherrill,43,Politician\n',
'4000437,Nelson,Pierce,69,Actor\n',
'4000438,Vincent,Woodward,34,Electrical engineer\n',
'4000439,Nicholas,Godfrey,69,Environmental scientist\n',
'4000440,Hannah,Langston,25,Pilot\n',
'4000441,Alice,Eaton,32,Chemist\n',
'4000442,Elisabeth,Lowe,34,Farmer\n',
'4000443,Matthew,Stanton,51,Secretary\n',
'4000444,Thomas,Fuller,51,Childcare worker\n',
'4000445,Benjamin,Simmons,70,Veterinarian\n',
'4000446,Kristine,Schultz,66,Architect\n',
'4000447,Jordan,Knight,65,Electrician\n',
'4000448,Guy,Klein,34,Designer\n',
```

```
'4000449,Terry,Garcia,30,Automotive mechanic\n',
'4000450,Floyd,Schroeder,37,Farmer\n',
'4000451,Russell,Hess,21,Architect\n',
'4000452,Marsha,Gold,36,Athlete\n',
'4000453,Kim,Hensley,64,Photographer\n',
'4000454,Virginia,Turner,22,Computer hardware engineer\n',
'4000455,Deborah,French,56,Accountant\n',
'4000456,Peter,Hughes,38,Computer software engineer\n',
'4000457,Bernard,Pate,34,Doctor\n',
'4000458,Harold,Burnett,40,Musician\n',
'4000459,Gretchen,Francis,60,Politician\n',
'4000460,Cheryl,Horn,40,Veterinarian\n',
'4000461,Audrey,Forrest,50,Architect\n',
'4000462,Alan,Levin,43,Statistician\n',
'4000463,Wayne,Weiner,70,\n',
'4000464,Nancy,Durham,60,Nurse\n',
'4000465,Peter,Guthrie,21,Nurse\n',
'4000466,Benjamin,Hensley,37,Pilot\n',
'4000467,Derek,Freedman,66,Recreation and fitness worker\n',
'4000468,Marlene,Wiggins,65,Agricultural and food scientist\n',
'4000469,Clara,Best,51,Pharmacist\n',
'4000470,Ashley,Beatty,46,Computer support specialist\n',
'4000471,Kara,Crawford,40,Social worker\n',
'4000472,Kristin,Drake,22,Financial analyst\n',
'4000473,Edwin,Curtis,25,Civil engineer\n',
'4000474,Molly,Walter,30,Architect\n',
'4000475,Marshall,Dunlap,43,Statistician\n',
'4000476,Elaine,Jenkins,47,Writer\n',
'4000477,Victor,Hood,48,Firefighter\n',
'4000478,Tonya,Ellis,26,Firefighter\n',
'4000479,Cecil,Jiang,74,Photographer\n',
'4000480,Melissa,Johnson,64,Recreation and fitness worker\n',
'4000481,Christina,Craig,48,Dancer\n',
'4000482,Allison,Norman,25,Musician\n',
'4000483,Gretchen,McIntyre,38,Statistician\n',
'4000484,Bonnie,Brantley,68,Firefighter\n',
'4000485,George,Kelley,58,Judge\n',
'4000486,Barbara,Smith,58,Designer\n',
'4000487,Evelyn,Lyons,51,Childcare worker\n',
'4000488,Rita,Wall,58,Photographer\n',
'4000489,Dan,Quinn,31,Agricultural and food scientist\n',
'4000490,Juan,Hicks,37,Accountant\n',
'4000491,Jesse,Garrison,59,Pharmacist\n',
'4000492,Ronnie,Watts,29,Actor\n',
'4000493,Hilda,Dickerson,53,Farmer\n',
'4000494,Stephen,Waller,75,\n',
'4000495,Lorraine,Carter,65,Financial analyst\n',
'4000496,Chris,Robinson,74,Economist\n',
'4000497,Katie,Katz,26,Dancer\n',
'4000498,Richard,Hull,69,Accountant\n',
'4000499,Steve,Bowling,61,Recreation and fitness worker\n',
'4000500,Monica,Brantley,58,Engineering technician\n',
'4000501,Kenneth,Brock,68,Loan officer\n',
'4000502,Leigh,James,37,Teacher\n',
'4000503,Glenda,McMillan,61,Police officer\n',
'4000504,Ricky,Hu,50,Teacher\n',
```

```
'4000505,Rick,Waller,54,Teacher\n',
'4000506,Wanda,Abbott,36,Agricultural and food scientist\n',
'4000507,Heidi,McKee,68,Engineering technician\n',
'4000508,Robin,Waters,63,Computer hardware engineer\n',
'4000509,Dan,Sims,33,Loan officer\n',
'4000510,Kathy,Henderson,69,Coach\n',
'4000511,Alicia,Rao,68,Loan officer\n',
'4000512,Sherry,Bray,67,Chemist\n',
'4000513,Chris,Scarborough,45,Engineering technician\n',
'4000514,Nathan,Ford,36,Financial analyst\n',
'4000515,Sidney,Blum,54,Photographer\n',
'4000516,Clyde,Kenney,25,Pharmacist\n',
'4000517,Barry,Gordon,58,Actor\n',
'4000518,Shannon,Blair,72,Librarian\n',
'4000519,Melinda,Moore,67,Chemist\n',
'4000520,Maureen,Kemp,45,Artist\n',
'4000521,Pat,Hutchinson,42,Computer support specialist\n',
'4000522,Neal,Brennan,28,Therapist\n',
'4000523,Anne,Little,64,Nurse\n',
'4000524,Clifford,Gill,40,Computer support specialist\n',
'4000525,Nicole,Keller,55,Human resources assistant\n',
'4000526,Rachel,Rosenthal,56,Automotive mechanic\n',
'4000527,Carolyn,McConnell,31,Firefighter\n',
'4000528,Samuel,Sawyer,63,Pharmacist\n',
'4000529,Theodore,McCall,68,Lawyer\n',
'4000530,Eleanor,Coates,31,Reporter\n',
'4000531,Glenn,Hicks,40,Librarian\n',
'4000532,Wesley,Davidson,73,Electrical engineer\n',
'4000533,Ralph,Hawkins,34,Agricultural and food scientist\n',
'4000534,Annie,Lindsay,38,Statistician\n',
'4000535,Beth,Gonzalez,41,Human resources assistant\n',
'4000536,Claire,Gray,66,Therapist\n',
'4000537,Jonathan,English,67,Agricultural and food scientist\n',
'4000538,Bradley,Duke,27,Police officer\n',
'4000539,Hugh,Webb,71,Firefighter\n',
'4000540,Nicholas,Baldwin,71,Childcare worker\n',
'4000541,Penny,Lamb,32,Agricultural and food scientist\n',
'4000542,Gloria,Shaffer,33,Pilot\n',
'4000543,Kelly,Wang,31,Accountant\n',
'4000544,Tommy,Burgess,38,Librarian\n',
'4000545,Danielle,Smith,22,Pharmacist\n',
'4000546,Calvin,Fletcher,55,Judge\n',
'4000547,Emily,Boyd,61,Psychologist\n',
'4000548,Luis,Hirsch,72,Librarian\n',
'4000549,Kimberly,Currie,74,Financial analyst\n',
'4000550,Paige,McKenzie,36,Chemist\n',
'4000551,Sidney,Weber,26,Electrician\n',
'4000552,Robert,Honeycutt,63,Computer hardware engineer\n',
'4000553,Karen,Manning,30,Recreation and fitness worker\n',
'4000554,Kyle,Bolton,25,Writer\n',
'4000555,Mary,Ritchie,57,Politician\n',
'4000556,Evelyn,Baldwin,71,Statistician\n',
'4000557,Lynda,Riley,38,Electrician\n',
'4000558,Beth,Swanson,64,Photographer\n',
'4000559,Erin,Huffman,37,Electrician\n',
'4000560,James,Gibson,35,Computer support specialist\n',
```

```
'4000561,Leigh,Yates,45,Secretary\n',
'4000562,Jordan,Wrenn,61,Pilot\n',
'4000563,Willie,Green,34,Civil engineer\n',
'4000564,Wade,Harris,63,Pharmacist\n',
'4000565,Lori,Hayes,51,Firefighter\n',
'4000566,Charlotte,Hamrick,31,Therapist\n',
'4000567,Rhonda,Hawley,37,Reporter\n',
'4000568,Steve,Koch,36,Nurse\n',
'4000569,Nicholas,McKenzie,33,Actor\n',
'4000570,Vicki,Harrell,74,Civil engineer\n',
'4000571,Stephanie,Parsons,29,Politician\n',
'4000572,Jose,McGuire,61,Photographer\n',
'4000573,George,Stephenson,42,Electrician\n',
'4000574,Marlene,Baxter,37,Recreation and fitness worker\n',
'4000575,Catherine,Summers,27,Real estate agent\n',
'4000576,Carrie,Welch,67,Doctor\n',
'4000577,Leo,Nixon,46,Coach\n',
'4000578,Rita,Kelly,32,Automotive mechanic\n',
'4000579,Joan,Sumner,28,Accountant\n',
'4000580,Marsha,Cobb,56,Chemist\n',
'4000581,Wesley,Bruce,39,Civil engineer\n',
'4000582,Cathy,Newton,37,Teacher\n',
'4000583,Kevin,Rogers,66,Lawyer\n',
'4000584,Craig,Sanchez,60,Chemist\n',
'4000585,Lester,Finch,65,Human resources assistant\n',
'4000586,Joy,Silverman,28,Coach\n',
'4000587,Kate,Horn,54,Doctor\n',
'4000588,Bernice,Richardson,71,Real estate agent\n',
'4000589,Theodore,Gay,60,Accountant\n',
'4000590,Cheryl,Chase,53,Secretary\n',
'4000591,Jamie,Gallagher,69,Librarian\n',
'4000592,Audrey,Kern,50,Nurse\n',
'4000593,Jennifer,Scott,43,Carpenter\n',
'4000594,Lewis,Bradley,31,Carpenter\n',
'4000595,Ben,Puckett,25,Photographer\n',
'4000596,Allen,Sanchez,59,Recreation and fitness worker\n',
'4000597,Alex,Yang,62,Athlete\n',
'4000598,Jennifer,Brantley,47,Musician\n',
'4000599,Doris,Bunn,43,Veterinarian\n',
'4000600,Jonathan,Link,44,Financial analyst\n',
'4000601,Allan,Nguyen,51,Pilot\n',
'4000602,Shawn,Stephens,32,Veterinarian\n',
'4000603,Stephanie,Horne,60,Real estate agent\n',
'4000604,Carol,Burton,39,Doctor\n',
'4000605,Melanie,Diaz,56,Musician\n',
'4000606,Samantha,Berry,31,Pilot\n',
'4000607,Renee,Knowles,63,Judge\n',
'4000608,Cathy,Freeman,42,Computer hardware engineer\n',
'4000609,Heidi,Hernandez,71,Nurse\n',
'4000610,Gayle,Roach,50,Athlete\n',
'4000611,Rita,Hardison,29,Athlete\n',
'4000612,Jacob,Wolf,54,Statistician\n',
'4000613,Steven,Boyd,71,Physicist\n',
'4000614,Tracy,Caldwell,61,Electrician\n',
'4000615,Eva,Mann,30,Police officer\n',
'4000616,Lorraine,McLeod,43,Physicist\n',
```

```
'4000617,Willie,Stanton,63,Musician\n',
'4000618,Joel,Park,33,Engineering technician\n',
'4000619,Vanessa,Chang,63,Psychologist\n',
'4000620,Becky,Newton,42,Judge\n',
'4000621,Melinda,Phillips,28,Civil engineer\n',
'4000622,Harvey,Whitaker,44,Farmer\n',
'4000623,Lee,Pitts,40,Photographer\n',
'4000624,Aaron,McLean,35,Social worker\n',
'4000625,Jack,Barton,30,Veterinarian\n',
'4000626,Suzanne,Gould,67,Loan officer\n',
'4000627,Katherine,Atkins,69,Recreation and fitness worker\n',
'4000628,Robyn,Shapiro,58,Architect\n',
'4000629,Julia,Vincent,49,Therapist\n',
'4000630,Bob,Harrell,40,Electrician\n',
'4000631,Brooke,Boswell,66,Childcare worker\n',
'4000632,Janet,Lassiter,58,Financial analyst\n',
'4000633,Kristen,Fisher,59,Recreation and fitness worker\n',
'4000634,Jenny,Case,64,Agricultural and food scientist\n',
'4000635,Jeanette,Parsons,63,Architect\n',
'4000636,Megan,McPherson,46,Athlete\n',
'4000637,Joel,Wiley,42,Architect\n',
'4000638,Don,Schwartz,35,Coach\n',
'4000639,Florence,McFarland,23,Electrical engineer\n',
'4000640,Jon,Baker,53,Civil engineer\n',
'4000641,Annie,Holden,36,Childcare worker\n',
'4000642,Kent,Hartman,67,Photographer\n',
'4000643,Eleanor,Schwartz,56,Computer support specialist\n',
'4000644,Nina,Nguyen,62,Doctor\n',
'4000645,Clara,Houston,53,Statistician\n',
'4000646,Bonnie,Friedman,57,Computer hardware engineer\n',
'4000647,Dwight,Adcock,67,Recreation and fitness worker\n',
'4000648,Angela,Stephens,53,Secretary\n',
'4000649,Marion,McClure,35,Coach\n',
'4000650,Curtis,Proctor,63,Computer hardware engineer\n',
'4000651,Tammy,Lang,63,Musician\n',
'4000652,Gilbert,Berger,32,Athlete\n',
'4000653,Edwin,Aldridge,28,Pilot\n',
'4000654,Emily,Davies,34,Coach\n',
'4000655,Hannah,Wall,23,Librarian\n',
'4000656,Ronald,Miles,64,Dancer\n',
'4000657,Helen,Bolton,62,Librarian\n',
'4000658,Barbara,Morgan,53,Childcare worker\n',
'4000659,Milton,Fisher,29,Dancer\n',
'4000660,Anne,Stephens,63,Carpenter\n',
'4000661,Lois,Holmes,31,Photographer\n',
'4000662,Dorothy,Ferrell,53,Firefighter\n',
'4000663,Claire,Henry,29,Police officer\n',
'4000664,Anita,Hedrick,63,Computer software engineer\n',
'4000665,Ronald,Horne,74,Architect\n',
'4000666,Allen,Weiss,38,Computer hardware engineer\n',
'4000667,Zachary,Singh,66,Firefighter\n',
'4000668,Kyle,Blalock,70,Politician\n',
'4000669,Seth,Aldridge,51,Judge\n',
'4000670,Jeff,Ritchie,65,Photographer\n',
'4000671,Sandra,Grossman,50,Coach\n',
'4000672,Jackie,Pugh,39,Real estate agent\n',
```

```
'4000673,Audrey,Olson,36,Computer support specialist\n',
'4000674,Dana,Fernandez,39,Recreation and fitness worker\n',
'4000675,Hazel,Arnold,69,Veterinarian\n',
'4000676,Joann,Stanley,47,Computer hardware engineer\n',
'4000677,Audrey,Field,70,Carpenter\n',
'4000678,Walter,Farmer,50,Human resources assistant\n',
'4000679,Herbert,Jernigan,59,\n',
'4000680,Harold,Bowers,57,Librarian\n',
'4000681,Sherry,Crabtree,55,Actor\n',
'4000682,Jeanne,Crabtree,24,Pharmacist\n',
'4000683,Rick,Clements,61,Nurse\n',
'4000684,Kristen,Spivey,59,Chemist\n',
'4000685,Patrick,Archer,27,Veterinarian\n',
'4000686,Sherri,Owen,66,Childcare worker\n',
'4000687,Herbert,Strickland,74,Accountant\n',
'4000688,Miriam,Berg,25,Judge\n',
'4000689,Maurice,Gibbons,60,Coach\n',
'4000690,Sandy,Warner,55,Judge\n',
'4000691,Shannon,Bray,69,Designer\n',
'4000692,Bradley,Eason,39,Doctor\n',
'4000693,Lester,Hoover,67,Computer hardware engineer\n',
'4000694,Tom,Park,54,Doctor\n',
'4000695,Chris,Anderson,52,\n',
'4000696,Milton,Li,33,Doctor\n',
'4000697,Eugene,Elmore,31,Dancer\n',
'4000698,Ashley,Pearson,52,Judge\n',
'4000699,Lois,Harper,36,Politician\n',
'4000700,Annette,Chu,39,Computer support specialist\n',
'4000701,Jenny,Schultz,56,Computer support specialist\n',
'4000702,Zachary,Black,26,Childcare worker\n',
'4000703,Barry,Mitchell,66,Pharmacist\n',
'4000704,Jerome,Sharp,56,Athlete\n',
'4000705,Cynthia,Glover,69,Physicist\n',
'4000706,Marilyn,Cates,32,Automotive mechanic\n',
'4000707,Natalie,Martin,52,Physicist\n',
'4000708,Brenda,Lowry,57,Statistician\n',
'4000709,Marsha,Cooke,28,Photographer\n',
'4000710,William,Fink,44,Veterinarian\n',
'4000711,Gloria,Barrett,36,Farmer\n',
'4000712,Lester,Olson,62,Lawyer\n',
'4000713,Vanessa,Melton,40,Veterinarian\n',
'4000714,Pauline,Coley,74,Farmer\n',
'4000715,Hannah,Mueller,26,Artist\n',
'4000716,Warren,Paul,32,Politician\n',
'4000717,Teresa,Daniel,26,Statistician\n',
'4000718,Joan,Padgett,39,Agricultural and food scientist\n',
'4000719,Jonathan,Daniels,28,Writer\n',
'4000720,Denise,Hayes,34,Psychologist\n',
'4000721,Bobby,Hines,26,Pilot\n',
'4000722,Erin,Pridgen,44,Carpenter\n',
'4000723,Harry,Stone,64,Computer support specialist\n',
'4000724,Priscilla,Hayes,28,Computer support specialist\n',
'4000725,Renee,Harris,65,Architect\n',
'4000726,Bob,Walter,72,Statistician\n',
'4000727,Julian,Woods,40,Financial analyst\n',
'4000728,Annie,Jennings,35,Nurse\n',
```

```
'4000729,Wendy,Lopez,37,Politician\n',
'4000730,Bernard,McCarthy,41,Electrician\n',
'4000731,Alvin,Frederick,27,Recreation and fitness worker\n',
'4000732,Denise,Lopez,60,Veterinarian\n',
'4000733,Gary,Scarborough,27,Firefighter\n',
'4000734,Oscar,Brandt,65,Firefighter\n',
'4000735,Nina,Nolan,70,Librarian\n',
'4000736,Ann,Chandler,25,Electrician\n',
'4000737,Shirley,Carlton,57,Librarian\n',
'4000738,Joanne,Katz,60,Recreation and fitness worker\n',
'4000739,Patsy,Parrott,41,Therapist\n',
'4000740,Franklin,Corbett,75,Firefighter\n',
'4000741,Bradley,Godfrey,34,Photographer\n',
'4000742,Marguerite,Cooke,56,Computer support specialist\n',
'4000743,Sherri,Pate,61,Accountant\n',
'4000744,Dorothy,Barber,39,Firefighter\n',
'4000745,Brent,Fletcher,68,Photographer\n',
'4000746,Regina,Schroeder,34,Environmental scientist\n',
'4000747,Calvin,Lindsay,35,Firefighter\n',
'4000748,Gilbert,Boswell,37,Financial analyst\n',
'4000749,Wesley,Buckley,60,Nurse\n',
'4000750,Erika,Harmon,53,Dancer\n',
'4000751,Don,Walters,70,Electrical engineer\n',
'4000752,Ethel,Stevens,23,Designer\n',
'4000753,Rodney,Knight,29,Carpenter\n',
'4000754,Dennis,Rowland,72,Coach\n',
'4000755,Zachary,Lindsay,67,Agricultural and food scientist\n',
'4000756,Harriet,Bowling,40,Pharmacist\n',
'4000757,Amanda,Kirby,24,Electrician\n',
'4000758,Jose,Benson,65,Computer hardware engineer\n',
'4000759,Lynn,Anthony,43,Social worker\n',
'4000760,Shelley,Dunn,27,Chemist\n',
'4000761,Mike,Hill,51,Chemist\n',
'4000762,Laurence,Lang,58,Teacher\n',
'4000763,Amanda,Grimes,31,Physicist\n',
'4000764,Bernard,Bowers,38,Loan officer\n',
'4000765,Janice,Bowden,21,Politician\n',
'4000766,Hazel,Underwood,71,Agricultural and food scientist\n',
'4000767,Luis,Zhang,24,Pharmacist\n',
'4000768,Ralph,Godwin,54,Judge\n',
'4000769,Geraldine,Rice,53,Judge\n',
'4000770,Claire,Townsend,57,Financial analyst\n',
'4000771,Jeanne,Lin,70,Computer hardware engineer\n',
'4000772,Willie,Pitts,29,Actor\n',
'4000773,Alfred,Koch,62,Judge\n',
'4000774,Eddie,Callahan,43,Pilot\n',
'4000775,Christopher,Long,34,Environmental scientist\n',
'4000776,Phyllis,Norton,28,Economist\n',
'4000777,Peggy,Blackburn,57,Police officer\n',
"4000778,Elaine,O'Connell,70,Agricultural and food scientist\n",
'4000779,Jason,Bowling,21,Coach\n',
'4000780,Neal,Robinson,75,Artist\n',
'4000781,Diana,Pritchard,73,Therapist\n',
'4000782,Nina,Lawson,51,Loan officer\n',
'4000783,Maxine,Dickerson,57,Photographer\n',
'4000784,Ted,Livingston,70,Economist\n',
```

```
'4000785,Allen,Hansen,33,Secretary\n',
'4000786,Gordon,Berman,41,Teacher\n',
'4000787,Billy,Carroll,27,Librarian\n',
'4000788,Veronica,Kearney,38,Photographer\n',
'4000789,Hazel,Peterson,60,Designer\n',
'4000790,Eddie,Richards,40,Chemist\n',
'4000791,Katherine,Sutherland,25,Dancer\n',
'4000792,William,McCormick,64,Politician\n',
'4000793,Nina,Beach,48,Civil engineer\n',
'4000794,Ben,Wu,51,Environmental scientist\n',
'4000795,Frances,Hunt,33,Artist\n',
'4000796,Thelma,Carver,71,Teacher\n',
'4000797,Dennis,Anthony,56,Musician\n',
'4000798,Christina,Livingston,48,Police officer\n',
'4000799,James,Floyd,51,Accountant\n',
'4000800,Frederick,McCall,67,Veterinarian\n',
'4000801,Alfred,Haynes,36,Farmer\n',
'4000802,Oscar,Gunter,36,Musician\n',
'4000803,Joanne,Solomon,48,Doctor\n',
'4000804,Richard,Harris,27,Actor\n',
'4000805,Christine,Cline,36,Police officer\n',
'4000806,Theodore,McKay,33,Accountant\n',
'4000807,Jack,Braun,68,Financial analyst\n',
'4000808,Rhonda,Preston,49,Actor\n',
'4000809,Beth,Hayes,59,Designer\n',
'4000810,Douglas,Burnette,41,Secretary\n',
'4000811,Joe,Finch,45,Dancer\n',
'4000812,Sue,Levine,37,Farmer\n',
'4000813,Dennis,Lynch,66,Teacher\n',
'4000814,Sandra,Simpson,28,Social worker\n',
'4000815,Julie,Galloway,53,Actor\n',
'4000816,Henry,Dickson,51,Electrical engineer\n',
'4000817,Vernon,Murphy,57,Firefighter\n',
'4000818,Lloyd,Cannon,30,Nurse\n',
'4000819,Chris,Fleming,43,Social worker\n',
'4000820,George,Hanson,32,Pharmacist\n',
'4000821,Warren,Blackwell,72,Computer software engineer\n',
'4000822,Connie,Zimmerman,45,Computer hardware engineer\n',
'4000823,Theresa,Dyer,60,Veterinarian\n',
'4000824,Harvey,Greenberg,65,Economist\n',
'4000825,Janet,Quinn,49,Secretary\n',
'4000826,Charlene,Sullivan,72,Librarian\n',
'4000827,Gayle,Stanley,27,Coach\n',
'4000828,Julian,Hendrix,66,Electrical engineer\n',
'4000829,Paula,Barber,70,Nurse\n',
'4000830,Geraldine,High,30,Human resources assistant\n',
'4000831,Constance,Pickett,57,Electrical engineer\n',
'4000832,Brian,Copeland,72,Loan officer\n',
'4000833,Vivian,Beck,74,Recreation and fitness worker\n',
'4000834,Kim,McKenna,65,Childcare worker\n',
'4000835,Sarah,King,39,Reporter\n',
'4000836,Holly,Stone,33,Coach\n',
'4000837,Brett,Benton,59,Librarian\n',
'4000838,Frances,Boyette,37,Chemist\n',
'4000839,Phyllis,Byers,25,Environmental scientist\n',
'4000840,Gwendolyn,Cook,74,Firefighter\n',
```

```
'4000841,Carl,Nixon,42,Athlete\n',
'4000842,Marianne,Mayo,25,Coach\n',
'4000843,Gayle,Hardison,71,Coach\n',
'4000844,Jim,Marks,39,Firefighter\n',
'4000845,Lynda,Ball,26,Economist\n',
'4000846,Thomas,Kirk,53,Politician\n',
'4000847,Natalie,Cooke,66,Police officer\n',
'4000848,Brooke,Sutton,54,Psychologist\n',
'4000849,Darlene,Gibson,64,Childcare worker\n',
'4000850,Lloyd,Haynes,34,Civil engineer\n',
'4000851,Sherri,Klein,69,Nurse\n',
'4000852,Ruth,Tyson,46,Librarian\n',
'4000853,Vicki,Payne,73,Judge\n',
'4000854,Nelson,Francis,31,Computer hardware engineer\n',
'4000855,Courtney,Roth,59,Automotive mechanic\n',
'4000856,Claude,Nixon,71,Firefighter\n',
'4000857,Louise,Coble,38,Lawyer\n',
'4000858,Vivian,Walters,62,Psychologist\n',
'4000859,Marcia,Hewitt,63,Computer software engineer\n',
'4000860,Amy,Langley,30,Artist\n',
'4000861,Chris,Scott,58,Electrician\n',
'4000862,Edgar,Willis,36,Politician\n',
'4000863,Willie,Denton,73,Secretary\n',
'4000864,Renee,Daly,52,Childcare worker\n',
'4000865,Robyn,Lam,49,Lawyer\n',
'4000866,Patrick,Fox,52,Childcare worker\n',
'4000867,Wade,Franklin,30,Electrician\n',
'4000868,Courtney,McIntosh,72,Real estate agent\n',
'4000869,Bonnie,Tyler,30,Farmer\n',
'4000870,Douglas,Hanna,63,Judge\n',
'4000871,Neal,Davenport,62,Photographer\n',
'4000872,Scott,Barton,33,Photographer\n',
'4000873,Justin,Chambers,52,Computer support specialist\n',
'4000874,Debra,Thomas,74,Veterinarian\n',
'4000875,Norman,Arthur,26,Photographer\n',
'4000876,Audrey,Law,23,Physicist\n',
'4000877,Franklin,Coley,59,Electrician\n',
'4000878,Chris,Vaughn,36,Accountant\n',
'4000879,Bobby,Case,38,Artist\n',
'4000880,Karen,Reed,71,Therapist\n',
'4000881,Carl,Hardy,32,Recreation and fitness worker\n',
'4000882,Yvonne,Beatty,43,Real estate agent\n',
'4000883,Henry,Dale,71,Veterinarian\n',
'4000884,Danielle,Russell,54,Pharmacist\n',
'4000885,Allison,Whitley,74,Carpenter\n',
'4000886,Guy,Curry,48,Lawyer\n',
'4000887,Caroline,McNeill,33,Accountant\n',
'4000888,Tammy,Franklin,58,Dancer\n',
'4000889,Caroline,Lindsay,43,Loan officer\n',
'4000890,Hannah,Casey,30,\n',
'4000891,Sharon,Meadows,54,Teacher\n',
'4000892,Ricky,Casey,64,Athlete\n',
'4000893,Gene,Love,59,Pilot\n',
'4000894,Frances,Fitzpatrick,53,Athlete\n',
'4000895,Sean,Mann,44,Nurse\n',
'4000896,Florence,Knowles,66,Secretary\n',
```

```
'4000897,Thelma,Hale,22,Real estate agent\n',
'4000898,Christian,Carlson,70,Athlete\n',
'4000899,Jordan,Barefoot,66,Environmental scientist\n',
'4000900,Jessica,Warren,61,Real estate agent\n',
'4000901,Lynne,Nelson,66,Actor\n',
'4000902,Anita,Lancaster,49,Human resources assistant\n',
'4000903,Jill,Kay,59,Reporter\n',
'4000904,Nathan,Burgess,47,Librarian\n',
'4000905,Brenda,Fitzpatrick,60,Photographer\n',
'4000906,Kathleen,Davies,40,Real estate agent\n',
'4000907,Randall,Moran,46,Musician\n',
'4000908,Carole,Ashley,59,Veterinarian\n',
'4000909,Penny,Caldwell,67,Musician\n',
'4000910,Megan,Kelley,69,Childcare worker\n',
'4000911,Cynthia,Mack,64,Athlete\n',
'4000912,Maria,Reilly,64,Pilot\n',
'4000913,Lucy,Copeland,36,Police officer\n',
'4000914,Sheryl,Love,64,Human resources assistant\n',
'4000915,Lucille,Conrad,64,Therapist\n',
'4000916,Warren,Padgett,55,Pharmacist\n',
'4000917,Gary,Poole,35,Artist\n',
'4000918,Leah,McKinney,64,Reporter\n',
'4000919,Lewis,Sawyer,31,Agricultural and food scientist\n',
'4000920,Guy,Dalton,69,Computer hardware engineer\n',
'4000921,Maria,Carey,37,Pharmacist\n',
'4000922,Maria,Stuart,35,Pilot\n',
'4000923,Maureen,Bowles,37,Social worker\n',
'4000924,Gina,Singleton,32,Environmental scientist\n',
'4000925,Todd,Britt,62,Environmental scientist\n',
'4000926,Katherine,Owens,46,Electrician\n',
'4000927,Lillian,Davenport,44,Teacher\n',
'4000928,Jennifer,Cox,36,Police officer\n',
'4000929,Jeff,Barton,49,Pharmacist\n',
'4000930,Jerome,Cooke,60,Architect\n',
'4000931,Greg,Tilley,58,Computer support specialist\n',
'4000932,Stacy,Pugh,57,Human resources assistant\n',
'4000933,Joel,Schultz,72,Financial analyst\n',
'4000934,Shannon,Connor,29,Reporter\n',
'4000935,Oscar,Herbert,35,Environmental scientist\n',
'4000936,Tina,Aycock,71,Physicist\n',
'4000937,Randy,Barry,60,Physicist\n',
'4000938,Claire,Bishop,22,Lawyer\n',
'4000939,Tony,Garrett,61,Firefighter\n',
'4000940,Sidney,Bailey,45,Agricultural and food scientist\n',
'4000941,Tracy,Riddle,60,Designer\n',
'4000942,Anna,Sawyer,73,Librarian\n',
'4000943,Luis,Burnett,72,Firefighter\n',
'4000944,Mary,Boyette,59,\n',
'4000945,Alfred,McKenzie,30,Doctor\n',
'4000946,Dianne,Sinclair,59,Electrician\n',
'4000947,Keith,Cannon,36,Politician\n',
'4000948,Leah,Freeman,59,Actor\n',
'4000949,Marc,Wallace,65,Musician\n',
'4000950,Vickie,Gilbert,29,Physicist\n',
'4000951,Ross,McNamara,42,Librarian\n',
'4000952,Roy,Mullen,51,Recreation and fitness worker\n',
```

```
      '4000953,Judy,Bradshaw,28,Psychologist\n',
      '4000954,Kimberly,Hinson,56,Engineering technician\n',
      '4000955,Helen,Jordan,34,Recreation and fitness worker\n',
      '4000956,Greg,Berger,27,Physicist\n',
      '4000957,Tracey,Upchurch,35,Environmental scientist\n',
      '4000958,Danny,Bowers,72,Pilot\n',
      '4000959,Christy,Allison,69,Dancer\n',
      '4000960,Dorothy,Alexander,67,Lawyer\n',
      '4000961,Charles,Coley,72,Designer\n',
      '4000962,Malcolm,Riley,63,Financial analyst\n',
      "4000963,Emily,O'Brien,56,Childcare worker\n",
      '4000964,Randall,Vaughan,30,Doctor\n',
      '4000965,Stacey,Hartman,62,Designer\n',
      '4000966,Gene,Chung,71,Economist\n',
      '4000967,Bob,Fischer,71,Automotive mechanic\n',
      '4000968,Regina,Sellers,42,Politician\n',
      '4000969,Jeanette,Montgomery,26,Reporter\n',
      '4000970,Judith,Snow,29,Politician\n',
      '4000971,Megan,McKnight,32,Physicist\n',
      '4000972,Betsy,McMahon,60,Designer\n',
      '4000973,Chris,Chu,26,Judge\n',
      '4000974,Cheryl,Crews,72,Accountant\n',
      '4000975,Vickie,Sharma,53,Statistician\n',
      '4000976,Marion,Puckett,32,Real estate agent\n',
      '4000977,Katharine,Pappas,73,Doctor\n',
      '4000978,Neil,Sharpe,74,Police officer\n',
      '4000979,Eileen,Olson,51,Judge\n',
      '4000980,Lester,Desai,34,Librarian\n',
      '4000981,Suzanne,Bowden,71,Musician\n',
      '4000982,Teresa,Hardy,41,Physicist\n',
      '4000983,Edgar,Stuart,44,Pilot\n',
      '4000984,Ann,Branch,75,Computer hardware engineer\n',
      '4000985,Anna,Harrell,41,Secretary\n',
      '4000986,Jeanette,Bowman,54,Firefighter\n',
      '4000987,Christina,Hubbard,25,Financial analyst\n',
      '4000988,Marcus,Nash,55,Nurse\n',
      '4000989,Tara,Wolfe,60,Artist\n',
      '4000990,Guy,Warren,40,Loan officer\n',
      '4000991,Albert,Brock,41,Actor\n',
      '4000992,Mark,Clarke,47,Photographer\n',
      '4000993,Dean,Leach,29,Pharmacist\n',
      '4000994,Tracy,Raynor,59,Electrical engineer\n',
      '4000995,Courtney,Hardy,33,Electrical engineer\n',
      '4000996,Martin,Stephens,29,Secretary\n',
      '4000997,Martin,Hester,31,Lawyer\n',
      '4000998,Marian,Murphy,32,Physicist\n',
      '4000999,Elsie,Capps,74,Human resources assistant\n',
      '4001000,Kim,McCormick,23,Lawyer\n',
      ...]
```

In [4]: `len(data)`

Out[4]: 9999

Ex. Write a function to clean the data

```
In [8]:  def clean_data(strg) :
             lst = strg.strip().split(",")
             lst[3] = int(lst[3])
             return lst
```

Ex. Apply the function to all the rows of data

```
In [10]:  customers = list(map(clean_data, data))  # List of lists
          customers[:3]
```

```
Out[10]:  [['4000001', 'Kristina', 'Chung', 55, 'Pilot'],
           ['4000002', 'Paige', 'Chen', 74, 'Teacher'],
           ['4000003', 'Sherri', 'Melton', 34, 'Firefighter']]
```

Ex. Extract data for all Pilots

```
In [13]:  pilots = list(filter(lambda cust : cust[4] == "Pilot", customers))
          len(pilots)
```

```
Out[13]:  209
```

Ex. Extract all senor citizens in the data

```
In [15]:  seniors = list(filter(lambda cust : cust[3] > 60, customers))
          len(seniors)
```

```
Out[15]:  2840
```

Object Oriented Programming Features -

1. Classe and Object
2. Constructor - **init**(self)
3. self - represent current instance
4. instance variables - belongs to each object
5. methods - algorithms to work on the instance
6. overriding built-in methods - **str**(), **repr**(), **lt**()

Ex. Create a list of customer objects

```
In [57]:  class Customer :  # Class definition
              def __init__(self, cid, fname, lname, age, prof):  # Constructor of class, para
                  self.cid = cid  # instance variable
                  self.name = fname + " " + lname
                  self.age = int(age)
                  self.profession = prof

              def age_groups(self) :
                  if self.age <35 :
                      return "Group A"
                  elif self.age < 60 :
                      return "Group B"
                  else:
```

```
            return "Group C"

    def __repr__(self):   # will be called when print the data directly
        return f"{self.name}({self.age})"

    def __str__(self):   # print(), str() will trigger __str__()
        return self.cid

    def __lt__(self, obj) :
        return self.age < obj.age

def clean_data(strg) :
    lst = strg.strip().split(",")
    cust = Customer(*lst)
    return cust

customers = list(map(clean_data, data))   # List of lists
customers[:3]
```

Out[57]: [Kristina Chung(55), Paige Chen(74), Sherri Melton(34)]

In [58]:
```
pilots = list(filter(lambda cust : cust.profession == "Pilot", customers))
len(pilots)
```

Out[58]: 209

In [59]:
```
seniors = list(filter(lambda cust : cust.age > 60, customers))
len(seniors)
```

Out[59]: 2840

In [60]: `customers[4].age_groups()`

Out[60]: 'Group C'

In [61]: `customers[4].age`

Out[61]: 74

In [62]: `customers[4]   # triggers __repr__()`

Out[62]: Karen Puckett(74)

In [48]: `str(customers[4])   # triggers __str__()`

Out[48]: '4000005'

Ex. Sort the customers by their age

In [54]: `sorted(customers, key = lambda cust : cust.age)`

```
Out[54]:  [Shirley Merritt(21),
           Kristin Alexander(21),
           Gail Whitehead(21),
           Donna Rice(21),
           Russell Dalton(21),
           Russell Hess(21),
           Peter Guthrie(21),
           Janice Bowden(21),
           Jason Bowling(21),
           Ian Hodges(21),
           Toni Byers(21),
           Edwin Ball(21),
           Brandon Nance(21),
           Lindsay Reynolds(21),
           Paige Wolfe(21),
           Nina Humphrey(21),
           Derek Foster(21),
           Penny Lu(21),
           Vanessa Franklin(21),
           Carole Rogers(21),
           Kristine Brown(21),
           Gilbert Ball(21),
           Charlene Snow(21),
           Stephanie Levine(21),
           Edwin Lim(21),
           Erin Marsh(21),
           Barry Bowles(21),
           Russell Huang(21),
           Valerie Smith(21),
           Paula Jiang(21),
           Stacey Stroud(21),
           Greg Michael(21),
           Bruce Werner(21),
           Yvonne Andrews(21),
           Stanley Frost(21),
           Derek Dudley(21),
           Henry Bond(21),
           Carol Weaver(21),
           Tommy Parrott(21),
           Bernard Kerr(21),
           Jonathan Wolf(21),
           Tara Hess(21),
           Terry Palmer(21),
           Annette Lane(21),
           Leigh Doyle(21),
           Michele Vick(21),
           Elisabeth Bowers(21),
           Maureen Oliver(21),
           Steven Chu(21),
           Kevin Blackburn(21),
           Jay Stephenson(21),
           Edgar Jackson(21),
           Emma Paul(21),
           Brett Matthews(21),
           Maureen Fischer(21),
           Lois Dyer(21),
```

Vanessa Yu(21),
Paula Lang(21),
Marjorie Poe(21),
Toni Maxwell(21),
Glenda Manning(21),
Beth Oakley(21),
Pam Adcock(21),
Grace Parrish(21),
Brent Strickland(21),
Seth Roach(21),
Frank Hendrix(21),
Dan Carson(21),
Valerie Kent(21),
Laurence Stevens(21),
Howard Abrams(21),
Cecil Lynn(21),
Katie Connor(21),
Rebecca Abrams(21),
Brooke Harvey(21),
Vicki Randall(21),
Eddie Potter(21),
Chris Barbee(21),
Yvonne Burnett(21),
Dana McKay(21),
Jenny Stokes(21),
Vernon Shapiro(21),
Stacy Barton(21),
Bonnie Stephens(21),
Holly Koch(21),
Wanda Byrne(21),
Geraldine Yu(21),
Edith Siegel(21),
Sandy Jacobs(21),
Peggy Russell(21),
Jerry Simon(21),
Megan Hamrick(21),
Joshua Daniel(21),
Robyn Lyon(21),
Edwin Williamson(21),
John Eason(21),
Sarah Cochran(21),
Debra Park(21),
Nicole Church(21),
Charles Porter(21),
Kyle Sigmon(21),
Priscilla Freeman(21),
Leo Rankin(21),
Jennifer Kirk(21),
Lorraine Beck(21),
Jay Becker(21),
Leigh Edwards(21),
Gordon House(21),
Karl Adams(21),
Herbert Jones(21),
Jill Barnes(21),
Emma Ball(21),

Geraldine Mitchell(21),
Stacey Rouse(21),
Judith Cooper(22),
Vincent Goldman(22),
Brett Snyder(22),
Leon Bender(22),
Virginia Turner(22),
Kristin Drake(22),
Danielle Smith(22),
Thelma Hale(22),
Claire Bishop(22),
Holly Hardy(22),
Faye Moon(22),
Heather Levy(22),
Eugene Shah(22),
Rodney Cook(22),
Allison Adams(22),
Dorothy Bell(22),
Larry Lamm(22),
Juan Andrews(22),
Tracy Gillespie(22),
Peggy Kramer(22),
Rosemary Boyle(22),
Johnny Gibbons(22),
Linda Dolan(22),
Julia Case(22),
Cecil Conner(22),
Natalie Silver(22),
Frank Marcus(22),
Marlene Davenport(22),
Tara McLaughlin(22),
Michelle Stern(22),
Thelma Sherman(22),
Anna Bruce(22),
Annie Pugh(22),
Andrew McAllister(22),
Frederick Case(22),
Douglas McNeill(22),
Kathleen Chen(22),
Christine Klein(22),
Cynthia Barr(22),
Gail Andrews(22),
Theodore Cole(22),
Phyllis Cobb(22),
Claire Craven(22),
Kristin George(22),
Florence Chambers(22),
Alex Leach(22),
Jessie Barber(22),
Sharon Merrill(22),
Lindsay Aldridge(22),
Ryan Kirby(22),
Tim Wagner(22),
Max Schultz(22),
Marian Parrott(22),
Keith Joyce(22),

Guy Church(22),
Annette Hahn(22),
Marguerite Bunn(22),
Lynn Barnes(22),
Nelson Honeycutt(22),
Leslie Pugh(22),
Stacy May(22),
Lloyd Henson(22),
Meredith Pittman(22),
Laura Carey(22),
Jeanne Conway(22),
Lauren Keith(22),
Emily Mitchell(22),
Bill Elmore(22),
Daniel Archer(22),
Jill Singleton(22),
Leslie Ross(22),
Wanda Schneider(22),
Danny Barrett(22),
Neil Gold(22),
Michael Nichols(22),
Carmen Hartman(22),
Don Ayers(22),
Claudia Sims(22),
Karen Kennedy(22),
Benjamin Moody(22),
Rebecca Gardner(22),
Jesse Kerr(22),
Ron Gallagher(22),
Timothy Jacobson(22),
Greg Anderson(22),
Paula Sparks(22),
George Kendall(22),
Harriet Wade(22),
Christine Barrett(22),
Peter Perez(22),
Tara Duke(22),
Peggy Langley(22),
Joshua Blanton(22),
Ruth Lloyd(22),
Katharine Powell(22),
Edgar Alexander(22),
Norma Everett(22),
Marion Buchanan(22),
Lauren Dickerson(22),
Yvonne Waters(22),
Roger Hanna(23),
Milton Starr(23),
Ted Weinstein(23),
Allison Choi(23),
Florence McFarland(23),
Hannah Wall(23),
Ethel Stevens(23),
Audrey Law(23),
Kim McCormick(23),
Regina Gregory(23),

Cecil Bruce(23),
Laurie Porter(23),
Robyn Keith(23),
Dan Goldman(23),
Bruce Kaufman(23),
Elisabeth Carroll(23),
Marie Mack(23),
Donna Dudley(23),
Cynthia Forbes(23),
Holly Solomon(23),
Harriet McLeod(23),
Bryan Howell(23),
Randall Byrne(23),
Sean Oakley(23),
Sally Myers(23),
Kevin Snow(23),
Kim Bolton(23),
Angela Alexander(23),
Toni Dickerson(23),
Kenneth Gorman(23),
Joyce Godfrey(23),
Meredith Wu(23),
Phyllis Duffy(23),
Jim Anthony(23),
Harold Webb(23),
Catherine Wiggins(23),
Pamela Huang(23),
Allan Bond(23),
Lucy Bowers(23),
Brian Jernigan(23),
Molly Browning(23),
Tara Bowers(23),
Christine Ballard(23),
Cheryl Lutz(23),
Joel Donnelly(23),
Peter Rice(23),
Gail Best(23),
Ryan Swanson(23),
Glenda Kessler(23),
Seth Cochran(23),
Toni Guthrie(23),
Kathleen Lindsay(23),
Don Hall(23),
Kent Simmons(23),
Jeffrey Holden(23),
Evan Wise(23),
Susan Keith(23),
Eileen Miller(23),
Sally Stephens(23),
Wayne Marsh(23),
Pat McDowell(23),
Edward Bauer(23),
Paul O'Connor(23),
James Rhodes(23),
Penny Lloyd(23),
Kerry Mitchell(23),

Edith Bloom(23),
Toni Currie(23),
Alvin Lambert(23),
Marguerite Jones(23),
Kerry Tan(23),
Albert Higgins(23),
Greg Beard(23),
Tamara Cain(23),
Jason McFarland(23),
Timothy Norris(23),
Jonathan Pearson(23),
Sheryl Yu(23),
Charlie O'Donnell(23),
Malcolm Camp(23),
Scott Larson(23),
Kathy Forbes(23),
Seth Hedrick(23),
Erika Francis(23),
Jamie Townsend(23),
Samantha Gross(23),
Lee Weber(23),
Joe Vogel(23),
Sarah West(23),
Scott McGuire(23),
Brenda Knox(23),
Christian Silver(23),
Wesley McKay(23),
Tammy Blackburn(23),
Eleanor Warren(23),
Miriam Hanson(23),
Nina McMillan(23),
Ray Underwood(23),
Rita Holder(23),
Phillip Carey(23),
Donna Hale(23),
Cheryl Chan(23),
Jimmy Heller(23),
Erin McKenzie(23),
Terri Gibbs(23),
Doris Simpson(23),
Kimberly Wang(23),
Frederick Wagner(23),
Gretchen Goldstein(24),
Natalie Walton(24),
Leon Pearson(24),
Ashley Houston(24),
Jeanne Crabtree(24),
Amanda Kirby(24),
Luis Zhang(24),
Clyde Howe(24),
Ellen Chappell(24),
Tiffany Cannon(24),
Molly Farmer(24),
Kate Langley(24),
Melanie Melton(24),
Derek Riley(24),

Marcia Sparks(24),
Jimmy Schroeder(24),
Molly Strauss(24),
Henry Potter(24),
Phyllis Mercer(24),
Stacey Kaufman(24),
Juanita Cox(24),
Nelson Lyons(24),
Paul Lester(24),
Glenn Martin(24),
Tracy Proctor(24),
Bob Jacobs(24),
Marsha Cash(24),
Ray Lam(24),
Sherry Adcock(24),
Jay Chase(24),
Daniel Pittman(24),
Ernest Sawyer(24),
Joe Feldman(24),
Susan Hardy(24),
Ashley Christian(24),
Edward Mueller(24),
Kyle Tyson(24),
Glen Wood(24),
Lester Hodges(24),
Nicole Tan(24),
Sean Pickett(24),
Neil Hull(24),
Clara Gibson(24),
Evan Rosenthal(24),
Joann Coble(24),
Shelley Melvin(24),
Jane Roberts(24),
Vicki Carver(24),
Evelyn Rhodes(24),
Jessie McLean(24),
Katherine Hewitt(24),
Tony Adams(24),
Martha Hewitt(24),
Sean Bernstein(24),
Paula Atkins(24),
Linda Wilkerson(24),
Dorothy Beach(24),
Carole Jordan(24),
Carlos Davies(24),
Sheryl Bradford(24),
Brent Allison(24),
Katie Kerr(24),
Timothy Davies(24),
Thomas Braun(24),
Patricia Ferrell(24),
Frances Hanna(24),
Gregory Schneider(24),
Emily Bland(24),
Joanna McKee(24),
Vicki Parsons(24),

Becky Bailey(24),
Caroline Spears(24),
Paige Rollins(24),
Deborah Shelton(24),
Melanie Butler(24),
Michelle Bradshaw(24),
Laurie Herbert(24),
Ruby Sawyer(24),
Danny Barber(24),
Christine Gunter(24),
Colleen Cannon(24),
Carlos Greenberg(24),
Jenny Bland(24),
Albert Ellis(24),
Leah Kinney(24),
Todd Clark(24),
Clifford Wooten(24),
Priscilla Reed(24),
Leo Pratt(24),
Louise Talley(24),
Elisabeth Lewis(24),
Ronald Wallace(24),
Don Doyle(24),
Stephen Wilkinson(24),
Bruce Case(24),
Esther Pickett(24),
Gordon Morrison(24),
Christian Yates(24),
Jim Burnette(24),
Harriet Sloan(24),
Monica Wright(24),
Judy Barr(24),
Erika Boyle(24),
Jack Walsh(24),
Tonya Gill(24),
Max Porter(24),
Joe Whitley(24),
Kurt Levine(24),
Anthony Hutchinson(24),
Debbie Sharma(24),
Shelley Weeks(25),
Stacy Olsen(25),
Keith Chandler(25),
Tamara Hunt(25),
Mary Cochran(25),
Hannah Langston(25),
Edwin Curtis(25),
Allison Norman(25),
Clyde Kenney(25),
Kyle Bolton(25),
Ben Puckett(25),
Miriam Berg(25),
Ann Chandler(25),
Katherine Sutherland(25),
Phyllis Byers(25),
Marianne Mayo(25),

Christina Hubbard(25),
Annie Lucas(25),
Samuel Goodwin(25),
Debbie Chappell(25),
Patrick Padgett(25),
Vickie Stephens(25),
Alice Garrett(25),
Gerald Walsh(25),
Scott Donnelly(25),
Frank Fleming(25),
Vicki Gould(25),
Jeff Hunt(25),
Charlie Glover(25),
Samantha Conrad(25),
William Sherrill(25),
Chris Cochran(25),
James Cochran(25),
Gladys Dorsey(25),
Steven Townsend(25),
Janice McMahon(25),
Charlotte Meyer(25),
Victor Capps(25),
Shirley Hansen(25),
Crystal Martinez(25),
Sharon Cooper(25),
Maria Avery(25),
Zachary Long(25),
Carrie Hardin(25),
Jennifer Hobbs(25),
Jordan Giles(25),
Marsha Pollard(25),
Ruth Clark(25),
John Archer(25),
Julie Bloom(25),
Kathy Thomson(25),
Teresa Kay(25),
Kara Simmons(25),
Thelma Byers(25),
Jonathan Yang(25),
Lindsay Norton(25),
Jackie Desai(25),
Lorraine Jensen(25),
Lillian Walton(25),
Lynne Stokes(25),
Shawn Buckley(25),
Louis Haas(25),
Glen Baldwin(25),
Benjamin Wood(25),
Katharine Hall(25),
Bonnie Washington(25),
Dolores Benton(25),
Gloria Henson(25),
Alan Boyle(25),
Edward Russell(25),
Deborah Byrd(25),
Nicholas Peele(25),

Jacqueline Wilkerson(25),
Edna Strauss(25),
Suzanne Maxwell(25),
Vincent Jernigan(25),
Mary Stevens(25),
Marian Wang(25),
Louise Short(25),
Raymond Schwarz(25),
Luis Everett(25),
Florence Wright(25),
Billy Wang(25),
Milton Sweeney(25),
Marjorie Fitzgerald(25),
Roger Watts(25),
Gretchen Ayers(25),
Marianne Fox(25),
Gail Stuart(25),
Katherine Burns(25),
Christopher Myers(25),
Norman Hong(25),
Philip Kramer(25),
Francis Garcia(25),
Christopher MacDonald(25),
Lawrence Weiss(25),
Philip Cunningham(25),
James Vogel(25),
Sandy Sun(25),
Irene Cates(25),
Regina Hubbard(25),
Meredith Justice(25),
Ruth Hardy(25),
Victor Choi(25),
Gwendolyn Gilliam(25),
Gloria Godwin(25),
Ernest Harrison(25),
Milton Joyner(25),
Ken Marshall(25),
Heather Conner(25),
Robin Bynum(25),
Lori Lang(25),
Alan McConnell(25),
Charlie Carlson(25),
Elizabeth Green(25),
Kathleen Russell(25),
Vincent Grady(25),
Tiffany Gardner(25),
Jose Dale(25),
Edward Prince(25),
Walter Blackburn(25),
Elisabeth Foster(25),
Cindy Glass(25),
Calvin Crawford(25),
William Dickson(25),
Neal McNeill(25),
Jimmy Stark(25),
Lori Knowles(25),

Nancy Capps(25),
Esther Gold(25),
Rick Pridgen(25),
Evelyn Hunter(25),
Ronnie Perry(25),
Joel Han(25),
Karl Wrenn(25),
Jim Hurley(25),
Betty Fleming(25),
Kathleen Hodges(25),
Dana Kent(25),
Fred Garner(25),
Earl Osborne(25),
Timothy French(25),
Rose Greenberg(25),
Laurie McKenzie(25),
Dean Barker(25),
Tom Hurst(25),
James Siegel(25),
David Roy(25),
Bonnie Mann(25),
Larry Callahan(25),
Michele Hamrick(25),
Gayle Cassidy(25),
Lorraine Fischer(25),
Christopher Stein(25),
Jesse Kirkland(25),
Ian Hester(25),
Renee Clarke(25),
Michael Rowland(25),
Maria Hewitt(25),
Lois Lawson(25),
Arnold Creech(25),
Vicki Ellis(25),
Timothy Hendrix(25),
Nina Hood(25),
Brian Frederick(25),
Marian Griffin(25),
Marie Pace(25),
Alvin Middleton(25),
Lisa Eaton(25),
Tamara Curtis(25),
Seth McGee(25),
Victoria McDaniel(25),
Harold Hudson(25),
Brad Ballard(25),
Megan Bullock(25),
Marian George(25),
Sandy Raynor(26),
Mike Whitehead(26),
Charlene Heath(26),
Alison MacDonald(26),
Diana Crane(26),
Neal O'Connor(26),
Bill Klein(26),
Dean Lutz(26),

Tonya Ellis(26),
Katie Katz(26),
Sidney Weber(26),
Zachary Black(26),
Hannah Mueller(26),
Teresa Daniel(26),
Bobby Hines(26),
Lynda Ball(26),
Norman Arthur(26),
Jeanette Montgomery(26),
Chris Chu(26),
Bill Roberson(26),
Debra Palmer(26),
Amanda Larson(26),
Ruby Stout(26),
Nicholas Lancaster(26),
Claudia Williamson(26),
Angela Turner(26),
Diana Vincent(26),
Kathy Riggs(26),
Earl Honeycutt(26),
Eileen Pittman(26),
Tina Bernstein(26),
Troy Browning(26),
Heidi Bowles(26),
Eva Gill(26),
Calvin Godwin(26),
Theresa Schroeder(26),
Linda Decker(26),
Rosemary Choi(26),
Christy Stevenson(26),
Billy Brooks(26),
Harriet Hawley(26),
Martha Werner(26),
Carl Hinson(26),
Joyce Stern(26),
Ted Cassidy(26),
Brent Moss(26),
Frances Capps(26),
Vernon Gibbs(26),
Brad Ray(26),
Sherri Kelly(26),
Geoffrey Roy(26),
Marlene Bailey(26),
Sheila Coley(26),
Teresa Joyner(26),
Michele Davenport(26),
Crystal Morse(26),
Samantha Bullock(26),
Peggy Corbett(26),
Caroline Mayo(26),
Meredith Cummings(26),
Sam Chandler(26),
Bernard Booth(26),
Tom Owen(26),
Francis Wise(26),

Carolyn Willis(26),
Jill Neal(26),
Nicole Nicholson(26),
Miriam Medlin(26),
Mitchell Carr(26),
Vickie Wooten(26),
Leigh Conner(26),
Constance Reese(26),
Eleanor Simmons(26),
Juanita Peele(26),
Juan Eason(26),
Lindsay Blair(26),
Marvin Townsend(26),
Randy Morton(26),
Bernice Douglas(26),
Tonya Batchelor(26),
Lewis Armstrong(26),
Alice Reddy(26),
Audrey English(26),
Norma Hale(26),
Ted Melvin(26),
Keith Dunn(26),
Lynn Cook(26),
Gilbert Adkins(26),
Leslie Eason(26),
Kevin Raynor(26),
Hugh Lane(26),
Sam Duke(26),
Gerald Humphrey(26),
Janet Goldman(26),
Frank Sherrill(26),
Bobby Sawyer(26),
Herbert Doyle(26),
Joyce Woodard(26),
Allen Massey(26),
Edna Hampton(26),
Nicholas Garrett(26),
June Odom(26),
Peter Kenney(26),
Harold Hobbs(26),
Sandy Ross(26),
Lloyd Simpson(26),
Ray Hartman(26),
Steven Dolan(26),
Ruth Rice(26),
Arthur Shannon(26),
Connie Blackburn(26),
Carolyn Nolan(26),
Earl Perez(26),
Jose Hicks(26),
Wesley Field(26),
Judy O'Connell(26),
Katharine McFarland(26),
Edna Cassidy(26),
Eddie Fletcher(26),
Judy Tuttle(26),

Dan Stephens(26),
Jason Hopkins(26),
Jessie Petersen(26),
John Dickerson(26),
Zachary Haas(26),
Harold Norman(26),
Lawrence Lim(26),
Frederick Parks(26),
Tamara Currin(26),
Theresa McDonald(26),
Howard Kenney(26),
Christopher Washington(26),
Sidney Terry(26),
Stacy Byers(26),
Zachary Dougherty(26),
Dorothy Shore(26),
James Joyce(26),
Nathan Hobbs(26),
Melissa Sherman(26),
Chris Teague(26),
Max Hewitt(26),
Kay Fletcher(26),
Phyllis Bell(26),
Jessie Williams(26),
Edith Link(26),
Ted Hobbs(26),
Terry Desai(26),
Jennifer Whitley(26),
Richard Patrick(26),
Anne Greer(26),
Leah Parks(26),
Ross Puckett(26),
Guy Marcus(26),
Kathleen Fletcher(26),
Julia Dillon(26),
Evan Carver(26),
Hilda Sherrill(26),
Anita Humphrey(26),
Jeremy Savage(26),
Teresa Dudley(26),
Oscar Bernstein(26),
Bruce McGuire(26),
Gene Daniel(26),
Carlos Baker(26),
Ross Powers(26),
Marion Meadows(26),
Gerald Adcock(26),
Mitchell Terrell(26),
Hazel Merritt(26),
Amy Miller(26),
Carole White(26),
Lee Hardy(26),
Clyde McFarland(26),
Shirley Kenney(26),
Wesley Douglas(26),
Ken Hewitt(26),

Jeanne Powers(26),
Tracy Clayton(26),
Maxine Ennis(26),
Marc Choi(26),
Marian Singer(26),
Melanie Lucas(26),
Kenneth Leonard(26),
Harold O'Neill(26),
Francis Wood(26),
James Small(26),
Leslie Cook(26),
Colleen Kelley(26),
Penny Ross(26),
Audrey Garcia(26),
Janet Winters(26),
Stuart Park(26),
Grace House(26),
Claire Taylor(26),
Edith Briggs(26),
Marcus Jensen(26),
Philip Peele(26),
Jon Barrett(26),
Rhonda Pritchard(26),
Eva Schmidt(26),
Connie Powers(26),
Rachel Sutton(26),
Kristin Leonard(26),
Kimberly Kane(26),
Amy Lester(26),
Kristin Schroeder(26),
Jon Gray(26),
Becky Hull(26),
Kelly Allen(26),
Marshall French(26),
Marian Solomon(27),
Elsie Barton(27),
Douglas Ross(27),
Melinda Proctor(27),
Samantha Hardin(27),
Scott Golden(27),
Terry Barton(27),
Miriam Wong(27),
Jill Kent(27),
Malcolm Chan(27),
Bradley Duke(27),
Catherine Summers(27),
Patrick Archer(27),
Alvin Frederick(27),
Gary Scarborough(27),
Shelley Dunn(27),
Billy Carroll(27),
Richard Harris(27),
Gayle Stanley(27),
Greg Berger(27),
Paige Dudley(27),
George Mathews(27),

Sylvia Hsu(27),
Cameron Bowers(27),
Allen Randall(27),
Nelson Medlin(27),
Jeff Dodson(27),
Tony Fletcher(27),
Ross Hawkins(27),
Marcus Howe(27),
Sidney Frye(27),
Charlene Burnette(27),
Emma Day(27),
Sean Snow(27),
Amy Peters(27),
Bryan Weiss(27),
Ronald Brandon(27),
Ronnie Myers(27),
Aaron McKenzie(27),
Sam Blum(27),
Robyn Dickinson(27),
Claudia Hill(27),
Connie Sellers(27),
Anthony Buckley(27),
Natalie McKinney(27),
Crystal Kramer(27),
Maxine Holden(27),
Ken Ritchie(27),
Stacy Floyd(27),
Glenn Phelps(27),
Juanita Lanier(27),
Roy Gentry(27),
Karen Noble(27),
Tracy Sparks(27),
James Kang(27),
Ruth Bass(27),
Adam Carroll(27),
Annie Lim(27),
Howard French(27),
Andrew Rowland(27),
Terry Schultz(27),
Russell Park(27),
Bryan Kennedy(27),
Jessica Friedman(27),
Glenda Parrish(27),
Raymond Crane(27),
Leah McCoy(27),
Joel Blanchard(27),
Veronica Gates(27),
Ronnie Boone(27),
Sharon Baxter(27),
Dwight Arnold(27),
Marcus Moran(27),
Becky English(27),
Bruce Hoyle(27),
Carla Ward(27),
Priscilla George(27),
Ralph Tate(27),

Laurence Benson(27),
Joe Lloyd(27),
Louis Langley(27),
Dean Hoyle(27),
James Burnette(27),
Vivian Albright(27),
Erin Austin(27),
Wendy Poe(27),
Bernice Powell(27),
Kevin Phillips(27),
Pauline Katz(27),
Nina Pratt(27),
Lori Ritchie(27),
Ann Braswell(27),
Tony Weeks(27),
Edgar Branch(27),
Elizabeth Hayes(27),
Heidi Blake(27),
Scott Knowles(27),
Mary Hancock(27),
Dale Gorman(27),
Evan Singer(27),
Brooke Reid(27),
Arthur McCarthy(27),
Connie Conrad(27),
Jane Spivey(27),
Renee McNamara(27),
Hannah Buchanan(27),
Arlene Armstrong(27),
Jordan Perkins(27),
Nancy Crane(27),
Nathan Callahan(27),
Florence Rosenberg(27),
Louis Sharp(27),
Donna Faulkner(27),
Heather Huff(27),
Joel Woodard(27),
Jane Case(27),
Emma Kearney(27),
Tony McDowell(27),
Kelly Swanson(27),
Annette Hoyle(27),
Troy Monroe(27),
Elisabeth Feldman(27),
Peter Starr(27),
Benjamin Quinn(27),
Connie Roach(27),
Debra Kumar(27),
Marion Mayer(27),
Jimmy Holloway(27),
Martin Houston(27),
Vernon Brock(27),
Lynn Silverman(27),
Katie Sharpe(27),
Erica Best(27),
Ryan Strauss(27),

```
      Vanessa Walsh(27),
      Glenda Howard(27),
      Matthew Hobbs(27),
      Victor Deal(27),
      Brenda Hodge(27),
      Laura Griffin(27),
      Rita Keller(27),
      Leroy Melvin(27),
      Leonard Greene(27),
      Laurie McNeill(27),
      Sidney Schultz(27),
      Marshall White(27),
      Dan Richmond(27),
      April Gardner(27),
      Annette Horner(27),
      Jeanette Barbee(27),
      Diana Rivera(27),
      Jenny Beasley(27),
      Kim Russell(27),
      Yvonne Swain(27),
      Kevin Beatty(27),
      Carlos Block(27),
      Tony Byrd(27),
      Gwendolyn Sweeney(27),
      Erik Christian(27),
      Roberta Hood(27),
      Leroy Michael(27),
      Vernon Hansen(27),
      Bradley Dougherty(27),
      Katie Kelly(27),
      Tonya Bridges(27),
      Sharon Simpson(27),
      Vanessa Meadows(27),
      Kathryn Foley(27),
      Randall Ivey(27),
      Audrey West(27),
      Cindy McLamb(27),
      Gary Saunders(27),
      Tara Parks(27),
      Alvin McKenna(27),
      Julie Lindsey(27),
      Tina Flowers(27),
      Angela Jernigan(27),
      Tom Fitzgerald(27),
      Annie Knowles(27),
      Albert Wiley(27),
      Theodore Gould(27),
      Franklin Vick(28),
      ...]
```

In [63]: `customers[0] < customers[1]`

Out[63]: True