# Part 1

1

|  |
| --- |
| >>>fullName = "Gordon Ng" |

2

|  |
| --- |
| >>> len(fullName)  9 |

3

|  |
| --- |
| >>> print(fullName[0:1])  G |

4

|  |
| --- |
| >>> print(fullName[-1:])  g |

5

|  |
| --- |
| fullName[name\_len]  Traceback (most recent call last):  File "<pyshell#6>", line 1, in <module>  fullName[name\_len]  NameError: name 'name\_len' is not defined |

# Part 2

|  |
| --- |
| >>> firstName = fullName[:6]  >>> lastName = fullName[-2:]  >>> greeting = "Hello, " + firstName  >>> lastName >= "M"  True  >>> "an" in fullName  False  >>> fullName8 = fullName \* 8 |

# Part 3

|  |
| --- |
| >>> fullName.index("Go")  0  >>> lastSpace = fullName.rindex(" ")  >>> lastName2 = fullName[lastSpace + 1 :]  >>> firstName2 = fullName[:lastSpace]  >>> fullNameNoVowels = fullName.replace("o", "-")  >>> x = fullName.lower()  >>> x2 = x[0:1]  >>> y = fullName.upper()  >>> y2 = y[1:]  >>> fullNameCapital = x2 + y2  >>> print(fullNameCapital)  gORDON NG  >>> firstSpaceLocation = header.index(" ")  >>> newHeader = header[firstSpaceLocation + 1:]  >>> secondSpaceLocation = newHeader.index(" ")  >>> newHeader2 = newHeader[:secondSpaceLocation]  >>> atLocation = newHeader2.index("@")  >>> domain = newHeader2[atLocation:]  >>> print (domain)  @gmail.com |

# Part 4

|  |
| --- |
| >>> x = input("Enter an integer: ")  Enter an integer: 123  >>> x  '123'  >>> type(x) == str  True  >>> x[0]  '1'  This displays the first character in the index of x which is 123. The first character is 1.  >>> x + 10  Traceback (most recent call last):  File "<pyshell#66>", line 1, in <module>  x + 10  TypeError: can only concatenate str (not "int") to str  I had a traceback. What happened is that I tried to add a string to an integer which failed.  >>> x = int(x)  >>> x[0]  Traceback (most recent call last):  File "<pyshell#68>", line 1, in <module>  x[0]  TypeError: 'int' object is not subscriptable  >>> x + 10  133  I made x into a string meaning I can no longer determine the first character in the string as it is now an integer. I got 133 since its now 123+10.  >>> x = input("Enter an integer: ")  Enter an integer: hello  >>> x  'hello'  >>> type(x) == str  True  >>> x[0]  'h'  >>> x+10  Traceback (most recent call last):  File "<pyshell#74>", line 1, in <module>  x+10  TypeError: can only concatenate str (not "int") to str  >>> x = int(x)  Traceback (most recent call last):  File "<pyshell#75>", line 1, in <module>  x = int(x)  ValueError: invalid literal for int() with base 10: 'hello'  x  >>> x[0]  'h'  >>> x + 10  Traceback (most recent call last):  File "<pyshell#77>", line 1, in <module>  x + 10  TypeError: can only concatenate str (not "int") to str  I entered a word which int could not convert to an integer making it impossible to add a number to it.  >>> variable = "constant"  >>> x = variable  >>> y = "variable"  >>> x  'constant'  >>> y  'variable'  >>> print(x,y)  constant variable  >>> print(x == "variable", y == "variable")  False True  >>> print(x == variable, y == variable)  True False |

# Part 6

|  |
| --- |
| >>> for chr in fullName:  print(chr)    G  o  r  d  o  n    N  g  prints every letter individually |

|  |
| --- |
| >>> for letter in fullName:  print(letter.lower())    g  o  r  d  o  n    n  g  lowercase all printed letters |

|  |
| --- |
| >>> for letter in fullName:  print(letter, result)  result += 1    G 0  o 1  r 2  d 3  o 4  n 5  6  N 7  g 8  >>> print(fullName, result)  Gordon Ng 9  Individual letters and a counter that can describe the position of the letters in the string. It has not changed |

|  |
| --- |
| >>> result = ""  >>> for x in fullName:  result = result + x.upper() + x.lower()    >>> print(result)  GgOoRrDdOoNn NnGg  >>>  My name printed in alternating letters where results defines it as an uppercase letter + a lowercase letter on each run where a letter changes. |

|  |
| --- |
| >>> for y in fullName:  result = y + result    >>> print (result)  gN nodroG  my name is being printed backwards, I think it might be because y is being reversed in some way so that its working backwards on the string? |