Introduction to Data Science

Moinul Islam

9/16/2022

Table of contents

Introduction to Data Science

This is a data science book for beginners. This book contains the basic Python and R programming skills to start your career as a data scientist

1 Introduction to Python

1.1 Prerequisites

- There are three things we need to run the code in this lecture:
  - Python 3.10 (https://www.python.org/downloads/)
  - PyCharm community or professional (https://www.jetbrains.com/pycharm/)
  - A handful of other packages

1.2 Variables

- Let's start using a variable
- Variable names can contain only letters, numbers and underscores
- Space are not allowed in variable name
- Variables name should be short and descriptive

```
message = "Hello world!"
print(message)
```

```
Hello world!
```

1.3 Strings

- A *string* is a series of characters
- In python, anything inside a quotes is considered a string
- We can use single or double quotes around the strings

```
message = "Python is a programming language"
print(message)
```

```
Python is a programming language
```

```
message = 'I told my friend, "Python is a programming language"'
print(message)
```

```
I told my friend, “Python is a programming language”
```

1.3.1 Change case of string

- A lower case string can be changed to a title

```
name = "moinul islam"
print(name.title())
```

```
Moinul Islam
```

- We can also change the string to all upper or all lower

```
name = "moinul islam"
print(name.upper())
```

```
MOINUL ISLAM
```

1.4 Numbers

- Numbers are used quite often in python
  - Integers: We can add (+), multiply (*), exponent (**), and divide (/) integers in Python

```
print(2+3)
```

```
5
```

```
print(3-2)
```

```
1
```

```
print(2*3)
```

```
6
```

```python
print(3**3)
```

```
27
```

```python
print(3/2)
```

```
1.5
```

  - Python supports the order of operation too.

```python
print(2 + 3*4)
```

```
14
```

```python
print((2+3)*4)
```

```
20
```

  - Floats: Python calls any number with a decimal point a *float*

```python
print(2*0.2)
```

```
0.4
```

  - Integers and floats: When we divide two numbers, even if they are integers, you will always get a float

```python
print(4/2)
```

```
2.0
```

- Underscore in numbers
  - When you are writing long numbers, you can group digits using underscores

```python
universe_age = 14_000_000_000
print(universe_age)
```

```
14000000000
```

- Multiple assignments
  - We can assign the value to more than one variables using just a single line

```python
x, y, z = 12, 3, 5
print(x, y, z)
print(z)
```

```
12 3 5

5
```

- Constants
  - A *constant* is a variable whose value stay the same throughout the life of a program.
  - Python does not have built in command for constant
  - However, a variable name with all capital letters treated as constant

```python
ELON_MASK  = 1_000_000_000_000
print(ELON_MASK)
```

```
1000000000000
```

1.5 Comments

- In Python, hash (#) indicates a comment.
- Anything following a # mark in your code is ignored by Python

```python
# Say hello to your friends
print("Hello friends")
```

```
Hello friends
```

1.6 Lists

- List the elements of a variable

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
print(bicycles)
```

```
[‘trek’, ‘cannondale’, ‘redline’, ‘specialized’]
```

- Access an element from a list

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
print(bicycles[0])
```

```
trek
```

1.6.1 Neat outcome

- You can format the element “trek” even more neatly by using title() method

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
print(bicycles[0].title())
```

```
Trek
```

1.6.2 Print string

- You can print the strings that you are interested
- It starts counting from zero

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
print(bicycles[1])
```

```
cannondale
```

- Python has special syntax to call the last item of a list

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
print(bicycles[-1])
```

```
specialized
```

1.7 Functions

```python
bicycles = ["trek", "cannondale", "redline", "specialized"]
message = f"my first bicycle was a {bicycles[0].title()}"
print(message)
```

```
my first bicycle was a Trek
```

- Modifying elements in a list

2 Introduction to R

2.1 Prerequisites

- There are four things we need to run the code in this lecture:
  - R (https://www.r-project.org/)
  - RStudio (https://www.rstudio.com/products/rstudio/download/)
  - A collection of R packages called the *tidyverse* (https://www.tidyverse.org/)
  - A handful of other packages

2.2 Use necessary libraries

```
library("tidyverse")
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.3.6     v purrr   0.3.4
v tibble  3.1.8     v dplyr   1.0.9
v tidyr   1.2.0     v stringr 1.4.1
v readr   2.1.2     v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library("nycflights13")
library("gapminder")
library("Lahman")
```

2.3 Running R code

References