Python for Data Science

Install Anaconda and Spyder

What is Anaconda?

- Free and open-source distribution of the python and R
- Predominantly used for Data Science, Machine Learning and large scale data processing
- Over 12 million user
- 1400 packages..

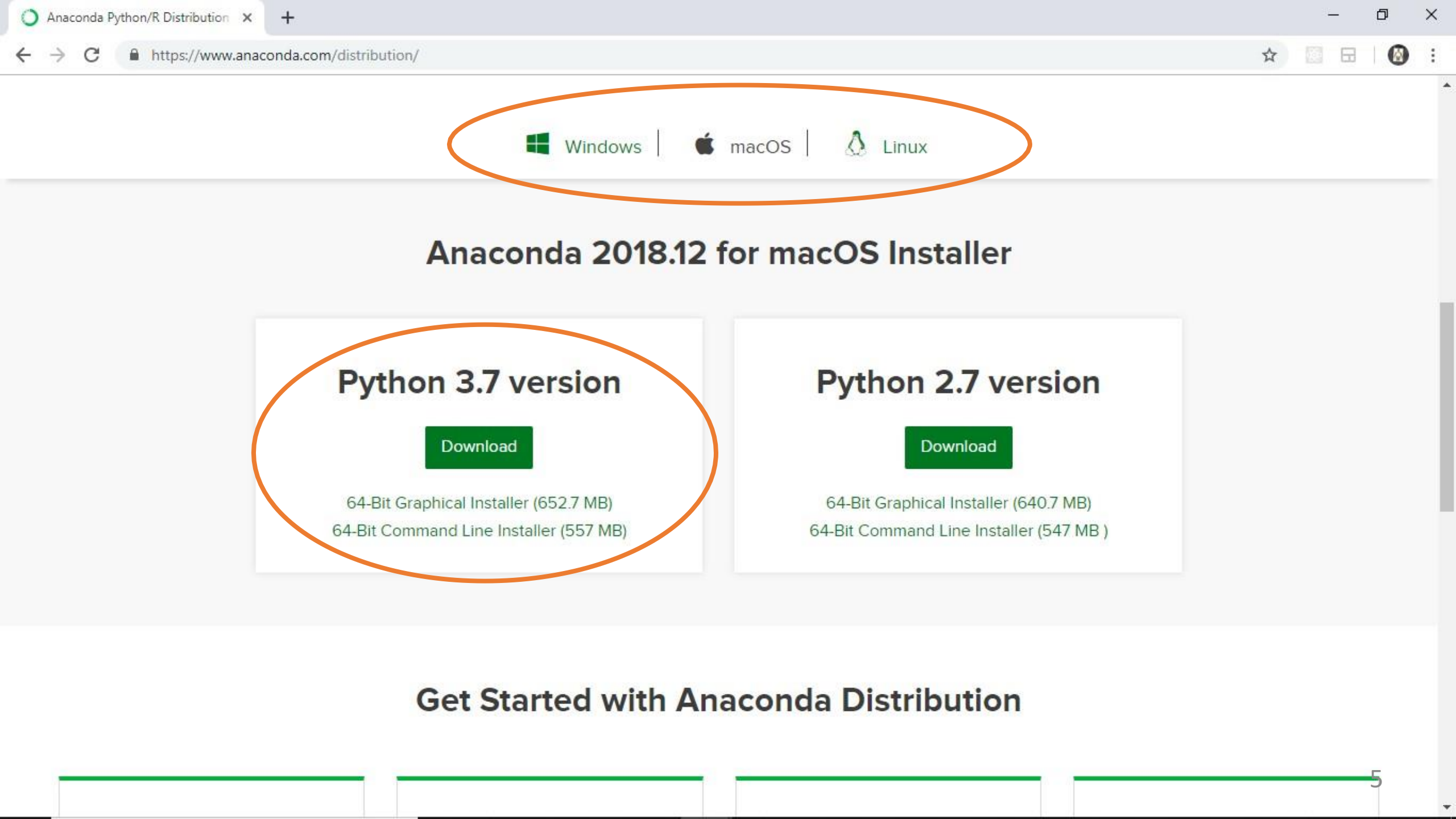
System Requirements

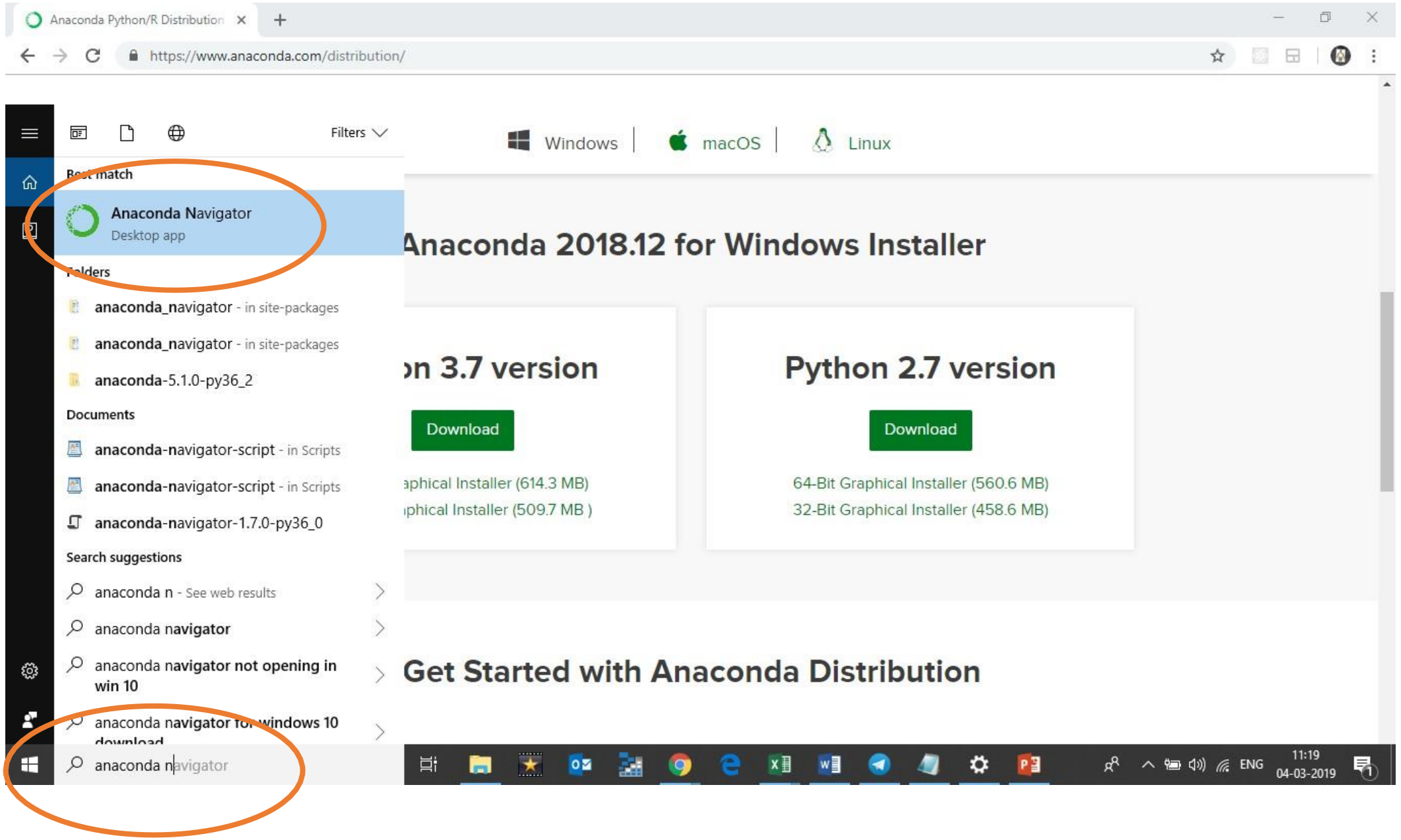
- Operating system: Windows 7 or newer, 64-bit macOS 10.10+, or Linux, including Ubuntu, RedHat, CentOS 6+, and others.

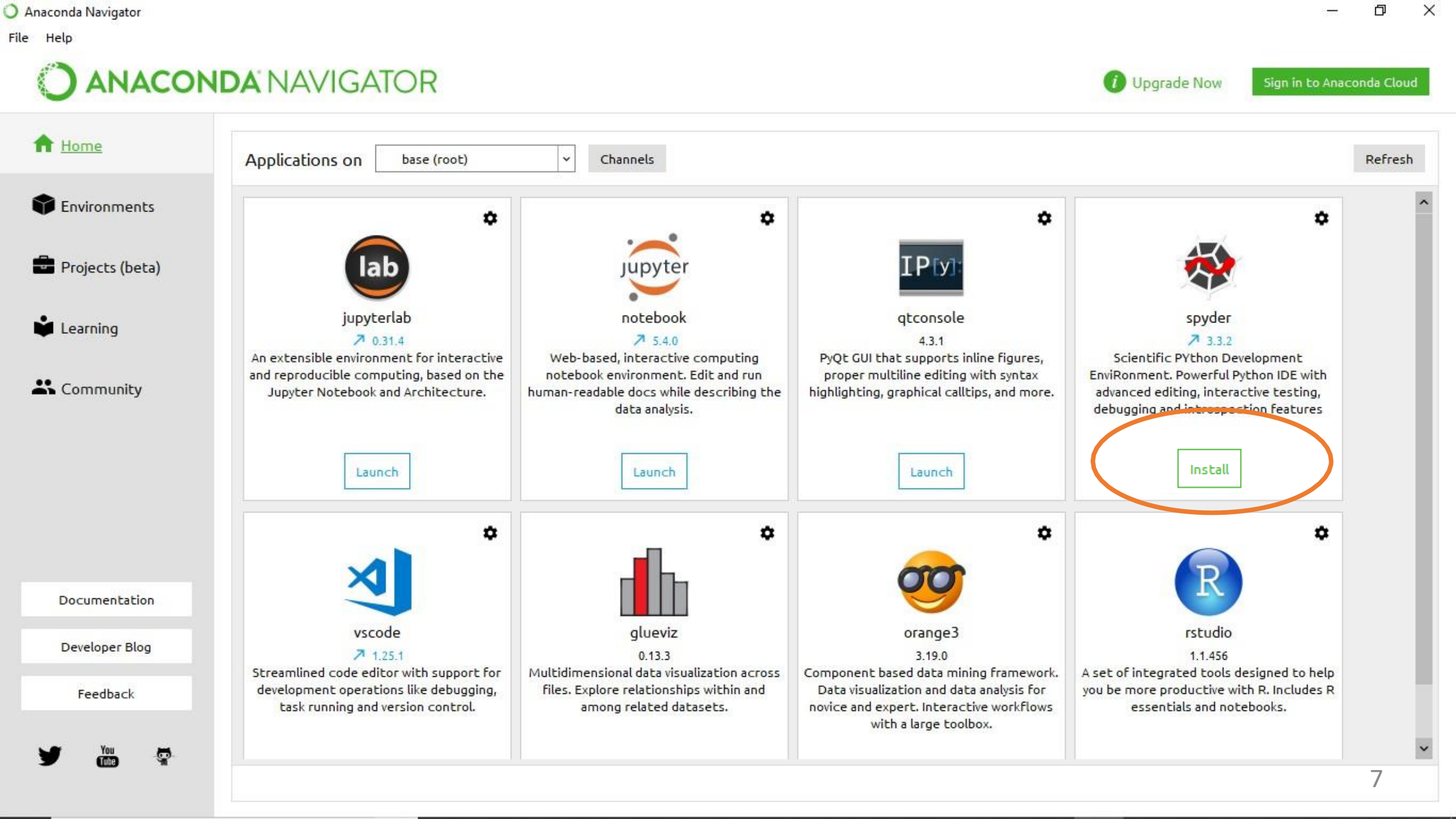
- Older versions of Anaconda are available in archive

- System architecture: Windows- 64-bit x86, 32-bit x86; MacOS- 64-bit x86; Linux- 64-bit x86, 64-bit Power8/Power9.

- Minimum 5 GB disk space to download and install.


Anaconda Python/R Distribution
https://www.anaconda.com/distribution/
Windows
macOS
Linux
Anaconda 2018.12 for macOS Installer
Python 3.7 version
Download
64-Bit Graphical Installer (652.7 MB)
64-Bit Command Line Installer (557 MB)
Python 2.7 version
Download
64-Bit Graphical Installer (640.7 MB)
64-Bit Command Line Installer (547 MB)
Get Started with Anaconda Distribution

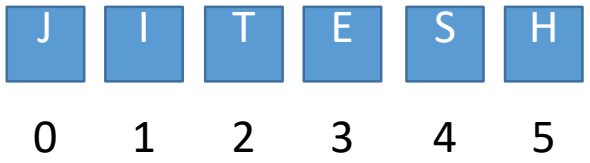

Anaconda Python/R Distribution
https://www.anaconda.com/distribution/
Filters
Anaconda Navigator
Desktop app
anaconda_navigator - in site-packages
anaconda_navigator - in site-packages
anaconda-5.1.0-py36_2
Documents
anaconda-navigator-script - in Scripts
anaconda-navigator-script - in Scripts
anaconda-navigator-1.7.0-py36_0
Search suggestions
anaconda n - See web results
anaconda navigator
anaconda navigator not opening in win 10
anaconda navigator
Windows
macOS
Linux
Anaconda 2018.12 for Windows Installer
Python 2.7 version
Download
64-Bit Graphical Installer (560.6 MB)
32-Bit Graphical Installer (458.6 MB)
Get Started with Anaconda Distribution
ENG
11:19
04-03-2019


Anaconda Navigator
File Help
ANACONDA NAVIGATOR
Upgrade Now
Sign in to Anaconda Cloud
Home
Environments
Projects (beta)
Learning
Community
Documentation
Developer Blog
Feedback
Applications on
base (root)
Channels
Refresh
jupyterlab
0.31.4
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch
notebook
5.4.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch
qtconsole
4.3.1
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.
Launch
spyder
3.3.2
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Install
vscode
1.25.1
Streamlined code editor with support for development operations like debugging, task running and version control.
glueviz
0.13.3
Multidimensional data visualization across files. Explore relationships within and among related datasets.
orange3
3.19.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
rstudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Variable Types

Variables

Jitesh



- Reserved memory locations or space
- Standard Data Types of variables
  - Numbers
  - String
  - Boolean
  - List
  - Tuple
  - Dictionary

True
False

34

56

7383

3.0

4.56

Integers

Float

List

- Sequence of elements
- Similar to array in other programming languages
- Elements are indexed

List1 = [“Jitesh”, “John”, “Alia”, “Lise”]

0 1 2 3

Tuple

- Sequence of elements
- Similar to array in other programming languages
- Elements are indexed

tuple1 = (“Jitesh”, “John”, “Alia”, “Lise”)

0 1 2 3

- Tuples are immutable
- Faster to process

Dictionary

- Collection of key-value pairs
- Values can be accessed using the Key
- Used for JSON format conversion

Address

Key	Value
Street	180 Adams Street
City	Chicago
State	IL
Country	USA

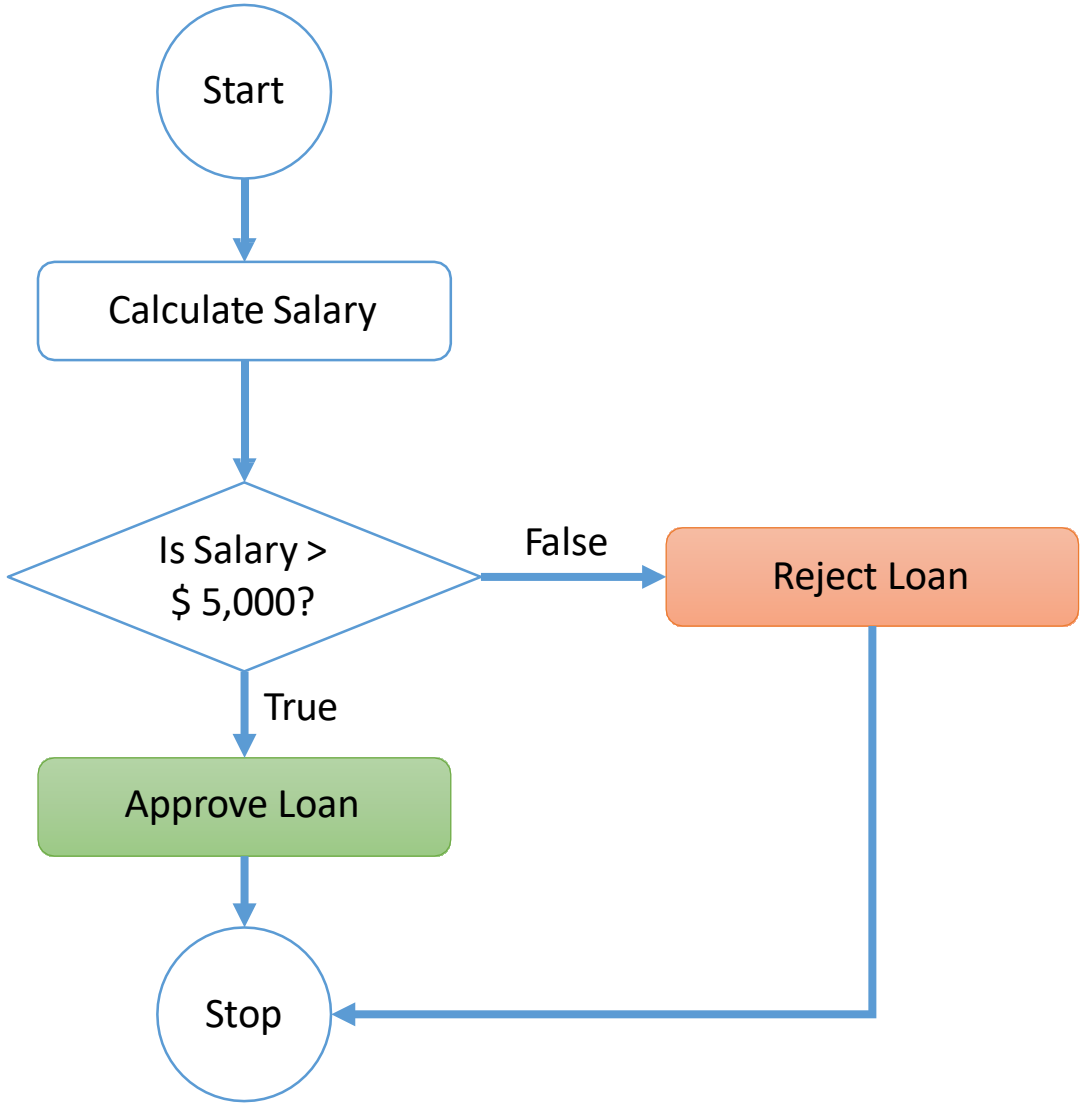
Address = { 'Street': '180 Adams Street', 'City': 'Chicago', 'State' : 'IL', 'Country': 'USA'}

Address['Street'] → '180 Adams Street'

Address['State'] → 'IL'

Conditional Statement

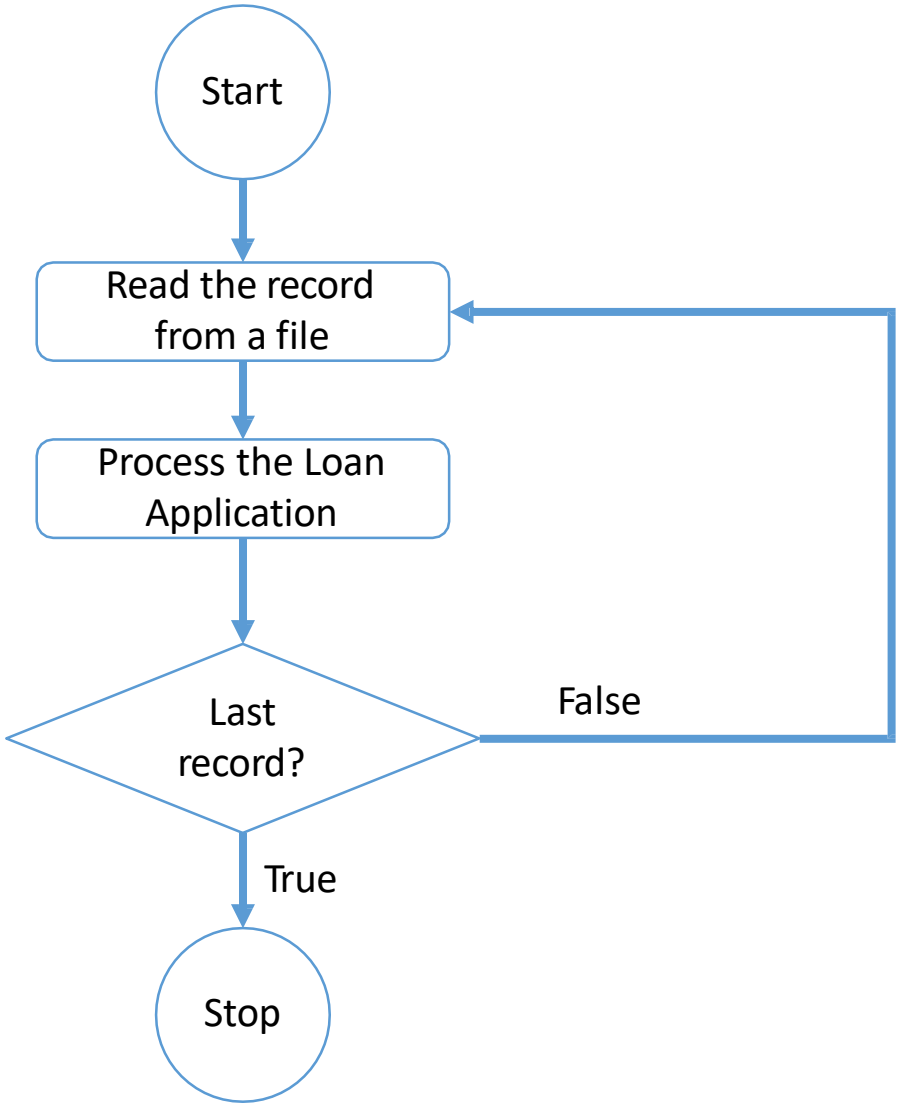
Conditional Statement




```
1
2 salary = 6000
3
4 if salary > 5000:
5     print("Approved")
6 else:
7     print("Rejected")
8
9
10 varT = True
11 varF = False
12
13 if varT:
14     print("Value is True")
15 else:
16     print("value is False")
17
```

Loops or Iterative Statements

Loops


Start
Read the record from a file
Process the Loan Application
Last record?
False
True
Stop

For Loop

```
1
2 for variables in sequence:
3     statements
```

Execute the block as long as there is value in the sequence.

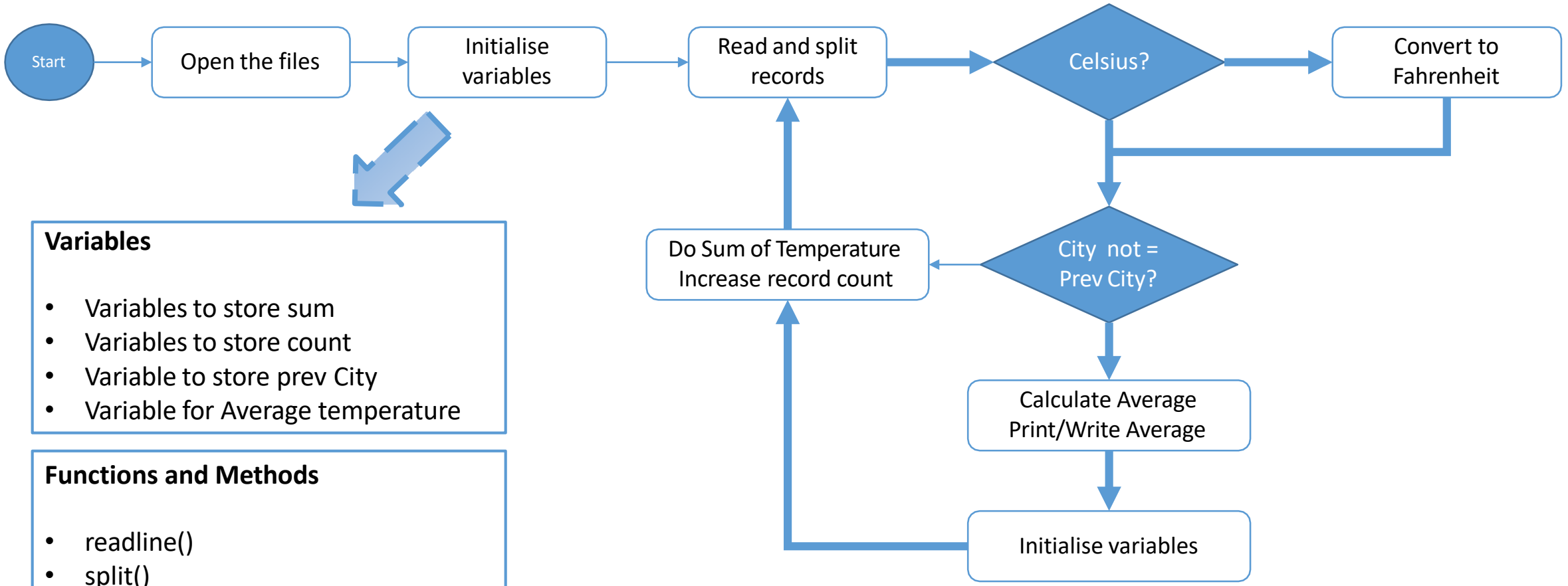
While Loop

```
1
2 while <condition>:
3     statements
4
```

Execute statements “while” the condition is true

```
 8
 9 i=0
10 while i < 10:
11     print ("hello ")
12     i+= 2
```

File Processing Exercise Basic Flowchart



Variables

- Variables to store sum
- Variables to store count
- Variable to store prev City
- Variable for Average temperature

Functions and Methods

- readline()
- split()
- seek()
- rstrip()
- float for string to float conversion
- str for float to string conversion