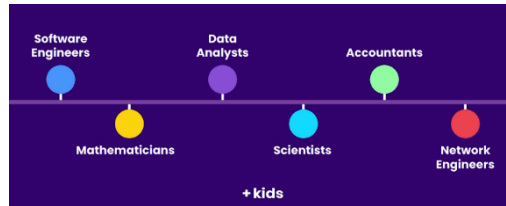


1. Getting Started

2. Fastest growing Language
3. Most Popular Language
4. With Python we can write programs fast in less time and with fewer lines of code
5. Ideal Programming language to learn first
6. Python is used by:

- Software Engineers
- Data Analysts
- Accountants
- Mathematicians
- Scientists
- Network Engineers
- + Kids 😊



```
C# str.Substring(0, 3)
JAVASCRIPT str.substr(0, 3)
PYTHON str[0:3]
```

7. Multi-purpose programming language

- AI/ML
- Hacking
- Software Testing
- Desktop Apps
- Mobile Apps
- Web Apps
- Data Analysis
- Automation



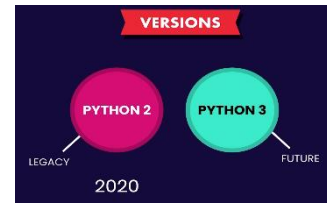
8. Salary of an average Python Developer is 115,835 USD\$ according to Indeed.com as per March 2018
9. Python is a high-level language which means it's easier for humans and the computer will take care of all the other complex tasks like memory management as we do in C++



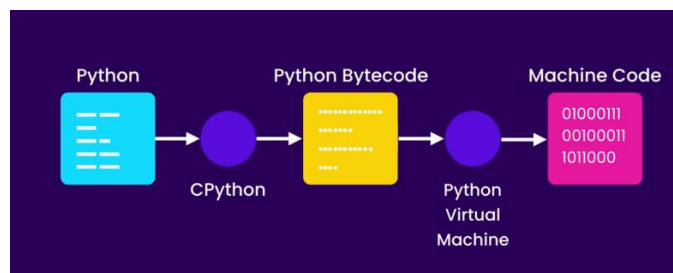
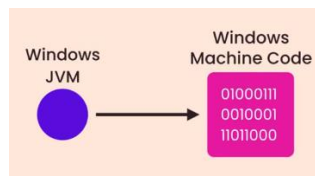
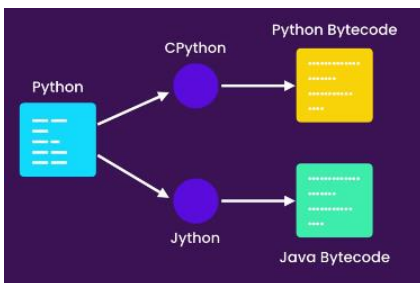
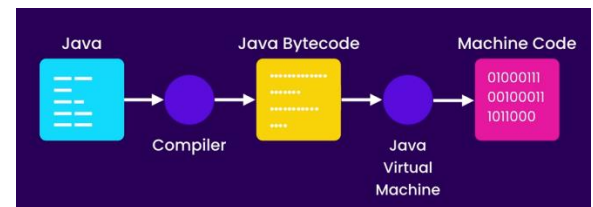
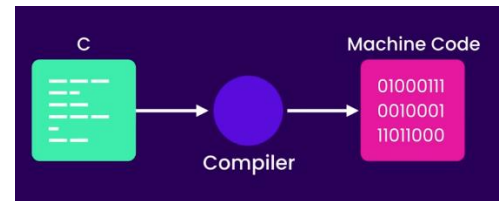
10. Python is Cross-Platform language you can build apps that can run over Windows, Macintosh and Linux operating systems
11. Python has a huge community, you can get help from various mediums such as video tutorials, courses, books, blogs, articles and code snippets
12. Python has a large ecosystem of libraries, frameworks and tools. Whatever you are going to do, it's most likely that someone out there has already build that stuff. So, you don't need to re-invent the wheel or may get help
13. Python has been in the market for almost 20 Years



14. Python 2 is legacy version and it's official support ended back in 2020, python 3 is the python for future
15. Python interpreter is basically a program which executes python code
16. In Programming, an expression is a piece of code which produces a value E.g: $2 + 2 \Rightarrow 4$
17. Syntax is the grammar of computer programming languages
18. An IDE (Integrated Development Environment) is basically a code editor with some fancy productivity booster features
19. Debugging is finding and fixing bugs in a program
20. Auto-completion lets you complete the syntax of your code
21. Python programs have .py file extension
22. Putting brackets in python () means calling a function which means executing that function
23. For Built –in Functions we use a metaphor, think of a remote of your Television, on that remote you have plenty of different built-in, default functions like volume up-down, channel change, turn-off etc.
24. Always use double or single quotes when working with strings
25. Linting is analyzing code for potential errors before execution, linters are small utility extensions which perform linting
26. Code formatting is making code cleaner and readable same as we format our articles, newspapers etc.
27. Code Snippets are the re-usable blocks of code
28. Unit testing involves writing a bunch of tests for our code and we can run these tests in an automated fashion to make sure that our code is behaving correctly
29. PEPs are python enhancement proposals
30. PEP 8 is the most common and famous PEP used by python developers
31. PEP 8 is a style guide for code formatting, if you follow it your code will end up being consistent
32. A Style guide is a document which defines a bunch of rules for formatting code
33. Formatter Autopep 8 is the official PEP formatter tool that comes with the official python extension and is most commonly used by python developers
34. FormatOnSave is the feature we can use in VSCode to format our code according to the PEP 8 rules by formatter autopep 8 on saving the file
35. We can use the code runner extension in VSCode for running our code
36. Code Runner now comes built-in VSCode
37. Python >>> 1- Language 2- Implementation
38. Python Language is the specification that defines a set of rules and grammar for writing python code



39. Python implementation is a program which understands the python rules and can execute python code
40. CPython is the default implementation written in C
41. Jython is written in Java
42. IronPython is written in C#
43. PyPy is a subset of Python itself
44. The technical reason behind having a variety of these implementations is one can use respective implementation for adding snippets of that language easily
45. Computers understand machine code
46. We Humans understand text based languages like python, C etc.
47. A C Compiler is a program which converts C code into machine code
48. Machine code is specific to the type of computer processor (Windows PC >> MAC)
49. Java Byte Code is a portable language not specific to a hardware platform (intermediate language)
50. Java Virtual Machine (JVM) converts Java Byte Code into Machine Code
51. JVM Windows knows how to convert Java byte code into a Machine code that windows processor can understand
52. Python code when executed using Jython converts initially to Java bytecode which provides developer the ability to add Java snippets



- We use variables to store data in computer's memory
- Variable is a label to a specific memory location and the data stored there
- Variable is just like a Package Box- You can put your luggage inside and can display name tags- same is the concept of variables
- Variable references to specific memory location
- Primitive types are the basic building blocks
- Primitive types are the basic data types offered by programming languages
- In python, primitive data types are
 - 1- Numbers
 - 2- Strings
 - 3- Booleans
- Numbers can be:
 - 1- Integers
 - 2- Floats
- While working with strings/text we need to surround text with double/triple quotes
- Booleans are YES/NO – True/False (English Language analogy)
- Booleans are used to make decisions
- Python is a case sensitive language- False != false
- Variable names should be descriptive and meaningful rather mystical, it will lead to a more maintainable code
- When Dealing with coordinates you can use mystical characters like x,y,z that's and exception
- Always use lower cased variable names
- Use underscores to separate variable names, it's easy to read
- Always use spaces while variable declaration – course = 5
- Normally we use triple quotes for a long message like body of an email
- In Python the strings are Zero 0-Indexed
- We use len function to find the length of a string/number of characters in string

- We use brackets notation to access any character & Slicing strings
- # Escape Sequences
- # \\
- # \'
- # \"
- # \n
- A comment is an additional note which is not executed by the python interpreter- denoted as #
- String concatenation is replaced by Formatted Strings
- Formatted String is actually an expression that is evaluated at run-time
- We can put any valid expression in curly braces of formatted strings as it's an expression to be evaluated at run-time

```
full_name = f"{first_name} {last_name}"
```

- In python we call functions as methods
- It's an OOP term, everything in Python is an object and has a function called as methods which we can access using the dot notation
- A module is a separate python file including some code
- Two objects of same type can be concatenated
- Falsy values in Python
- "" Empty String
- 0 Zero
- None

- We use comparison operators to compare values
- We can also use comparison operators over strings
- Bag > Apple == True Because B of Bag comes after A of Apple
- Ord function is used to show the numeric or ASCII representation of Alphabets
- Pay Great Attention to the Indentation
- AutoPep 8 Recommends 4-White Spaces as Indentation
- You can have as many elif statements as you want
- Else is optional
- Ternary Operator provides a clean look & feel of code
- In Python we have 3 Logical Operators to model more complex situations
- AND, OR & Not
- Logical Operators are short circuits, Evaluations ends-> (AND == First False)
(OR == First True)
- We use loops to create repetition
- Type() function expresses the type of an object
- Range is a complex data type and is iterable
- We use For Loops to iterate over iterable objects
- We should have a break away for Infinite loops

- Functions are re-usable chunks of code
- In Real scenarios we don't write whole program in one file
- Rather we break the code into more maintainable chunks of code in different files
- PEP8 recommends 2 lines break after function declaration
- Parameter is the input you define for your function
- Argument is the actual value for that parameter
- Parameter : Argument == > Key: Value
- By Default Parameters are required
- There are 2 Categories of Functions
- 1- Functions which perform a task
- 2- Functions which return a value
- Best Practice is to use #2
- By Default in Python every function provides NONE value which is an object that shows absence of a value
- NONE is classified as a function that carries out a task
- Keyword Arguments are helpful in understanding and reading purposes
- Optional parameters always should be placed after required ones
- Lists are modifiable and tuples are not
- *argument is used to have multiple arguments
- **argument is used to have multiple Keyword arguments or Key-Value Pairs
- Scope refers to the region of the code where a variable is defined
- Local variables are stick inside to a single function
- Local Variables have short life time
- Global Variables can be accessed anywhere in the file
- Global Variables are really evil ~ Memory 🐱
- Best Practices include the use of local variables
- The Use of Global keyword to manipulate the global variables is really a bad practice

- In Python every object in a list can be of different data type
- List == []
- Using the brackets notation [] we can access and play around with lists same as strings
- A list can be modified
- Enumerate function gives index information stored in a tuple → (0, 'a')
- Tuples are read only
- A simple, anonymous, one-line function which we can pass to other functions is called as Lambdas
- We can have a list of lists as well
- A Matrix is a 2-dimensional list
- Zeros = [0] * 100 we get a list of 100 zeroes as list items
- Lists can be concatenated
- List() function is used for the creation of list items by iteration
- Negative index like -1 will give last item of the list
- :: is step in Brackets notation → [::2]
- [::-1] causes the reversal of a list
- You can assign variables to different list items
- List unpacking is assigning list items to multiple variables
- *parameter → Python interprets it as a collection of arguments and store it in a list
- We can only unpack a list if we have the sufficient variables to accommodate list items
- Append method is used to insert something at end of a list
- Insert method is used to insert something at a specific position in a list
- Pop method is used to remove something at end of a list
- Remove is used to remove by name
- Del keyword is also good practice
- Clear Method is used to vanish all
- Count method gives the number of occurrence of a list item
- Index method gives the index of a certain list item
- In method is used to verify that a certain item is present in the list
- Sorted gives a new list while sort method modifies because it gets implement over the object list
- Map function gives off an iterable
- Map function uses a lambda function to iterate over lists to produce new lists
- Lambda function is used to reference other functions
- Filter function uses a lambda function to iterate over lists to produce new lists as per certain criteria of filtering list items

- List Comprehensions are python-only and are the best practices to be used instead of Map & Filter functions
- List Comprehension works on one list
- Zip function is used to comprehend two lists in an expanded fashion of tuples
- Stack resembles a stack of real world books
- Stack works on the principle of LIFO → Last in- First out
- Which means if you have a stack of books around, the last book you placed on the top of all books, you can remove that very first out
- A perfect real world example of Stack is the browsing history of your browser
- We can implement Stack in python by using a List
- An empty list is a falsy value
- Queue works on the principle of FIFO → First in-First out
- It resembles a real world queue or line
- Lists are used to implement queues
- Queue has a limitation regarding memory & performance due to the left shifting of each list item when it comes to large data, so we use deque to cope up with all the odds
- Deque auto shifts the list items
- A tuple is basically a read only list
- Tuple function is used to convert a list into a tuple
- We cant modify a tuple however we can concatenate two tuples and multiply a tuple for generation
- A general rule of thumb is we only use tuples in real world scenarios when we try to avoid accidental mishandling of data in software
- When defining a tuple you can exclude the parenthesis
- Swapping variables in python is actually unpacking tuples under the hood
- Arrays perform a bit faster when large collection of numbers is involved as compared to lists
- In Arrays all the items should have same data type unlike lists
- Typecode is a string of single character which determines data type of your array
- Sets is data structure which is actually a collection of items with no duplicates
- Sets are used in powerful mathematical use-cases
- Sets are denoted by curly braces {}
- Sets don't support indexing properties
- Dictionary is actually a collection of Key-Value pairs
- A Real world example of dictionary is the phone book in which we have name of a person as a key and his/her contact details as value
- We can use comprehensions with lists, sets and dictionaries

- Comprehensions can be used instead of 1- Declaring 2- looping 3- Creating
- Tuple Comprehension expressions are basically Generators or Generator objects
- We use generators when we have an infinite stream of data
- Generator has a size of only 120 bytes
- Lists stores all the items whereas generator just spill out a new value at each iteration that's why we can't access to all the items working on
- We use unpacking variable to take out individual items from any iterable (*unpacking)
- Pretty printing is a useful module for print purposes

- An Exception is kind of error which terminates the execution of a program, more specifically this program threw an exception error
- As a programmer it's your job to handle all these exceptions and avoid your program from crashing
- Try-Except block is used to handle all exceptions
- We can handle a number of exceptions by having several exception clauses
- Similar Except clauses execute only once
- We use a last exception clause "Finally" to release all the stuff ranging from closing files, database connections, network ports etc
- It's used for cleaning up the memory and release external resources
- The **With** statement automatically releases all the external resources
- With and finally clause are alternatives
- Raising exceptions can be costly and not a good practice often
- Using Timeit module we can find the execution time of a python code
- Pass statement is used only to fill the total (No Task)
- Avoiding raising exceptions

- Class is a blue print for creating new objects
- Every object (int, string etc) in python has a class which produces that objects
- An Object is an instance/example of a class
- Human is a class & Mosh, Brad etc are objects
- Naming convention for classes is CamelCase or Pascal
- All functions in our classes should have atleast one parameter called as self
- isinstance is a method to find the instance/object of a class
- Constructor is a special method which is called when we create a new object
- Magic method is known as constructor → `__init__`
- Self is the reference to the current object
- Objects also have attributes which are basically variables containing data about a specific object
- Like javascript the objects in python are dynamic
- Class level attributes are shared across all instances/objects of that type
- Instance level attributes are limited to that instance or object only
- Factory method is called factory as it acts as a factory of manufacturing or creating object
- Design pattern → Factory method helps in initializing the values which are complex and helps design clean code
- Cls is the default parameter when we define a class method
- @classmethod is called as decorator and it is used to enhance the behavior of an object or function
- Magic methods are called automatically by python interpreter
- `__str__` magic method is used when we want to convert an object to string
- `__gt__` Defines behavior for the greater-than operator
- We do have Normal arithmetic operators like `__add__`
- Data structures are also known as container types
- We can design our own custom container types
- We use private members in a class to avoid accidental access outside
- Private members can be formed by prefixing `__private`
- The aim of private members is not security rather it's a warning
- We can still access a private member by using `__dict__` magic method which holds all the attributes in this class
- Pythonic is referred to any best practice in which full potential of python is used unlike unpythonic
- A Property is an object that sits in front of an attribute and allows us get and set values of that attribute
- A property looks like a proper attribute from outside but internally has 2 methods which we call

- Getters
- Setters
- @Property decorator
- We want our classes and objects to be minimal with minimum number of functions or methods exposed to the outside
- DRY (Don't Repeat Yourself)
- Inheritance is the mechanism which allows us to define the common behaviors or common functions in one class and then inherit that in other classes
- Parent-Base | Child – Sub
- Object class is the base class and every class we declare in python directly or indirectly is derived from the Object class
- IsSubClass method is used to verify an object is a sub class of other
- Method overriding is replacing method of base class in child class
- We can avoid method overriding by the use of super() function to access and change base class
- Inheritance is a good thing it helps avoid code repetition and allows code reusability
- However, too much of a good thing is a bad thing → Too much inheritance between classes causes complexity in software
- Inheritance abuse → multi level inheritance → Avoid it → Just upto 2-3 levels
- If you don't handle the multiple inheritance properly you will be going to invite all sort of bugs
- Flyer + Swimmer → FlyerFish Example of Multiple Inheritance
- We can make custom error exception we can derive it from base exception class in python
- A good example of inheritance in real world is handling data from a data stream, may be from file on hard drive, a process on memory or a network stream
- An abstract base is like a half baked cookie, it provides some common code to it's derivatives
- Abc is AbstractBaseClass
- Abstract and Concrete classes
- Abstract base classes don't allow instances
- Concrete base classes allow instances to be developed
- Polymorphism → Poly → Many | Morps → Forms
- Polymorphism is usage of a method in several forms
- Example: Draw() is being used in different forms as draw(Interface), draw(toolbar) etc
- Duck Typing → If it walks like a duck & quacks like a duck! It is a duck

- Duck typing is supported by python because it's dynamically typed language & it doesn't check the type of object, it only looks for the existence of certain method is available or not
- We can easily extend the functionality of built-in data types such as lists & strings etc.
- We use classes to build data and functionality in one Unit
- `id()` function provides the address of memory location where an object is stored
- Those classes which do not have any behavior/functionality are referred to as "Data Classes"
- If you are working with Data Classes it's better practice to use `NamedTuples` instead of classes

- In real world scenarios, we don't put all code in one file instead we have folders structures hierarchy and several files
- SuperMarket → Example of Sections → Edibles, Drinks etc
- A module is a file which contains some python code
- Naming conventions for files or modules is same as naming variables
- From sales here we just have the name and not the extension like sales.py
- If we press Ctrl + Space we will be able to see which objects the other module contains
- Don't blindly import all objects from a module using asterisk *, import only those which you need
- We can import a whole module or some objects from that module, there is no best practice and no performance issues as if we need only a single object, still the whole module needs to be loaded
- __Pycache__ includes the (cpython file) compiled versions of all the module we import into this program
- Python stores this compiled version to speed up module loading speed (Note: not the performance of application)
- Python checks the compatibility of compiled file as it's up to date or not by date etc
- Cpython file contains the python byte code
- Python doesnot create cache of the main file which is the entry point to that program which we load via command line
- When we import a certain module, python searches for it in all the program directories (the folders we have made) and all the default directories python has made during installation + if any new framework, package or library has been installed
- We can use sys → path to print all the directories
- A package is a container for one or more python modules
- We create __init__.py to declare a folder as a package
- A package is mapped to a folder and a module is mapped to a file
- Sub-Package is a sub-folder
- In intra package references we use absolute and relative ways of references
- Best Practice is absolute way
- The Built-in dir function, with dir we can get the list of methods and attributes defined in an object
- The name of module which starts our program is always __main__
- With this code you can use your modules as scripts

```
if __name__ == "__main__":
    print("Sales started")
    calc_tax()
```

- Python standard library contains a huge bulk of pre-built features or modules which we can easily incorporate into our real world projects
- In a raw string, a back slash is not considered as an escape character
- Path class is the foundation for working with paths and directories

```
from pathlib import Path

path = Path("ecommerce/__init__.py")
path.exists()
path.is_file()
path.is_dir()
print(path.name)
print(path.stem)
print(path.suffix)
print(path.parent)
path = path.with_suffix(".txt")
print(path)
```

-
- Posix Path (for debian based OS including MacOS) & Windows Path

```
from pathlib import Path

path = Path("ecommerce")
# path.exists()
# path.mkdir()
# path.rmdir()
# path.rename("ecommerce2")

paths = [p for p in path.iterdir() if p.is_dir()]
py_files = [p for p in path.glob("*.py")]
print(py_files)
```

-
- UNIX Epoch time refers to the start of time on a computer, on unix based platforms it's Jan,1, 1970
- We have time library for time related things, pathlib for paths etc
- We have shutil for high level operations such as copying and moving files and folders in easy way
- Open method for reading/writing a file is deprecated, the best practice is to use the path.read() method, because in this case we don't have the worries of closing the files using with or manually
- CSV, a comma separated file is simplified spreadsheet file stored as plain text
- We use csv module for working with CSV file, the possibilities are endless!
- JSON → JavaScriptObjectNotation is a human readable data format, trending tech
- JSON files are actually "An Array of Dictionaries"
- SQLite is a very light weight database we use for small-medium sized apps
- We have Time, DateTime & TimeDelta → Durations | for time chores
- Random library is useful in guesses etc , possibilities are endless!
- Webbrowser library is used for opening web browser
- In real world scenarios, companies have email templates for various purposes

- Email.MIME.multipart,image,text for several ways of emails
- We have smtplib for smtp servers connection for sending emails
- Sys.argv are used for command line arguments
- Subprocess module is used to explore and run other programs and processes

- Pypi Python package index is a repository of pre-made python features which you can add to your projects, just like NPM for Javascript
- Pip is a tool for installing, uninstalling and upgrading python package indexes
- Pip is developed independently
- Semantic version 20.12.9 means major, minor and patches or bug fixes
- We need virtual environments of pip versions to have different version of any pypi running side by side
- Venv or virtual environment directory is an isolated directory
- Pipenv combines pip and virtual environment into a combined tool chain for ease
- Pipenv is the dependency manager in python as npm to JS
- Pipfile and pipfile.lock are created by pipenv in the isolated virtual environment
- Pipfile is used to install the dependences if on another machine
- Pipfile.lock is used to install the dependences with same versions on another machine
- We can upload our own Python packages to pypi
- Configuration files are needed to upload a package to pypi
- Intellisense is actually displaying the documentation of a pypi
- DocString is a special documentation style in python
- Pydoc is a utility for reading documentation of any module