

**SIX WEEKS SUMMER TRAINING**

**REPORT**

on

***OBJECT ORIENTED PROGRAMMING USING PYTHON***

Submitted by

**Atharva Santosh Todmal**

**Registration No.**

**12010145**

**Programme Name: B. Tech CSE**

Under the Guidance of

**School of Computer Science & Engineering Lovely Professional University, Phagwara**

(June-July, 2022)

**DECLARATION**

I hereby declare that I have completed my four weeks summer training at Board Infinity from May 2022 to July 2022 under the guidance of Sunil Khurana. I have declared that I have worked with full dedication during these four weeks of training and my learning outcomes fulfill the requirements of training for the award of degree of B. Tech CSE, Lovely Professional University, Phagwara.

(Signature of student) Name of Student Registration no:

Date: 17 – 07 - 2022

**ACKNOWLEDGEMENT**

I would like to extend my sincere gratitude to everyone who gave me the chance to finish this report. I want to express my sincere appreciation to my mentor, whose contribution of inspiring ideas and support enabled me to organize my project, particularly in producing this report. I want to thank my school, Lovely Professional University, for giving me the chance to study about this fascinating subject. I would also want to thank my professor for all of their help throughout this "advance python programming" course. I want to express my sincere gratitude to all of my fans once more.

**SUMMER TRAINING CERTIFICATE (FROM TRAINING INSTITUTE)**

****

**TABLE OF CONTENTS**

Enrollment in local colleges, 2005

|  |  |
| --- | --- |
| Content | Page |
|  |  |
| INTRODUCTION | 6-7 |
| Technology Learnt | 8-10 |
| Reason For choosing this Technology | 10 |
| Profile of the Problem | 11 |
| Existing System | 11 |
| Problem Analysis | 11 |
| Software Requirement Analysis | 11 |
| Design  Implementation  Learning outcome of Training  Gantt Chart  Bibliography | 12  13-14  15  15  16 |
|  |  |

**INTRODUCTION**

Python is a widely used high-level, general-purpose, interpreted, dynamic

programming language. Its design philosophy emphasizes code readability, and

its syntax allows programmers to express concepts in fewer lines of code than

would be possible in languages such as C++ or Java. The language provides

constructs intended to enable clear programs on both a small and large scale.

Python supports multiple programming paradigms, including object-oriented,

imperative and functional programming or procedural styles. It features a

dynamic type system and automatic memory management and has a large and

comprehensive standard library. Python interpreters are available for

installation on many operating systems, allowing Python code execution on a

wide variety of systems.

Python is a widely used high-level, general-purpose, interpreted, dynamic

programming language. Its design philosophy emphasizes code readability, and

its syntax allows programmers to express concepts in fewer lines of code than

would be possible in languages such as C++ or Java. The language provides

constructs intended to enable clear programs on both a small and large scale.

Python supports multiple programming paradigms, including object-oriented,

imperative and functional programming or procedural styles. It features a

dynamic type system and automatic memory management and has a large and

comprehensive standard library. Python interpreters are available for

installation on many operating systems, allowing Python code execution on a

wide variety of systems.

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry. Python Programming Language is very well suited for Beginners, also for experienced programmers with other programming languages like C++ and Java.

**Scripting language**

All scripting languages are programming languages. The scripting language is basically a language where instructions are written for a run time environment. They do not require the compilation step and are rather interpreted. It brings new functions to applications and glue complex system together. A scripting language is a programming language designed for integrating and communicating with other programming languages.

There are many scripting languages some of them are discussed below:

* **Bash:** It is a scripting language to work in the Linux interface. It is a lot easier to use bash to create scripts than other programming languages. It describes the tools to use and code in the command line and create useful reusable scripts and conserve documentation for other people to work with.
* **Node js:** It is a framework to write network applications using **JavaScript**. Corporate users of Node.js include IBM, LinkedIn, Microsoft, Netflix, PayPal, Yahoo for real-time web applications.
* **Ruby:** There are a lot of reasons to learn Ruby programming language. Ruby’s flexibility has allowed developers to create innovative software. It is a scripting language which is great for web development.
* **Python:** It is easy, free and open source. It supports procedure-oriented programming and object-oriented programming. Python is an interpreted language with dynamic semantics and huge lines of code are scripted and is currently the most hyped language among developers.
* **Perl:** A scripting language with innovative features to make it different and popular. Found on all windows and Linux servers. It helps in text manipulation tasks. High traffic websites that use Perl extensively include priceline.com, IMDB.

**Object-oriented programming**

OOP is a method of structuring a program by bundling related properties and behaviours into individual **objects**. In this tutorial, you’ll learn the basics of object-oriented programming in Python.

Conceptually, objects are like the components of a system. Think of a program as a factory assembly line of sorts. At each step of the assembly line a system component processes some material, ultimately transforming raw material into a finished product.

An object contains data, like the raw or pre-processed materials at each step on an assembly line, and behaviour, like the action each assembly line component performs.

**History of Python**

Python was conceived in the late 1980sby [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) at [Centrum Wiskunde & Informatica](https://en.wikipedia.org/wiki/Centrum_Wiskunde_%26_Informatica) (CWI) in the [Netherlands](https://en.wikipedia.org/wiki/Netherlands) as a successor to the [ABC programming language](https://en.wikipedia.org/wiki/ABC_(programming_language)), which was inspired by [SETL](https://en.wikipedia.org/wiki/SETL), capable of [exception handling](https://en.wikipedia.org/wiki/Exception_handling) and interfacing with the [Amoeba](https://en.wikipedia.org/wiki/Amoeba_(operating_system)) operating system. Its implementation began in December 1989. Van Rossum shouldered sole responsibility for the project, as the lead developer, until 12 July 2018, when he announced his "permanent vacation" from his responsibilities as Python's "[benevolent dictator for life](https://en.wikipedia.org/wiki/Benevolent_dictator_for_life)", a title the Python community bestowed upon him to reflect his long-term commitment as the project's chief decision-maker. In January 2019, active Python core developers elected a five-member Steering Council to lead the project.

Python 2.0 was released on 16 October 2000, with many major new features. Python 3.0, released on 3 December 2008, with many of its major features [backported](https://en.wikipedia.org/wiki/Backporting) to Python 2.6.x and 2.7.x. Releases of Python 3 include the 2to3 utility, which automates the translation of Python 2 code to Python 3.

Python 2.7's [end-of-life](https://en.wikipedia.org/wiki/End-of-life_(product)) was initially set for 2015, then postponed to 2020 out of concern that a large body of existing code could not easily be forward-ported to Python 3. No further security patches or other improvements will be released for it. With Python 2's [end-of-life](https://en.wikipedia.org/wiki/End-of-life_(product)), only Python 3.6.xand later were supported. Later, support for 3.6 was also discontinued. In 2021, Python 3.9.2 and 3.8.8 were expedited as all versions of Python (including 2.7) had security issues leading to possible [remote code execution](https://en.wikipedia.org/wiki/Remote_code_execution)[[56]](https://en.wikipedia.org/wiki/Python_(programming_language)#cite_note-56) and [web cache poisoning](https://en.wikipedia.org/wiki/Cache_poisoning).

In 2022, Python 3.10.4 and 3.9.12 were expedited and so were older releases including 3.8.13, and 3.7.13 because of many security issues in 2022. Python 3.9.13 is the latest 3.9 version, and from now on 3.9 (and older; 3.8 and 3.7) will only get security updates.



The designer of Python, [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum), at [OSCON](https://en.wikipedia.org/wiki/O%27Reilly_Open_Source_Convention) 2006

**TECHNOLOGY LEARNT**

**1. Introduction to Python**

Learn how to install Python, distinguish between important data types and use basic features of the Python interpreter, IDLE.

**2. Using Variables in Python**

Learn about numeric, string, sequence and dictionary data types and relevant operations while practicing Python syntax.

**3. Basics of Programming in Python**

Learn how to write programs using conditionals, loops, iterators and generators, functions and modules and packages.

**4. Principles of Object-oriented Programming (OOP)**

Learn about the important features of Object-oriented Programming while using Classes and Objects, two main aspects of the OOP paradigm.

**5. Connecting to SQLite Database**

Learn about relational databases while learning how to store and retrieve data from an SQLite database through Python.

**6. Developing a GUI with PyQT**

Learn how to install PyQt5 toolkit, Qt Designer and create a graphical user interface using common widgets and menu systems.

**7. Application of Python in Various Disciplines**

Learn about various resources to extend your learning for the Python programming language.

**Application of Python**

**1. Web and Internet Development**

Python lets you develop a web application without too much trouble. It has libraries for internet protocols like HTML and XML, JSON, e-mail processing, FTP, IMAP, and easy-to-use socket interface. Yet, the package index has more libraries:

* **Requests** – An HTTP client library
* **BeautifulSoup** – An HTML parser
* **Feedparser** – For parsing RSS/Atom feeds
* **Paramiko** – For implementing the SSH2 protocol
* **Twisted Python** – For asynchronous network programming

### 2. Desktop GUI Applications

Most binary distributions of Python ship with Tk, a standard GUI library. It lets you draft a user interface for an application. Apart from that, some toolkits are available:

* wxWidgets
* Kivy – for writing multitouch applications
* Qt via pyqt or pyside

### 3. Science and Numeric Applications

This is one of the widespread applications of Python programming. With its power, it comes as no surprise that Python finds its place in the scientific community. For this, we have:

* **SciPy** – A collection of packages for mathematics, science, and engineering.
* [***Pandas***](https://data-flair.training/blogs/pandas-tutorials-home/)– A data-analysis and -modeling library
* **IPython** – A powerful shell for easy editing and recording of work sessions. It also supports visualizations and parallel computing.
* Also, **NumPy** enables us to deal with complex numerical calculations.

### 4. Software Development

Software developers make use of Python as a support language. They use it for build-control and management, testing, and for a lot of other things:

* **SCons** – for build-control
* **Buildbot, Apache Gump** – for automated and continuous compilation and testing
* **Roundup, Trac** – for project management and bug-tracking.
* The roster of Integrated Development Environments

### 5. Education

Thanks to its simplicity, brevity, and large community, Python makes for a great introductory programming language. Applications of Python programming in education has a huge scope as it is a great language to teach in schools or even learn on your own.

### 6. Business

Python is also a great choice to develop **ERP** and **e-commerce** systems:

* **Tryton** – A three-tier, high-level, general-purpose application platform.
* **Odoo** – A management software with a range of business applications. With that, it’s an all-rounder and forms a complete suite of enterprise-management applications in-effect.

**Scope in Real World**

The Future Scope of a Python Developer is very bright the world is getting digitized. Python has become the core language as far as the success of these technologies is concerned. Let’s dive into the technologies which use python as a core element for research, production and further developments. The future is all about automating processes and utilizing the heaps of data to make intelligent decisions. This puts to the forefront technologies such as artificial intelligence (AI), machine and deep learning, Internet of Things (IoT), etc.

**REASON FOR CHOSING THIS TECHNOLOGY**

Python is one of the most user-friendly programming languages since it has a simple syntax and isn't overly complex, putting more of a focus on natural language. Python is one of the easiest programming languages to learn and use, making it possible to write and execute scripts quickly compared to other programming languages.

Python is a free and open-source programming language that makes it simpler for developers to obtain supporting libraries, modules, and other tools. The cost of the programming language is within reach for both developers and startup companies of all sizes.

The integration capabilities that make it simpler to construct web services are typically adored by Python developers. It provides strong control and is compatible with markup languages used for app development. Developers that want to learn Python may simply enroll in the certification program.

Python is a popular programming language because it makes it simple and quick for developers to create apps. Its simple syntax and concise code make it easy to use for both experienced users and new users.

With its wide support network and simple-to-integrate features, Python is incredibly simple to use. Because the programming language is simple to read and grasp, beginners will find it simpler to begin their coding careers. Python is a fantastic programming language for developing developers' abilities. For developers, this provides substantial reference and support libraries. Additionally, Python has a supportive community that is prepared to assist in case of difficulties.

The object-oriented language model is the basis of the Python programming language.

It can effectively manage the web development process in this way. Through its unit testing structure, the language has the ability to handle text.

A safe, reliable, quick, and dynamic web application is produced when all these properties are integrated with web development.

Without worrying about compatibility, you may create the backend in Python and create the interface and render it in whatever language. Given that Python is a well-known highly functional programming language, integrating it with other frameworks and languages is simple for developers.

**PROFILE OF THE PROBLEM**

Since, HuffPost Pollster will provide the information for the polls. After doing this project, I urge you to visit and play around with the political data there because there are some very great resources there. There are some issues with the election, such as transparency in the outcomes and data that is biassed in nature and even doesn't provide greater preference in the column or data's characteristic. As a result, you can observe that several of the implementation-related variables are biassed.

**EXISTING SYSTEM**

The polling data was obtained from HuffPost Pollster. We're going to import some data from the web using the requests module. See the documentation here for additional details about requests.

Additionally, StringIO will be used to manipulate the csv data we obtained from HuffPost. Using the file API, StringIO offers a practical way to interact with text in memory.

**PROBLEM ANALYSIS**

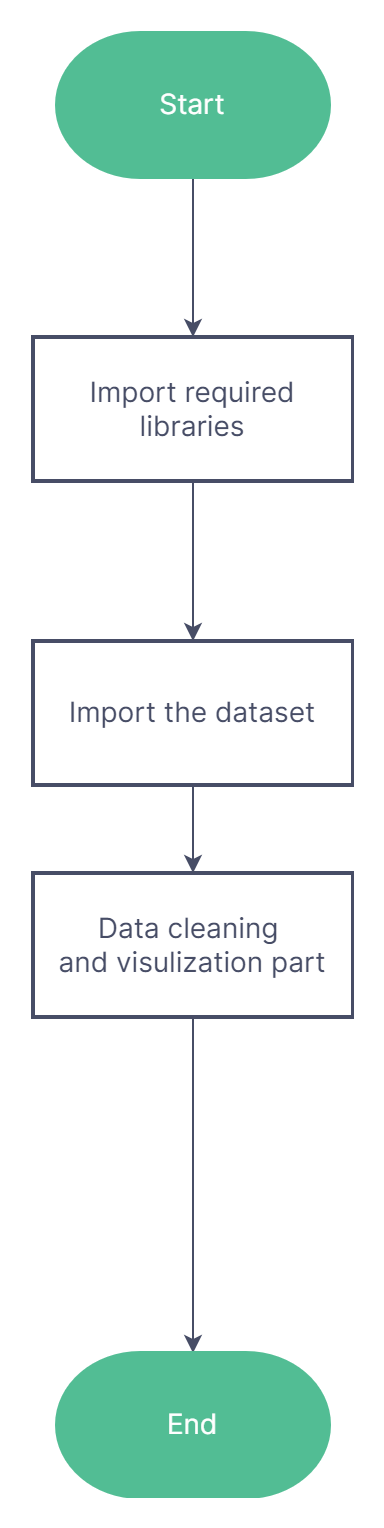
It would be wise to bear this in mind given the outcomes of the implementation phase, which were overall fairly neutral but still leaning towards Democratic Affiliation. So let's check whether grouping the data by population colour provides any more insight. The poll findings should, ideally, be a good depiction of the populations surveyed because other graphs have been produced that show a strong showing of potential voters and Registered Voters. Considering the uncertain component in particular, let's take a second, short look at the DataFrame that it provides after producing its graph to see how close these pools appear to be. As a consequence, we conducted a statistical analysis for the election outcomes and identified certain biassed and unbiased variables.

The suggested system or data visualization is quite simple and was just constructed to demonstrate a conceptual grasp of Python concept. It is also extremely viable because it does not call for any additional hardware or internet connectivity.

**SOFTWARE REQUIREMENT ANALYSIS**

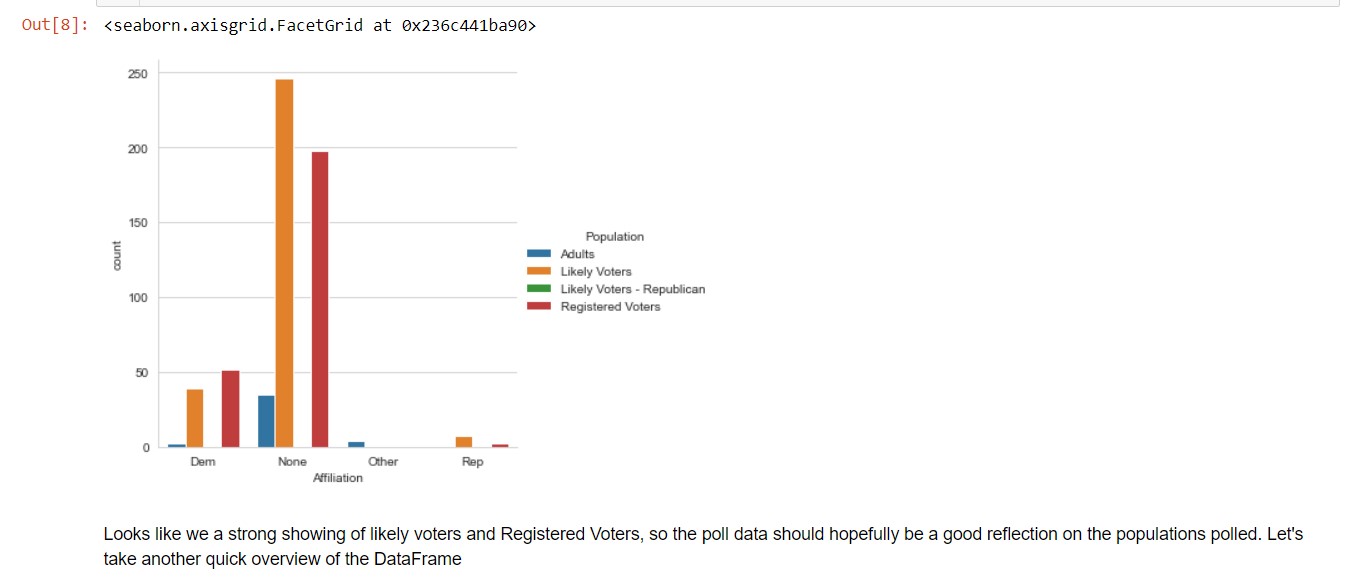
1. Jupyter Notebook
2. Windows 10, 11

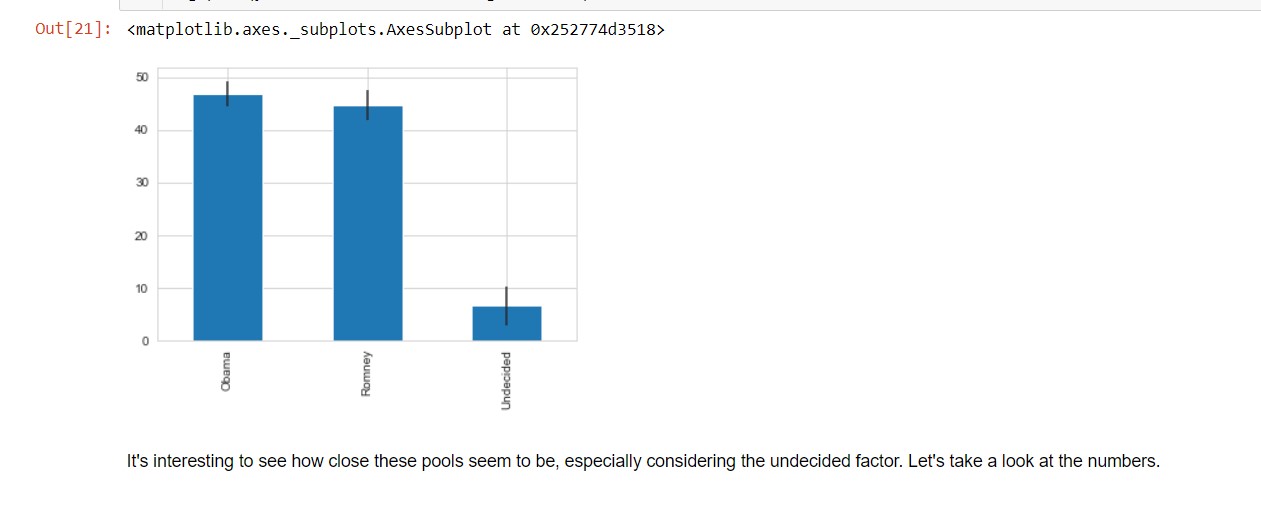
**DESIGN**

****

**IMPLEMENTATION**







**LEARNING OUTCOME FROM TRAIINING**

Although I learned pyhton Programming during the course, the explanation of classes and objects creating a class, dealing with object composition, encapsulation, inheritance, overriding object methods, defining and using functions and modules, working with recursion, the fundamentals of dealing with lists, a list of lists, tuples, dates, pandas and data visualization part, and times, how to get started with dictionaries, educational software, and exercises.

One of the first Python programmes you should learn is pandas since it's open source, simple to use, and allows you to work with a lot of data. Data aggregation, pivoting, flexible time series flexibility, and other features are all made possible by its quick and effective data manipulation. To help students write scripts for data analysis and manipulation, we taught them the fundamentals of Python, including its syntax, functions, and packages. We also covered the types of variables and their features, basic operators and statements, loops, and the three main data science packages: NumPy, Pandas, and Matplotlib.

**GANTT CHART**

**BIBLIOGRAPHY**

* [**https://en.wikipedia.org/wiki/Python\_(programming\_language)**](https://en.wikipedia.org/wiki/Python_(programming_language))
* [**https://docs.python.org/3/library/index.html**](https://docs.python.org/3/library/index.html)
* [**https://www.w3schools.com/python/**](https://www.w3schools.com/python/)
* [**https://www.tutorialspoint.com/python/index.htm**](https://www.tutorialspoint.com/python/index.htm)