

# INTERNSHIP REPORT



भारतीय प्रौद्योगिकी संस्थान हैदराबाद  
Indian Institute of Technology Hyderabad

UNDER THE SUPERVISION OF:  
**DR. ABHINAV KUMAR**  
**MR. NAGABHUSHAN**  
**ESWARA**

**PRATIKSHA GUPTA (B.TECH)**  
**ELECTRONICS AND COMMUNICATION ENGINEERING**  
**DEPARTMENT**  
**ATAL BIHARI VAJPAYEE GOVT. INSTITUTE OF ENGG. AND**  
**TECHNOLOGY [SHIMLA, H.P.]**  
**JUNE 2018**

1<sup>st</sup> June to 30<sup>th</sup> June

# MACHINE LEARNING WITH PYTHON

**Machine learning has become an integral part of many commercial applications and research projects, but this field is not exclusive to large companies with extensive research teams. With all the data available today, machine learning applications are limited only by our imagination.**

**Machine learning is the science of getting computers to act without being explicitly programmed. In the past decade, machine learning has given us self-driving cars, practical speech recognition, effective web search, and a vastly improved understanding of the human genome. Machine learning is so pervasive today that we probably use it dozens of times a day without knowing it. Many researchers also think it is the best way to make progress towards human-level AI. In this class, we will learn about the most effective machine learning techniques, and gain practice implementing them and getting them to work for our self. More importantly, we'll learn about not only the theoretical underpinnings of learning, but also gain the practical know-how needed to quickly and powerfully apply these techniques to new problems. This content provides a broad introduction to machine learning, datamining, and statistical pattern recognition. Topics include: (i) Supervised learning (parametric/non-parametric algorithms, support vector machines, kernels, neural networks). (ii) Unsupervised learning (clustering, dimensionality reduction, recommender systems, deep learning).**

Python is an [interpreted high-level programming language](#) for [general-purpose programming](#). Created by [Guido van Rossum](#) and first released in 1991, Python has a design philosophy that emphasizes [code readability](#), notably using [significant whitespace](#). It provides constructs that enable clear programming on both small and large scales.

Python features a [dynamic type](#) system and automatic [memory management](#). It supports multiple [programming paradigms](#), including [object-oriented](#), [imperative](#), [functional](#) and [procedural](#), and has a large and comprehensive [standard library](#).

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python was designed to be highly readable. It uses English keywords frequently where as other languages use abbreviations, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – we can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

## Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** – we can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

## **NumPy:-**

**NumPy is the fundamental package for scientific computing with Python. It contains among other things:**

- **a powerful N-dimensional array object**
- **sophisticated (broadcasting) functions**
- **tools for integrating C/C++ and Fortran code**
- **useful linear algebra, Fourier transform, and random number capabilities**

**Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.**

01

Lorem ipsum dolor sit amet, harum patrioque cu vim.

02

Lorem ipsum dolor sit amet, harum patrioque cu vim.

03

Lorem ipsum dolor sit amet, harum patrioque cu vim.

## Heading 1

You can easily change the formatting of selected text in the document by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or the format that you specify directly.

### Heading 2

To change the overall look of your document, choose a new Theme element on the Page Layout tab. To change the looks available in the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.

### Heading 3

On the Insert tab, the galleries include items that are designed to coordinate with the overall look of your document. You can use these galleries to insert headers, footers, lists, cover pages, and other document building blocks. When you create pictures, charts, or diagrams, they also coordinate with your document look.

You can easily change the formatting of selected text in the document by choosing a look for the selected text from the Quick Styles gallery on the Home tab. You can also format text directly by using the other controls on the Home tab. Most controls offer a choice of using the look from the current theme or the format that you specify directly.

To change the overall look of your document, choose new Theme elements on the Page Layout tab. To change the look of the Quick Style gallery, use the Change Current Quick Style Set command. Both the Themes gallery and the Quick Styles gallery provide reset commands so that you can always restore the look of your document to the original contained in your current template.