

CS-115 - Fundamentals of Python and IoT

CourseTitle	Fundamentals of Python and IoT	CourseCode	CS-115
SEMESTER/ YEAR	I Year	CourseGroup	
Teaching Scheme	4:1:0	Credits	3
Methodology	Lecture+ Tutorial	Total Contact Periods	150
CIE	60 Marks	SEE	40 marks

PREREQUISITES:

Basic knowledge of Mathematics

COURSE OUTCOMES:

At the end of the course, the students will have the ability to

CO1	Develop simple Python programs using basic syntax, data types, operators, and input/output statements for elementary automation tasks.
CO2	Apply control structures and functions to design modular, efficient, and error-free Python programs.
CO3	Use Python data structures and file handling techniques to store, process, and log data for automation applications.
CO4	Explain the fundamentals, architecture, and components of IoT systems relevant to mechanical engineering applications.
CO5	Interface sensors and actuators with embedded platforms using Python to develop basic IoT-based monitoring and control systems.
CO6	Analyze IoT applications in mechanical engineering domains such as manufacturing, energy, automotive, and predictive maintenance.

COURSE CONTENTS

UNIT–I: Introduction to Python & Programming Basics (

DURATION:25 Hrs

- Features of Python
- Applications of Python in automation and engineering
- Python installation and development environments (IDEs)
- Python syntax and keywords
- Variables and data types
- Operators and expressions
- Input and output operations
- Simple Python programs for automation

UNIT–II: Control Structures & Functions

DURATION:25 Hrs

- Conditional statements (if, if–else, elif)
- Looping statements (for, while, nested loops)
- Control statements (break, continue, pass)
- Functions and modular programming
- Lambda functions

- Error handling concepts
- Debugging techniques
- Code optimization practices

UNIT–III: Data Structures & File Handling

DURATION:25 Hrs

- Strings and string operations
- Lists, tuples, sets, and dictionaries
- Operations on data structures
- File handling concepts
 - Text files
 - CSV files
- Exception handling
- Data logging techniques for automation systems

UNIT–IV: Fundamentals of IoT and Smart Devices

DURATION:25 Hrs

- Introduction to Internet of Things (IoT)
- Basic architecture of IoT systems
- Sensors and actuators used in mechanical systems
- Embedded platforms for IoT
- Introduction to IoT communication technologies

UNIT–V: Python for Hardware & IoT Integration

DURATION:25 Hrs

- Introduction to Python GPIO programming
- Serial communication using Python (UART, USB)
- Interfacing sensors and actuators using Python
- Python programming with Raspberry Pi
- Development of simple Python-based IoT applications

UNIT–VI: IoT Applications in Mechanical Engineering

DURATION:25 Hrs

- Industrial Internet of Things (IIoT)
- Condition monitoring and predictive maintenance
- IoT applications in energy and utilities
- IoT in automotive and transportation systems
- IoT in smart factories and safety systems
- IoT challenges and security issues
- Future trends in IoT

Learning Outcomes:

UNIT–I: Introduction to Python & Programming Basics

After completing this unit, students will be able to:

1. Define Python and explain its key features.
2. Describe the advantages of Python in automation and engineering applications.
3. Identify real-world applications of Python in mechanical engineering.
4. Install Python on different operating systems.
5. Select and use suitable Python IDEs.
6. Explain Python program structure and syntax rules.
7. Declare and use variables correctly.
8. Identify different Python data types.
9. Apply type conversion techniques.
10. Use arithmetic, relational, and logical operators.
11. Evaluate expressions using operators and precedence rules.
12. Perform formatted input and output operations.
13. Write simple Python programs for basic automation tasks.
14. Interpret and correct basic syntax errors.

UNIT–II: Control Structures & Functions

After completing this unit, students will be able to:

1. Explain the need for control structures in programming.
2. Use if, if-else, and nested conditional statements.
3. Apply elif ladders for multiple decision making.
4. Implement for and while loops.
5. Use break, continue, and pass statements.
6. Design programs involving nested loops.
7. Explain the concept of functions in Python.
8. Define and call user-defined functions.
9. Pass arguments and return values from functions.
10. Apply modular programming concepts.
11. Use lambda functions for simple operations.
12. Identify common programming errors.
13. Apply debugging techniques to locate errors.
14. Improve program efficiency using code optimization practices.

UNIT–III: Data Structures & File Handling

After completing this unit, students will be able to:

1. Explain the importance of data structures in programming.
2. Perform string operations and manipulations.
3. Create and modify lists.
4. Use tuples and explain their immutability.

5. Apply set operations.
6. Create and manipulate dictionaries.
7. Choose appropriate data structures for a given problem.
8. Explain file handling concepts in Python.
9. Read data from and write data to text files.
10. Perform file operations on CSV files.
11. Handle runtime errors using exception handling.
12. Use try, except, else, and finally blocks.
13. Develop programs for data logging applications.
14. Store and retrieve automation data using files.

UNIT–IV: Fundamentals of IoT and Smart Devices

After completing this unit, students will be able to:

1. Define Internet of Things (IoT).
2. Explain the evolution and need for IoT.
3. Describe the basic architecture of IoT systems.
4. Identify various components of an IoT system.
5. Explain the role of sensors in IoT.
6. Explain the role of actuators in IoT.
7. Identify sensors used in mechanical engineering applications.
8. Identify actuators used in mechanical systems.
9. Explain embedded platforms used for IoT.
10. Compare common embedded platforms for IoT.
11. Explain basic IoT communication concepts.
12. Identify wired and wireless communication technologies used in IoT.
13. Explain data flow in an IoT system.
14. Illustrate simple IoT system block diagrams.

UNIT–V: Python for Hardware & IoT Integration

After completing this unit, students will be able to:

1. Explain the role of Python in hardware interfacing.
2. Identify GPIO pins of embedded platforms.
3. Configure GPIO pins for input and output operations.
4. Control LEDs and relays using Python programs.
5. Explain serial communication principles.
6. Use UART and USB communication in Python.
7. Interface basic sensors using Python.
8. Read sensor data using Python programs.
9. Interface actuators using Python.
10. Explain the architecture of Raspberry Pi.
11. Develop Python programs using Raspberry Pi.
12. Integrate sensors with Raspberry Pi using Python.
13. Develop simple IoT-based monitoring systems.
14. Implement Python-based control applications.

UNIT–VI: IoT Applications in Mechanical Engineering

After completing this unit, students will be able to:

1. Define Industrial Internet of Things (IIoT).
2. Explain the importance of IoT in manufacturing industries.
3. Identify IoT applications in condition monitoring.
4. Explain predictive maintenance concepts.
5. Apply IoT concepts to energy management systems.
6. Identify IoT applications in utilities and power systems.
7. Explain IoT applications in automotive systems.
8. Describe IoT use in transportation systems.
9. Explain the role of IoT in smart factories.
10. Identify IoT-based safety and monitoring systems.
11. Analyze benefits of IoT in mechanical engineering.
12. Identify challenges in IoT implementation.
13. Explain security issues in IoT systems.
14. Discuss future trends in IoT technology.

CO–PO Mapping Table

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1:	3	2	1	2	-	–	1
CO2:	2	3	2	2	-	–	1
CO3:	2	3	2	3	-	–	1
CO4:	3	2	-	1	2	-	1
CO5:	2	3	3	3	2	1	1
CO6:	2	3	2	2	3	1	2

Reference Books

1. Internet of Things-A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759
3. Dr. SRN Reddy, Rachit Thukral and Manasi Mishra, “Introduction to Internet of Things: A practical Approach”, ETI Labs
4. Raj Kamal, “Internet of Things: Architecture and Design”, McGrawHills

Suggested E-learning references

1. <https://internetofthingsagenda.techtarget.com/>
2. <https://dzone.com/iot-developer-tutorials-tools-news-reviews>
3. <https://blog.bosch-si.com/>
4. <https://www.hackster.io/>
5. <https://www.libelium.com/>
6. <https://www.ibm.com/blogs/internet-of-things/>
7. <https://azure.microsoft.com/en-us/blog/topics/internet-of-things/>
8. <https://blog.arduino.cc/>
9. <https://www.raspberrypi.org/blog/>

State Board of Technical Education and Training, Telangana
Model Question paper
C-26; CS-115; FUNDAMENTALS OF PYTHON AND IOT
Mid Semester-I Examination

Time: 1 hour

Max.Marks: 20 Marks

PART-A

Answer **all** questions- Each Question carries **ONE**-mark

4x1 = 4 Marks

1. What is Python?
2. List any one feature of Python.
3. What is a conditional statement?
4. Name any one looping statement in Python.

PART-B

Answer two questions- Each Question carries **THREE** marks

2x3 = 6 Marks

- 5(a) Explain the importance of Python in automation applications.
OR
5(b) Describe different data types available in Python.
- 6(a). Explain the use of if-else statements with an example.
OR
6(b). Differentiate between for loop and while loop.

PART-C

Answer two questions- Each Question carries **FIVE** marks

2x5 = 10 Marks

- 7(a). Write a Python program to read two numbers and print their sum and difference.
OR
7(b). Develop a Python program to calculate the area of a circle.
- 8(a). Write a Python program to check whether a given number is even or odd.
OR
8(b) Develop a Python program to find the factorial of a number using a loop..

State Board of Technical Education and Training, Telangana
Model Question paper
C-26; CS-115; FUNDAMENTALS OF PYTHON AND IOT
Mid Semester-II Examination

Time: 1 hour

Max.Marks: 20 Marks

PART-A

Answer all questions- Each Question carries ONE-mark

4x1 = 4 Marks

1. What is a list in Python?
2. What is a dictionary?
3. What is IoT?
4. Define sensor.

PART-B

Answer two questions- Each Question carries THREE marks

2x3 = 6 Marks

5(a) Explain the difference between list and tuple.

OR

5(b) Describe any three string operations in Python.

6(a). Explain the basic architecture of an IoT system.

OR

6(b). Describe the role of sensors and actuators in IoT.

PART-C

Answer two questions- Each Question carries FIVE marks

2x5 = 10 Marks

7(a). Write a Python program to store and display elements of a list.

OR

7(b). Develop a Python program to read data from a text file and display it.

8(a). Draw and explain the block diagram of a simple IoT system.

OR

8(b) Explain the working of an IoT-based temperature monitoring system.

State Board of Technical Education and Training, Telangana
Model Question paper
C-26; CS-115; FUNDAMENTALS OF PYTHON AND IOT
Semester End Exam

Time: 2-hour

Max.Marks: 40 Marks

PART-A

Answer **all** questions- Each Question carries **ONE**-mark

8x1 = 8 Marks

1. What is meant by a variable in Python?
2. What is a lambda function?
3. What is GPIO?
4. What is UART?
5. What is Raspberry Pi?
6. What is IIoT?
7. What is predictive maintenance?
8. What is a smart factory?

PART-B

Answer all questions- Each Question carries **THREE** marks

4x3 = 12 Marks

9(a) Describe input and output operations in Python.

OR

9(b) Explain GPIO programming using Python.

10(a). Describe common types of programming errors.

OR

10(b). Explain IoT applications in condition monitoring.

11(a) Explain the role of Python in sensor interfacing.

OR

11(b) Describe the features of Raspberry Pi.

12(a) Describe the role of IoT in energy management.

OR

12(b) Describe safety applications of IoT in industries.

PART-C

Answer all questions- Each Question carries FIVE marks

4x5 = 20 Marks

13(a). Write a simple Python program demonstrating formatted input and output.

OR

13(b) Write a Python program to blink an LED using GPIO pins.

14(a). Write a Python program to generate the multiplication table of a given number.

OR

14(b) Explain the use of IoT for predictive maintenance in machines.

15(a) Explain interfacing of a temperature sensor with Raspberry Pi using Python.

OR

15(b) Develop a Python program to read sensor data and display it.

16(a) Explain IoT applications in automotive and transportation systems.

OR

16(b) Analyze challenges and security issues in IoT implementation.