Code No: 23CS6421

**PVP23**

**PVP Siddhartha Institute OF TECHNOLOGY**

**(Autonomous)**

**ADVANCED PYTHON PROGRAMMING**

**Duration: 3 Hours Max. Marks: 70**

Note: 1. This question paper contains two Parts A and B.

2. Part-A contains 10 short answer questions. Each Question carries 2 Marks.

3. Part-B contains 5 essay questions with an internal choice from each unit. Each

question carries 10 marks.

4. All parts of the Question paper must be answered in one place

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part - A**

10 x 2 = 20 Marks

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Blooms Level | CO |
| 1 a) | Describe list comprehensions and their advantages | L2 | CO1 |
| 1 b) | Define decorators and their use in Python. | L2 | CO1 |
| 1c) | Contrast universal functions (ufuncs) in NumPy? | L2 | CO1 |
| 1d) | Describe datatypes in NumPy. | L2 | CO1 |
| 1e) | Illustrate the difference between one-dimensional and two-dimensional arrays using NumPy. | L2 | CO1 |
| 1 f) | Explain how to read a CSV file into a Pandas Data Frame and retrieve the first 5 columns. | L2 | CO1 |
| 1 g) | Show the process of creating a simple client-server application using sockets. | L2 | CO1 |
| 1 h) | Define multithreading in Python? | L2 | CO1 |
| 1 i) | Explain eager execution mode in TensorFlow? | L2 | CO1 |
| 1 j) | Define the role of Tensors in TensorFlow? | L2 | CO1 |

**Course Coordinator Module Coordinator HOD**

**Part –B**

5 x 10 = 50 Marks

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | Blooms Level | CO | Max. Marks |
| **UNIT-I** | | | | | |
| 2 | (a) | Explain various methods in regular expressions with suitable examples and build a program to validate phone numbers using regular expressions. | L2 | CO2 | 5M |
| (b) | Develop a Python program using with lambda, map (), filter (), and reduce () functions. | L3 | CO2 | 5M |
| **OR** | | | | | |
| 3 | (a) | Explain file handling in JSON. | L2 | CO2 | 5M |
| (b) | Develop a Python program to convert a Dictionary to a JSON object. | L3 | CO2 | 5M |
| **UNIT-II** | | | | | |
| 4 | (a) | Develop a Program to perform matrix multiplication using NumPy. | L3 | CO3 | 5M |
| (b) | Develop a method to transpose a matrix using NumPy array | L3 | CO3 | 5M |
| **OR** | | | | | |
| 5 | (a) | Illustrate the concept of masked arrays in NumPy | L2 | CO3 | 5M |
| (b) | Build a Python program that replaces all negative values with zero. | L3 | CO3 | 5M |
| **UNIT-III** | | | | | |
| 6 | (a) | Compare merge(), concat(), and join() functions in Pandas with examples. | L2 | CO3 | 5M |
| (b) | Explain Pandas data wrangling techniques | L2 | CO3 | 5M |
| **OR** | | | | | |
| 7 | (a) | Demonstrate how to visualize data using Pandas and Matplotlib. | L2 | CO3 | 5M |
| (b) | Develop a program to generate a line plot of a dataset using Pandas. | L3 | CO3 | 5M |
| **UNIT-IV** | | | | | |
| 8 | (a) | Explain the impact of the Global Interpreter Lock (GIL) on Python multithreading. | L2 | CO4 | 5M |
| (b) | Develop a Python program to implement a producer-consumer model using threading and synchronization. | L3 | CO4 | 5M |
| **OR** | | | | | |
| 9 | (a) | Explain how sockets facilitate network communication. | L2 | CO4 | 3M |
| (b) | Build a basic Python program to establish a connection between a client and server using TCP and UDP. | L3 | CO4 | 7M |
| **UNIT-V** | | | | | |
| 10 | (a) | Explain the structure of a computational graph in TensorFlow | L2 | CO5 | 5M |
| (b) | (b) List the normalization and standardization techniques to apply feature scaling in TensorFlow | L4 | CO5 | 5M |
| **OR** | | | | | |
| 11 | (a) | (a) Compare and contrast the difference between the Sequential API and Functional API in Keras. | L4 | CO5 | 5M |
| (b) | (b) Analyze the steps involved in creating, training, evaluating, saving, and reloading a Keras model using TensorFlow. Identify the key components in each stage and how they interact to form a complete workflow. | L4 | CO5 | 5M |

**Course Coordinator Module Coordinator HOD**