|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RegisterNumber** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**SRM Institute of Science and TechnologyCollegeofEngineeringandTechnology**

Set-B

**SchoolofComputing**

SRMNagar,Kattankulathur–603203,ChengalpattuDistrict,TamilNadu

# AcademicYear:2024-25(Even)

Test: FT1 Date:25-02-2025

CourseCode&Title:21CSS303T-Data Science Duration:50 Minutes

Year& Sem: IIIYear /VISem Max.Marks:25

CourseArticulationMatrix:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course  Outcome | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | - | - | - | - | 1 | - | - | - | - | - | - | - |
| CO2 | - | - | - | - | 1 | - | - | - | - | - | - | - |

**Note:** CO1 - To understand the relationship between data

CO2 - Identify the different data structures to represent data

**Part– A**

(5x2= 10 Marks)

Answer ALL the questions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Q.No | Question | Marks | BL | CO | PO | PI.Code |
| 1 | Given the NumPy array arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]]), write the code to extract the second column as a 1D array.  import numpy as np  arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])  # Extract the second column (index 1) as a 1D array  second\_column = arr[:, 1]  print(second\_column)  Output:  [2 5 8] | 2 | 3 | 2 | 5 | 5.4.2 |
| 2 | How do you select a column from a Pandas DataFrame? Write the code.  We can select a column from a Pandas DataFrame using its column name.  import pandas as pd  # Create a DataFrame  data = {'A': [1, 4, 7], 'B': [2, 5, 8], 'C': [3, 6, 9]}  df = pd.DataFrame(data)  # Select column 'B' as a Series  column\_b = df['B']  print(column\_b)  Output  0 2  1 5  2 8  Name: B, dtype: int64 | 2 | 1 | 2 | 5 | 5.4.1 |
| 3 | Mention two sources from which data can be acquired for analysis.  Two common sources from which data can be acquired for analysis are:   1. **Web APIs**    * Many online services provide APIs to fetch structured data in formats like JSON or XML.    * Example: Twitter API for social media analysis, OpenWeather API for weather data, and financial APIs for stock market data. 2. **Public Datasets and Open Data Portals**    * Governments, research organizations, and companies provide free datasets for public use.    * Example: Kaggle (https://www.kaggle.com/datasets), Google Dataset Search, and UCI Machine Learning Repository | 2 | 1 | 1 | 5 | 5.5.1 |
| 4 | Write a Python program to add, subtract, multiply and divide two Pandas Series  Sample Series: [2, 4, 6, 8, 10], [1, 3, 5, 7, 9]  import pandas as pd  # Create two Pandas Series  series1 = pd.Series([2, 4, 6, 8, 10])  series2 = pd.Series([1, 3, 5, 7, 9])  # Perform arithmetic operations  addition = series1 + series2  subtraction = series1 - series2  multiplication = series1 \* series2  division = series1 / series2 # This will perform element-wise division  # Display results  print("Addition:\n", addition)  print("\nSubtraction:\n", subtraction)  print("\nMultiplication:\n", multiplication)  print("\nDivision:\n", division)  Output  Addition:  0 3  1 7  2 11  3 15  4 19  dtype: int64  Subtraction:  0 1  1 1  2 1  3 1  4 1  dtype: int64  Multiplication:  0 2  1 12  2 30  3 56  4 90  dtype: int64  Division:  0 2.000000  1 1.333333  2 1.200000  3 1.142857  4 1.111111  dtype: float64 | 2 | 2 | 1 | 5 | 5.4.2 |
| 5 | What are Web APIs and how are they used in Data Acquisition?  **Web APIs** (Application Programming Interfaces) are a set of rules and protocols that allow different software applications to communicate with each other over the internet. They enable applications to request and exchange data, typically in a structured format like JSON or XML.  In the context of **Data Acquisition**, Web APIs are used to retrieve or send data from one system to another, allowing for the automation of data collection from remote sources, such as databases, external systems, or online services | 2 | 2 | 1 | 5 | 5.4.1 |

**Part– B**

(3x5= 15 Marks)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Q.No | Question | Marks | BL | CO | PO | PI.Code |
| 1 | Explain the complete Data Science Process in detail with suitable real-world examples.  (Diagram - 1 mark)  Explanation of each stage (4 marks) | 5 | 2 | 1 | 5 | 5.4.1 |
| 2 | You're tasked with exploring a large dataset using Pandas. You suspect there might be a relationship between two columns: 'age' (numerical) and 'purchase\_category' (categorical). Describe how you would use Pandas to investigate this potential relationship. Mention TWO specific Pandas functions you would use and explain their purpose in this context."  To explore the relationship between **'age'** (numerical) and **'purchase\_category'** (categorical), I would use the following two Pandas functions:  **1. groupby()** (2.5 marks)   * This function allows us to group data based on the categorical column ('purchase\_category') and then compute summary statistics for the numerical column ('age'). * **Purpose:** It helps in understanding the distribution of ages across different purchase categories. * **Example Usage:**   import pandas as pd  # Sample DataFrame  data = {'age': [25, 34, 45, 23, 41, 36, 29, 50],  'purchase\_category': ['Electronics', 'Clothing', 'Electronics', 'Books', 'Books', 'Clothing', 'Electronics', 'Books']}  df = pd.DataFrame(data)  **2. value\_counts() (on grouped data)** (2.5 marks)   * Purpose: Helps count the occurrences of different purchase categories within specific age groups to identify buying patterns. * Example Usage:   # Create age bins  df['age\_group'] = pd.cut(df['age'], bins=[20, 30, 40, 50, 60], labels=['20-30', '30-40', '40-50', '50-60'])  # Count how many purchases are made in each category within age groups  purchase\_counts = df.groupby('age\_group')['purchase\_category'].value\_counts()  print(purchase\_counts) | 5 | 2 | 2 | 5 | 5.5.1 |
| 3 | You are developing a price comparison tool to track the price of a specific product (e.g., "iPhone 15" or "Samsung Galaxy S23") from multiple e-commerce websites such as Amazon, eBay, and Walmart. Explain the key steps involved in performing web scraping for this task, covering aspects such as identifying the target websites, extracting the relevant data, handling dynamic content, and storing the collected information for further analysis.  Step 1: Identifying Target Websites (1 mark)   * Choose e-commerce platforms to track prices from, such as Amazon, eBay, Walmart, etc. * Analyze the website structure by inspecting product pages to find relevant elements (e.g., price, product name, availability). * Ensure that scraping these sites complies with their Terms of Service to avoid legal issues.   Step 2: Extracting Relevant Data (1 mark)  To extract product information, we need:   * Product name * Price * Availability * Seller information * Product URL   Step 3: Handling Dynamic Content (JavaScript-Rendered Websites) (1 mark)   * Some websites dynamically load prices using JavaScript, making BeautifulSoup insufficient. * Solution: Use Selenium or Scrapy to simulate user interaction and fetch content.   Step 4: Storing Collected Data (1 mark)  The extracted data should be stored for further analysis.  Step 5: Automating Price Tracking (1 mark)   * Use scheduled tasks (cron jobs on Linux, Task Scheduler on Windows) to run the scraper at intervals (e.g., daily). * Send email alerts when price drops below a threshold. | 5 | 3 | 2 | 5 | 5.5.1 |

**Course Outcome (CO)and Bloom’s level (BL)Coverage in Questions**