**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **21/may/2020** | **Name:** | **Prashantha naik** |
| **Course:** | **Tcs ion** | **USN:** | **4al17ec074** |
| **Topic:** | 1. **Learn Corporate Telephone Etiquette** 2. **Understand Accounting Fundamentals** 3. **Gain Foundational Skills in IT** | **Semester & Section:** | **6th B** |
| **Github Repository:** | **prashanth\_course** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report – Report can be typed or hand written for up to two pages.**  **About Learn Corporate Telephone Etiquette**  **First impression**   * Be alert * Be pleasant * Be natural * Be Distinctive   **Do’s during the call**   * Ask the purpose of call * Give due importance to the call * Transfer the call if required * Acknowledge the caller’s queries   **Don’ts during the telephone etiquette**   * Don’t bluff * Don’t speak negatively * Don’t sound weary * Don’t argue with the caller   **Taking messages**   * Takes the notes on paper * Keep your stationary ready * Jot down important information * Write neatly * Verify the details * Note the urgency of the message   **Essential guidelines for telephone etiquette**   * Answer the call promptly * Identify your organization and then yourself * Show your genuine interest in the caller * Address their problem positivily * Listen patiently   **Understand Accounting Fundamentals**   * Accounting is a system which collects and processes financial information of business * Process of accounting given below     **Accounting cycle:**    **Accounting principles and concepts**   * Dual aspect principle * Historical cost concept * Matching concept * Full disclosure concept * Variable and objective Evidence concept   **Debit and credit rules**    **Gain Foundational Skills in IT**   * **Learnt about what do the Recruiters except** * **We should no any of the programming knowledge** * **About schema/tables/columns** * **Should know the basic algorithm** * **Searching (Linear/binary),sorting(buble, selection, insertion, Merge)** * **Design pattern** * **About one digital skill** * **Should know the basics of AI, warehouse** * **Learnt about the http protocal** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **21/may/2020** | **Name:** | **Prashantha naik** | |
| **Course:** | **Python** | **USN:** | **4al17ec074** | |
| **Topic:** | 1. **More on Functions** 2. **File Processing** 3. **Imported Modules** 4. **Application 1: Build an Interactive English Dictionary** | **Semester&Section:** | **6th b** | |
| **Git hub repository** | **prashanth\_course** |  |  | |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session** | | | |
| **Report – Report can be typed or hand written for up to two pages.**  **More on function**   * **Learn about how to use function** * Functions can have more than one **parameter**: * def volume(a, b, c): * return a \* b \* c * Functions can have **default** parameters (e.g. coefficient): * def converter(feet, coefficient = 3.2808): * meters = feet / coefficient * return meters * print(converter(10))   Output: 3.0480370641306997  Arguments can be passed as **non-keyword** (positional) arguments (e.g. a) or **keyword** arguments (e.g. b=2 and c=10):   * def volume(a, b, c): * return a \* b \* c * print(volume(1, b=2, c=10)) * An **\*args**parameter allows the  function to be called with an arbitrary number of non-keyword arguments: * def find\_max(\*args): * return max(args) * print(find\_max(3, 99, 1001, 2, 8))   Output: 1001  **File Processing**   * **read** an existing file with Python: * with open("file.txt") as file: * content = file.read() * **create** a new file with Python and **write** some text on it: * with open("file.txt", "w") as file: * content = file.write("Sample text") * **append and read** a file with: * with open("file.txt", "a+") as file: * content = file.write("Even more sample text") * file.seek(0) * content = file.read()   **Imported Modules**   * **Builtin objects** are all objects that are written inside the Python interpreter in C language. * **Builtin modules** contain builtins objects. * Some builtin objects are not immediately available in the global namespace. They are parts of a builtin module. To use those objects the module needs to be **imported** first. * **Standard libraries** is a jargon that includes both builtin modules written in C and also modules written in Python. * **Standard libraries** written in Python reside in the Python installation directory as .py files. You can find their directory path with sys.prefix.   **Application 1: Build an Interactive English Dictionary**   1. import json 2. from difflib import get\_close\_matches 3. data = json.load(open("data.json")) 4. def translate(w): 5. w = w.lower() 6. if w in data: 7. return data[w] 8. elif w.title() in data: 9. return data[w.title()] 10. elif w.upper() in data: #in case user enters words like USA or NATO 11. return data[w.upper()] 12. elif len(get\_close\_matches(w, data.keys())) > 0: 13. yn = input("Did you mean %s instead? Enter Y if yes, or N if no: " % get\_close\_matches(w, data.keys())[0]) 14. if yn == "Y": 15. return data[get\_close\_matches(w, data.keys())[0]] 16. elif yn == "N": 17. return "The word doesn't exist. Please double check it." 18. else: 19. return "We didn't understand your entry." 20. else: 21. return "The word doesn't exist. Please double check it." 22. word = input("Enter word: ") 23. output = translate(word) 24. if type(output) == list: 25. for item in output: 26. print(item) 27. else: 28. print(output) | | | |