**DAILY ASSESSMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **25/05/2020** | **Name:** | **Dhavala** |
| **Course:** | **Digital Signal Processing** | **USN:** | **4AL17EC027** |
| **Topic:** | * **Introduction to Fourier Series & Fourier Transform** * **Fourier Series – Part 1** * **Fourier Series – Part 2** * **Inner Product in Hilbert Transform** * **Complex Fourier Series** * **Fourier Series using Matlab** * **(Use Octave to execute the code)** * **Fourier Series using Python**   **(Experience**  **implementation using**  **Python)**   * **Fourier Series and Gibbs Phenomena Using Matlab** | **Semester & Section:** | **6TH SEM & A Section** |
| **Github Repository:** | **Dhavala27** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
|  |
| **Report**     |  |  |  |  | | --- | --- | --- | --- | | **Date:** | **25/05/2020** | **Name:** | **Dhavala** | | **Course:** | **Python** | **USN:** | **4AL17EC027** | | **Topic:** | * **Fixing Programming Errors** * **Application 3: Build a Website Blocker** | **Semester & Section:** | **6TH SEM & A Section** | | **Github Repository:** | **Dhavala27** |  |  |  |  | | --- | | **FORENOON SESSION DETAILS** | | **Image of session** |   **Report**  The any() function  Hi! In case you didn't understand the if not any (website in line for website in website list) part in the previous video, here is another example:  >>> lines  = ["trees are good", "pool is fresh", "face is round"]  >>> website\_list = ["face", "clock", "trend"]  >>> for line in lines:  ...     any(website in line for website in website\_list)  ...  False  False  True  We start iterating over the items of website list using a for loop. In the first iteration we would have:  any (website in “trees are good” for website in website list)  Inside the parenthesis of any() there's another loop that iterates over website list:  ("face" in "trees are good")  ("clock" in "trees are good")  ("trend" in "trees are good")  If any of the above is True you get the expression evaluated to True. In this case none of them is True, so you get False.  If you want to return True (if all of them are True), use all() instead of any().  So, the part any (website in line for website in website list) part will either be equal to True  or False.  Scheduling a Python Program on a Server  Scheduling a Python program on a 24/7 server  Keeping your computer on 24-7 is not practical, so if you want to execute a Python script at a particular time every day, you probably need a computer that is on all the time.  PythonAnywhere gives you access to such a 24-7 computer. You can upload a Python script and schedule it to run at a certain time every day. This availability can be useful, for example, when you want to extract some values (e.g., weather data) from a website and generate a text file with the value or other reports every day.  To schedule a Python script for execution on PythonAnywhere, follow these simple steps:   1. Sign up for a free account at https://www.pythonanywhere.com. 2. Go to your *Dashboard*, *Files*, *Upload a File,* and upload the Python file you want to schedule for execution. 3. Go to *Tasks* and set the time of the day you want your script to be executed and type in the name of the Python file you  uploaded (e.g., *myscript.py*). Note that the time you enter should be in UTC. 4. Click the *Create* button and you’re done.   Your Python file will now be executed every day at your specified time. If you don't have a Python script and you’re still confused about the benefit of this, here is a very simple Python script that you can  use to try the above steps:  If you don’t have a Python script and you’re still confused about the benefits of this PythonAnywhere feature, here is a very simple Python script you can use to schedule for execution:   1. from datetime import datetime 2. with open(datetime.now().strftime("%Y-%m-%d-%H-%M-%S"), "w") as myfile: 3. myfile.write("Hi there!")   The above code creates a text file and writes the string “Hi there!”  in that text file. The name of the text file will be the current date and time. For example one file name example would be 2018-02-16-18-20-33.txt.  That name is generated by datetime.now() indicating the date and time the script was executed.  Every time the script is executed, the script generates a new text file with a different name. You will have a new text file created every day. |