**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **26-05-2020** | **Name:** | **BHOOMIKA HEBBAR** |
| **Course:** | **Digital signal processing** | **USN:** | **4AL17EC010** |
| **Topic:** | **Fourier Series & Gibbs Phenomena using Python, Fourier Transform and derivatives,convolution,laplace transform and z- transform** | **Semester & Section:** | **6th & A** |
| **Github Repository:** | **bhoomika\_python** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session**  **1.**    **2.**    **3.**    **4.** |
| **Report – Report can be typed or hand written for up to two pages.**  **Today I have learnt:**   * Fourier Series & Gibbs Phenomena using Python * Fourier Transform and its derivative * Fourier Transform and Convolution * Intuition of Fourier Transform and Laplace Transform * Laplace Transform of First order * Implementation of Laplace Transform using Matlab * Applications of Z-Transform * Find the Z-Transform of sequence using Matlab   Given the Fourier transform of a general function, find the Fourier transform of its derivative. Use this result to find the Fourier transform of a window function out of the Fourier transform of an antisymmetric pair of delta functions. (t)eiωtdt. (t)] = −iωF(ω).  Each of these sinusoidal terms has a magnitude (scale factor) and a phase (shift). – Note that in a computer, we can represent a function as an array of numbers giving the values of that function at equally spaced points fourier and Laplace transforms are very much related to each other, and both of them can help you solve differential equations, but the intuitive difference is that: The Fourier Transform is more useful for understanding the steady state response of a system.  Z transform is used to convert discrete time domain signal into discrete frequency domain signal. It has wide range of applications in mathematics and digital signal processing. It is mainly used to analyze and process digital data. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **26-05-2020** | **Name:** | **BHOOMIKA HEBBAR** | |
| **Course:** | **UDEMY PYTHON MEGA\_COURSE** | **USN:** | **4AL17EC010** | |
| **Topic:** | **Graphical User Interfaces with Tkinter, Interacting with Databases** | **Semester & Section:** | **6th &A** | |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session**  **1.**  **2.** | | | |
| **Report – Report can be typed or hand written for up to two pages.**  **Today I have learnt :**   * **Graphical user interface** * Introduction to Tkinter * Setting up a GUI with Widgets * Connecting GUI Widgets with Callback Functions * Create a Multi-widget GUI (Practice) * Solution * **Interacting with databases:** * Introduction to "Python with Databases" * Connecting and Inserting Data to SQLite via Python * Selecting, Inserting, Deleting, and Updating SQLite Records * Introduction to PostgreSQL Psycopg2 * Selecting, Inserting, Deleting, and Updating PostgreSQL Records * Querying data from a MySQL database * **Program:**     from tkinter import \*  # Create an empty Tkinter window  window=Tk()  def from\_kg():  # Get user value from input box and multiply by 1000 to get kilograms  gram=float(e2\_value.get())\*1000  # Get user value from input box and multiply by 2.20462 to get pounds  pound=float(e2\_value.get())\*2.20462    # Get user value from input box and multiply by 35.274 to get ounces  ounce=float(e2\_value.get())\*35.274   # Empty the Text boxes if they had text from the previous use and fill them again  t1.delete("1.0", END) # Deletes the content of the Text box from start to END  t1.insert(END,gram) # Fill in the text box with the value of gram variable  t2.delete("1.0", END)  t2.insert(END,pound)  t3.delete("1.0", END)  t3.insert(END,ounce)   # Create a Label widget with "Kg" as label  e1=Label(window,text="Kg")  e1.grid(row=0,column=0) # The Label is placed in position 0, 0 in the window   e2\_value=StringVar() # Create a special StringVar object  e2=Entry(window,textvariable=e2\_value) # Create an Entry box for users to enter the value  e2.grid(row=0,column=1)  # Create a button widget  # The from\_kg() function is called when the button is pushed  b1=Button(window,text="Convert",command=from\_kg)  b1.grid(row=0,column=2)  # Create three empty text boxes, t1, t2, and t3  t1=Text(window,height=1,width=20)  t1.grid(row=1,column=0)    t2=Text(window,height=1,width=20)  t2.grid(row=1,column=1)   t3=Text(window,height=1,width=20)  t3.grid(row=1,column=2)   # This makes sure to keep the main window open  window.mainloop()  **databases:**  You also need a Python library that interacts with MySQL databases. Many libraries are compatible, but I prefer *mysql.connector*. To install *mysql.connector*: simply execute **pip install mysql-connector**  or **pip3 install mysql-connector** depending on whether you use *pip* or *pip3*. Once you install the library, try this working example:   1. import mysql.connector 2. word = input("Enter a word in English and press Enter: ") 3. con = mysql.connector.connect( 4. user="ardit700\_student", 5. password = "ardit700\_student", 6. host="108.167.140.122", 7. database = "ardit700\_pm1database" 8. ) 9. cursor = con.cursor() 10. query = cursor.execute("SELECT \* FROM Dictionary WHERE Expression = '%s'" % word) 11. results = cursor.fetchall() 12. if results: 13. for result in results: 14. print(result[1]) 15. else: 16. print("We couldn't find any results about that.") | | | |