**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **04/06/2020** | **Name:** | **Prajwal Kamagethi Chakravarti P L** |
| **Course:** | **Python** | **USN:** | **4AL17EC073** |
| **Topic:** | **Application to Build a Web-based Financial Graph** | **Semester & Section:** | **6 & B** |
| **Github Repository:** | **https://github.com/alvas-education-foundation/Prajwal-Kamagethi.git** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of the session:**        **Report – Report can be typed or hand written for up to two pages.**  **In this section we learnt analyzing Stock Market Data,Plotting Stock Market Data Candlestick Charts,Updating Candlestick Charts with Bokeh Quadrants,Learnt to plot Candlestick Charts with Bokeh Rectangles,Creating Candlestick Segments,Stylizing the obtained Chart,Learnt the Concept Behind Embedding Bokeh,Sharing the Charts in a Flask Webpage Learnt how to Embed the Bokeh Chart in a Webpage and also learnt to Deploy the Chart Website to a Live Server.** |
| |  |  |  |  | | --- | --- | --- | --- | | **Date:** | **04-06-2020** | **Name:** | **Prajwal Kamagethi Chakravarti P L** | | **Course:** | **DIGITAL DESIGN USING HDL** | **USN:** | **4AL17EC073** | | **Topic:** | * **Hardware modelling using**   **Verilog**   * **FPGA and ASIC Interview questions** | **Semester & Section:** | **6TH & B** | | **Github Repository:** | **https://github.com/alvas-education-foundation/Prajwal-Kamagethi.git** |  |  | |
| **Image of the session:**        **Report: In this section, we learnt about the Verilog hardware description language.**  **•  Understood the difference between behavioral and structural design styles.**  **•  Learnt to write test benches and analyze simulation results.**  **•  Learnt to model combinational and sequential circuits.**  **•  Distinguish between good and bad coding practices.**  **•  Case studies with some complex designs.**  **TASK: Implement a simple T Flipflop and test the module using a compiler.**  **Design:**  **module tff ( input clk,**  **input rstn,**  **input t,**  **output reg q);**    **always @ (posedge clk) begin**  **if (!rstn)**  **q <= 0;**  **else**  **if (t)**  **q <= ~q;**  **else**  **q <= q;**  **end**  **endmodule**  **Testbench**  **module tb;**  **reg clk;**  **reg rstn;**  **reg t;**    **tff u0 ( .clk(clk),**  **.rstn(rstn),**  **.t(t),**  **.q(q));**    **always #5 clk = ~clk;**    **initial begin**  **{rstn, clk, t} <= 0;**    **$monitor ("T=%0t rstn=%0b t=%0d q=%0d", $time, rstn, t, q);**  **repeat(2) @(posedge clk);**  **rstn <= 1;**    **for (integer i = 0; i < 20; i = i+1) begin**  **reg [4:0] dly = $random;**  **#(dly) t <= $random;**  **end**  **#20 $finish;**  **end**  **endmodule** |