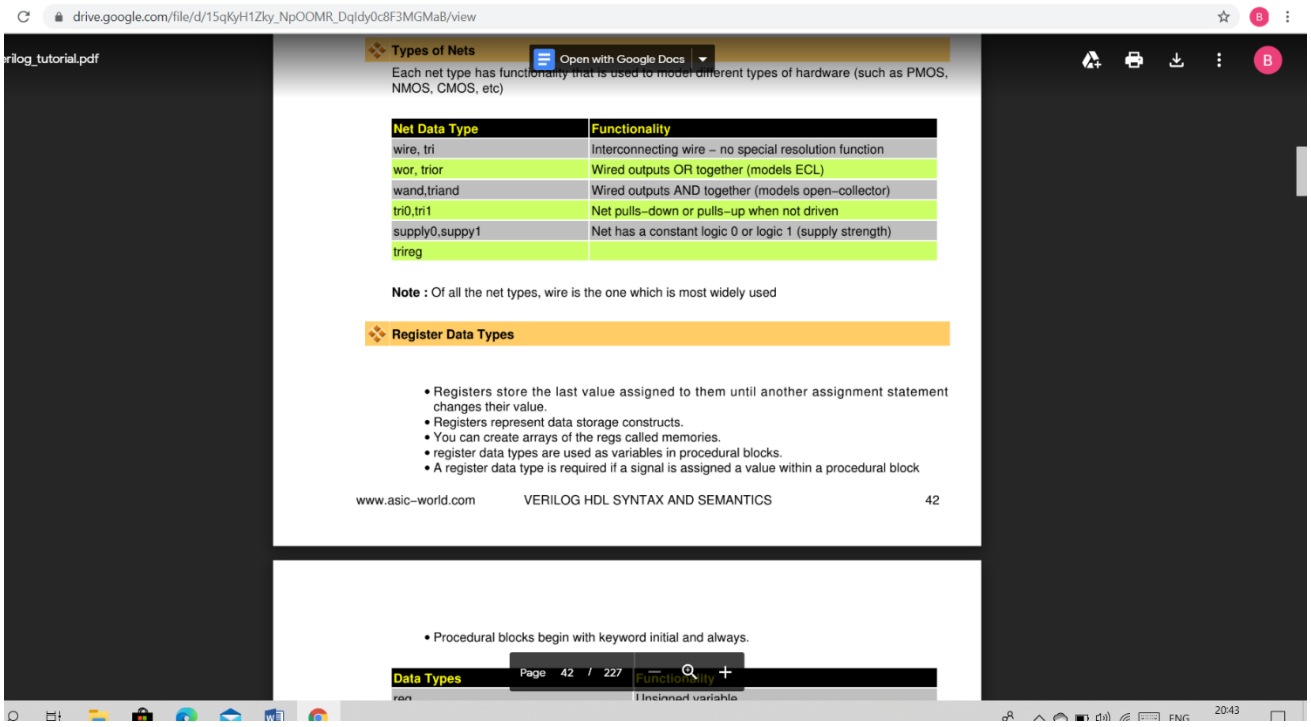
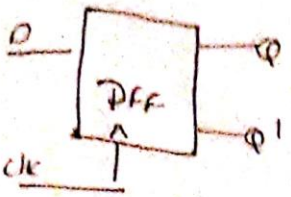
DAILY ASSESSMENT FORMAT

Date:	**5-06-2020**	**Name:**	**BINDUSHRI**
Course:	**Digital design using HDL**	**USN:**	**4AL17EC011**
Topic:	**1.verilog tutorial class** **2.FPGA projects** **3.task**		**6th A**
Github Repository:	**Bindushri**		

FORENOON SESSION DETAILS



Day5 Verilog Tutorials & Practice programs



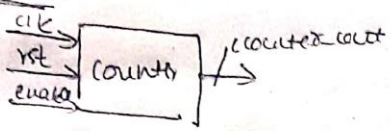
1. D flip-flop code
2. module d_ff(d, clk, q, q_bar);
3. input d, clk;
4. output q, q_bar;
5. wire d, clk;
6. reg q, q_bar;
7.
8. always @(posedge clk)
9. begin
10. q <= d;
11. q_bar <= !d;
12. end
13. endmodule.

- Bottom-up Design
- Top down Design.
- History of Verilog
- various stages of ASIC/FPGA.
  - Specification
  - High level design
  - Micro Design / Low level Design
  - RTL coding
  - Simulation
  - Synthesis
  - Place & Route
  - Post si validation

1. Basic program

```
module hello_world;
  initial begin
    $display("Hello world by Deepak");
    #10 $finish;
  end
endmodule
```

Counter



- 4-bit synchronous up counter
- active high, synchronous reset
- Active high enable

MUX

```
module mux_2to1 (a,b,sel,y);
input a,b;
output y;
input sel;
wire y;
assign y=(sel)?b:a;
endmodule
```

* if-else, case, would same as C

* if else & case statements require all the cases to covered for combinational logic

* for-loop same as C but no ++ & -- characters.

→ FPGAProjects

→ Task - day 5

Implement a verilog module the count number of 0's in a 16'bit number.

→

```
module num_zero_for(
    input [15:0] A,
    output reg [4:0] ones
);

integer i;

always @(A)
begin
count = 0;
for (i=0; i<16; i=i+1)
  if (A[i] == 1'b0)
     count = count+1;

end
endmodule.
```

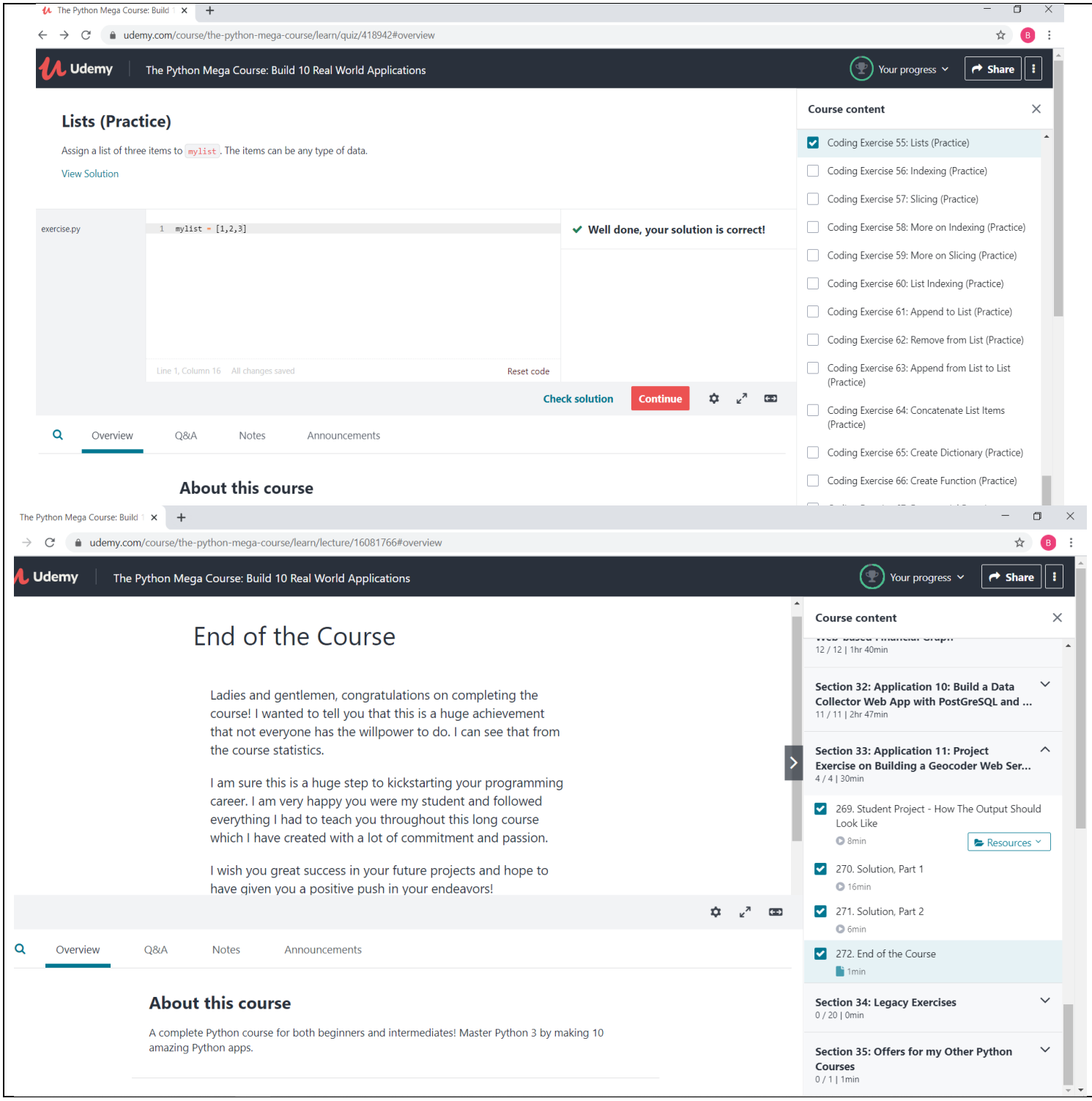
Date:5june2020

Name:Bindushri

Course: python **USN:4AL17EC011**

Topic:sec 33-34 **Sem&Sec:6th A**

AFTERNOON SESSION DETAILS


The Python Mega Course: Build
udemy.com/course/the-python-mega-course/learn/quiz/418942#overview
Udemy
The Python Mega Course: Build 10 Real World Applications
Your progress
Share
Lists (Practice)
Assign a list of three items to mylist. The items can be any type of data.
View Solution
exercise.py
1 mylist = [1,2,3]
Well done, your solution is correct!
Line 1, Column 16 All changes saved
Reset code
Check solution
Continue
Overview
Q&A
Notes
Announcements
About this course
Course content
Coding Exercise 55: Lists (Practice)
Coding Exercise 56: Indexing (Practice)
Coding Exercise 57: Slicing (Practice)
Coding Exercise 58: More on Indexing (Practice)
Coding Exercise 59: More on Slicing (Practice)
Coding Exercise 60: List Indexing (Practice)
Coding Exercise 61: Append to List (Practice)
Coding Exercise 62: Remove from List (Practice)
Coding Exercise 63: Append from List to List (Practice)
Coding Exercise 64: Concatenate List Items (Practice)
Coding Exercise 65: Create Dictionary (Practice)
Coding Exercise 66: Create Function (Practice)
The Python Mega Course: Build
udemy.com/course/the-python-mega-course/learn/lecture/16081766#overview
Udemy
The Python Mega Course: Build 10 Real World Applications
Your progress
Share
End of the Course
Ladies and gentlemen, congratulations on completing the course! I wanted to tell you that this is a huge achievement that not everyone has the willpower to do. I can see that from the course statistics.
I am sure this is a huge step to kickstarting your programming career. I am very happy you were my student and followed everything I had to teach you throughout this long course which I have created with a lot of commitment and passion.
I wish you great success in your future projects and hope to have given you a positive push in your endeavors!
Overview
Q&A
Notes
Announcements
About this course
A complete Python course for both beginners and intermediates! Master Python 3 by making 10 amazing Python apps.
Course content
12 / 12 | 1hr 40min
Section 32: Application 10: Build a Data Collector Web App with PostGreSQL and ...
11 / 11 | 2hr 47min
Section 33: Application 11: Project Exercise on Building a Geocoder Web Ser...
4 / 4 | 30min
269. Student Project - How The Output Should Look Like
8min
Resources
270. Solution, Part 1
16min
271. Solution, Part 2
6min
272. End of the Course
1min
Section 34: Legacy Exercises
0 / 20 | 0min
Section 35: Offers for my Other Python Courses
0 / 1 | 1min

5-06-2020 Friday

Section 33 Building a Geocoder web server

creating 4 folders
app1.py , app2.py , app3.py , app4.py

* Modification can be done on Precision Sec code
* import datetime in order to generate the geocoder.
* this will make Python to generate the CSV file at the o/p
*

exercise

1) create a variable named character and assign johnsnow

-> character = "Johnsnow"

2) create a variable named price and assign it value of 10.
in the next line print the value of variable price.

-> Price = 10;
print (price)

Certificate :