Date:- 3/6/20
Course:- Digital design Using HDL
Topic:- EDA Tool

Name:- Harshitha. T
USN:- 4AL17EC106
Sem & Sec:- 6th B

* Implement Inverter using the EDA tool

```
module inverter (y, a);
output y;
input a;
assign y = ~a;
endmodule
```

⇒ Test bench code

```
timescale 1ns / 1ps
module testbench();
reg a1;
wire y1;
inverter inv1 (a1, y1);
initial begin
a1 = 1'b1;
$ display ("a = %b", a1);
end
endmodule
```

* Ripple carry counter

```
module ripple_counter_4_bit (q, clk, reset);
input clk, reset;
output [3:0] q;
T_FF tff0 (q[0], clk, reset);
T_FF tff1 (q[1], q[0], reset);
T_FF tff2 (q[2], q[1], reset);
```

```
T_FF tff3 (q[3], q[2], reset);
endmodule
module T_FF (q, clk, reset);
input clk, reset;
output q;
wire d;
D_FF dff0 (q, d, clk, reset);
not n1 (d, q);
endmodule
module D_FF (q, d, clk, reset);
input d, clk, reset;
output reg q;
always @ (negedge clk or posedge reset)
begin
else
q <= d;
end
end module.
```

⇒ test bench code

```
module test
reg clk, reset;
wire[3:0]q;
ripple_carry_counter dec(q, clk, reset);
initial begin
$ dumpfile("dump.vcd");
clk = 1'b0;
reset = 1'b1;
#10 reset = 1'b0;
end
always #5 clk = ~clk;
```

⇒ endmodule.

Date :- 3/6/20

Course :- Python

Topic :- Application 8.

Name :- Harshitha T

USN :- 4AL17EC406

Sem & Sec :- 6th B

* Scraped website Data → How the output will look.
* Extracting the elements without unique identifiers
* Saving the Extracted data in csv files
* firstly, from flask import flask, render_template

```
app = Flask(__name__)
@app.routes('/plot')
def plot();
 from pandas_datareader import data
 import datatime
 import fix_yahoo_finance as yf
 yf.pdr_override()
 from bokeh.plotting import figure, show, output_file
 from bokeh.embed import components
 from bokeh.resources import CDN
start = datetime.datetime(2015,11,1)
end = datetime.datetime(2016,3,10)
df = data.get_data_yahoo(tickers="GOOG", start=start
                                        end=end)
def inc_dec(c,o);
if c>o;
value = 'Increase'
elif c<o;
value = "Decrease"
else:
```

```
    value = "Equal"
    return value
df["Status"] = [inc_dec(c,o) for c,o in zip(df.close, df.open)]
df["Middle"] = (df.open+df.close)/2
df["Height"] = abs(df.close - df.open)
p = figure(x_axis_type='datatime', width=1000, height=300)
p.title.text = "Candle stick chart"
p.grid.grid_line_alpha = 0.3
p.segment(df.index, df.High, df.Index, df.low, colour='Black')
script1, div1 = components(p)
cdn_js = CDN.js_files[0]
cdn_css = CDN.css_files[0]
return render_templates("plot.html",
script = script1,
divl = divl,
cdn_css = cdn_css,
cdn_js = cdn_js)
```