



The screenshot shows a Jupyter Notebook interface with a sidebar on the left containing a 'Table of contents' and a main editor area. The editor area is titled 'Praktikum' and contains a Python script. The script imports 'ipywidgets' and 'numpy', defines two buttons ('btn_zero' and 'btn_one'), and sets up a game simulation with progress bars for user and bot scores. The script includes functions for clicking the buttons and updating the game state.

```
from ipywidgets import *
import numpy as np

btn_zero = Button( description='0' )
btn_one = Button( description='1' )
btns = HBox( [btn_zero, btn_one] )
btns

usr_score = IntProgress( value=0, min=0, max=30, description='You:', bar_style='success')
bot_score = IntProgress( value=0, min=0, max=30, description='Bot:', bar_style='danger')
scoreboard = VBox( [usr_score, bot_score] )
scoreboard

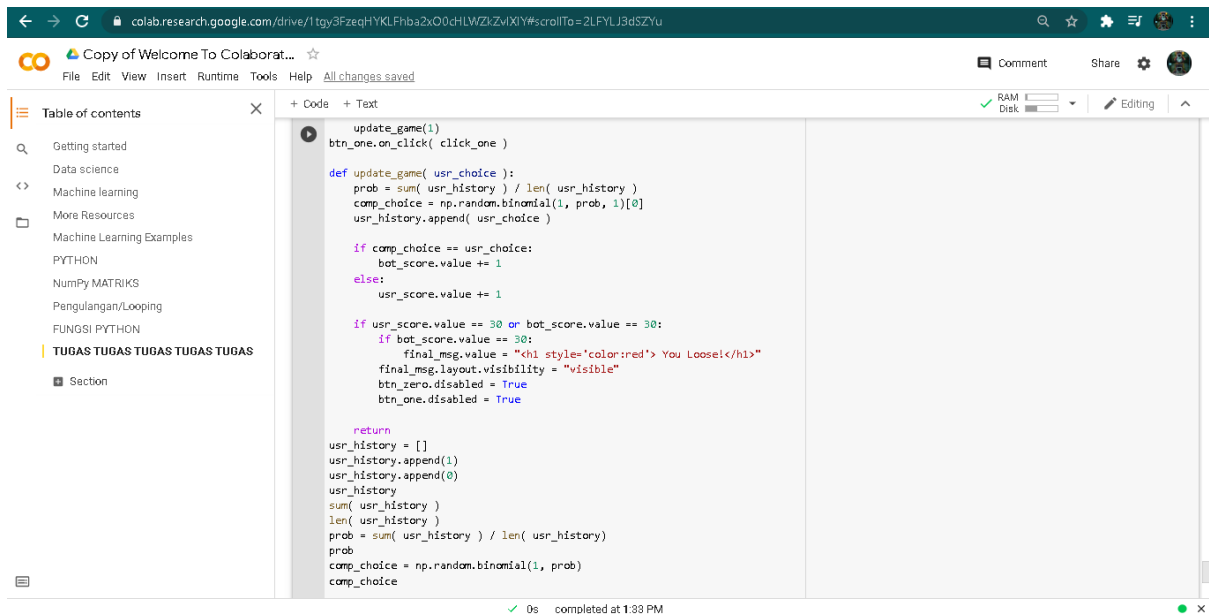
final_msg = HTML("<h1 style='color:green'> You Win!</h1>")
final_msg

game_box = VBox([ HBox( [scoreboard, final_msg] ),
                  btns
                ])
game_box

def click_zero(b):
    update_game(0)
    btn_zero.on_click( click_zero )

def click_one(b):
    update_game(1)
    btn_one.on_click( click_one )
```

0s completed at 1:38 PM



The screenshot shows a Jupyter Notebook interface with a sidebar on the left containing a 'Table of contents' and a main editor area. The editor area contains a Python script for a game simulation. The script defines a function 'update_game' that takes a user choice as input and updates the game state based on a random choice. It also includes a function for clicking the buttons and updating the game state.

```
update_game(1)
btn_one.on_click( click_one )

def update_game( usr_choice ):
    prob = sum( usr_history ) / len( usr_history )
    comp_choice = np.random.binomial(1, prob, 1)[0]
    usr_history.append( usr_choice )

    if comp_choice == usr_choice:
        bot_score.value += 1
    else:
        usr_score.value += 1

    if usr_score.value == 30 or bot_score.value == 30:
        if bot_score.value == 30:
            final_msg.value = "<h1 style='color:red'> You Loose!</h1>"
            final_msg.layout.visibility = "visible"
            btn_zero.disabled = True
            btn_one.disabled = True

        return

    usr_history = []
    usr_history.append(1)
    usr_history.append(0)
    usr_history
    sum( usr_history )
    len( usr_history )
    prob = sum( usr_history ) / len( usr_history )
    prob
    comp_choice = np.random.binomial(1, prob)
    comp_choice
```

0s completed at 1:38 PM

IS TUGAS

```
usr_score.value == 30 or bot_score.value == 30:
    if bot_score.value == 30:
        final_msg.value = "<h1 style='color:red'> You Loose!</h1>"
        final_msg.layout.visibility = "visible"
        btn_zero.disabled = True
        btn_one.disabled = True

    return
usr_history = []
usr_history.append(1)
usr_history.append(0)
usr_history
sum( usr_history )
len( usr_history )
prob = sum( usr_history ) / len( usr_history )
prob
comp_choice = np.random.binomial(1, prob)
comp_choice

display( game_box )
```

➡ You:  **You Loose!**
Bot: 

0 1