PRO-C204

LAN GAME

What is our GOAL for this CLASS?

In this class, we have learned to create a LAN server and use sockets to connect multiple clients to the server.

What did we ACHIEVE in the class TODAY?

- Created a LAN server.
- Created a GUI using tkinter to get the name of the player.
- Using sockets connected clients to the server.

Which CONCEPTS/ CODING BLOCKS did we cover today?

Add a bulleted list of new coding concepts that were covered in the class.

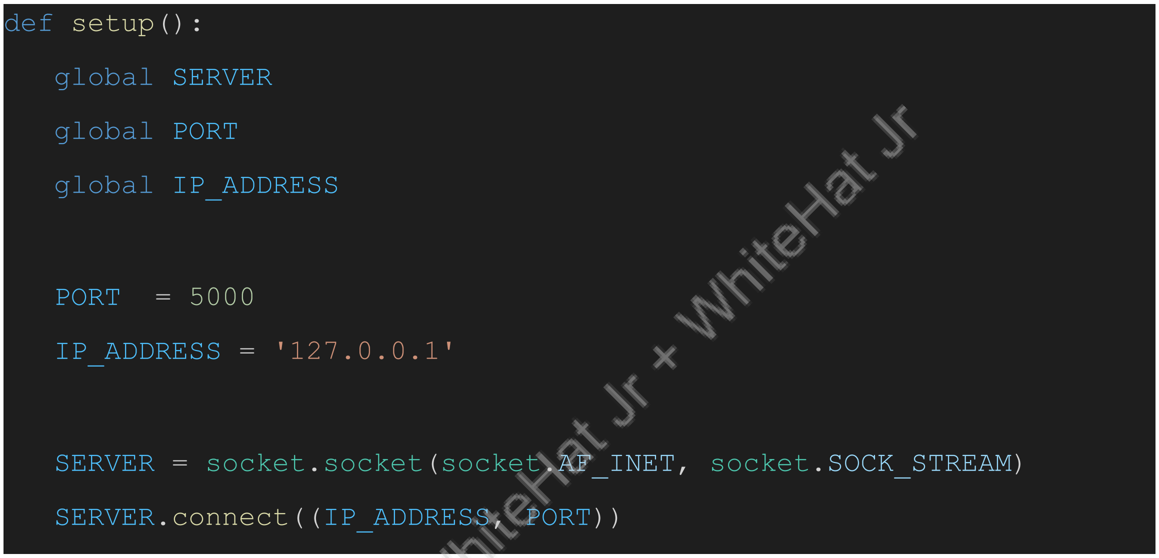
- Local host server.
- GUI using tkinter.
- Socket programming.

How did we DO the activities?

In earlier classes we have learned to create multiplayer games using a database where all the records of players were kept in the database.

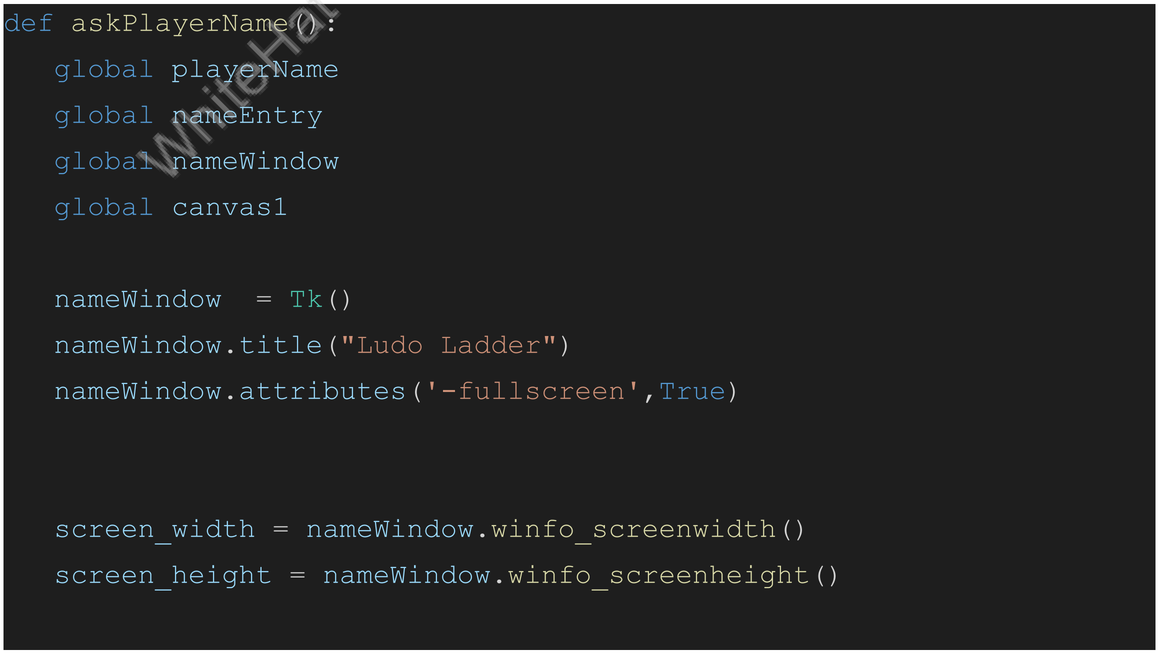
Today, you began by creating a LAN game where multiple players can join to play the game.

1. Create a Local host server.

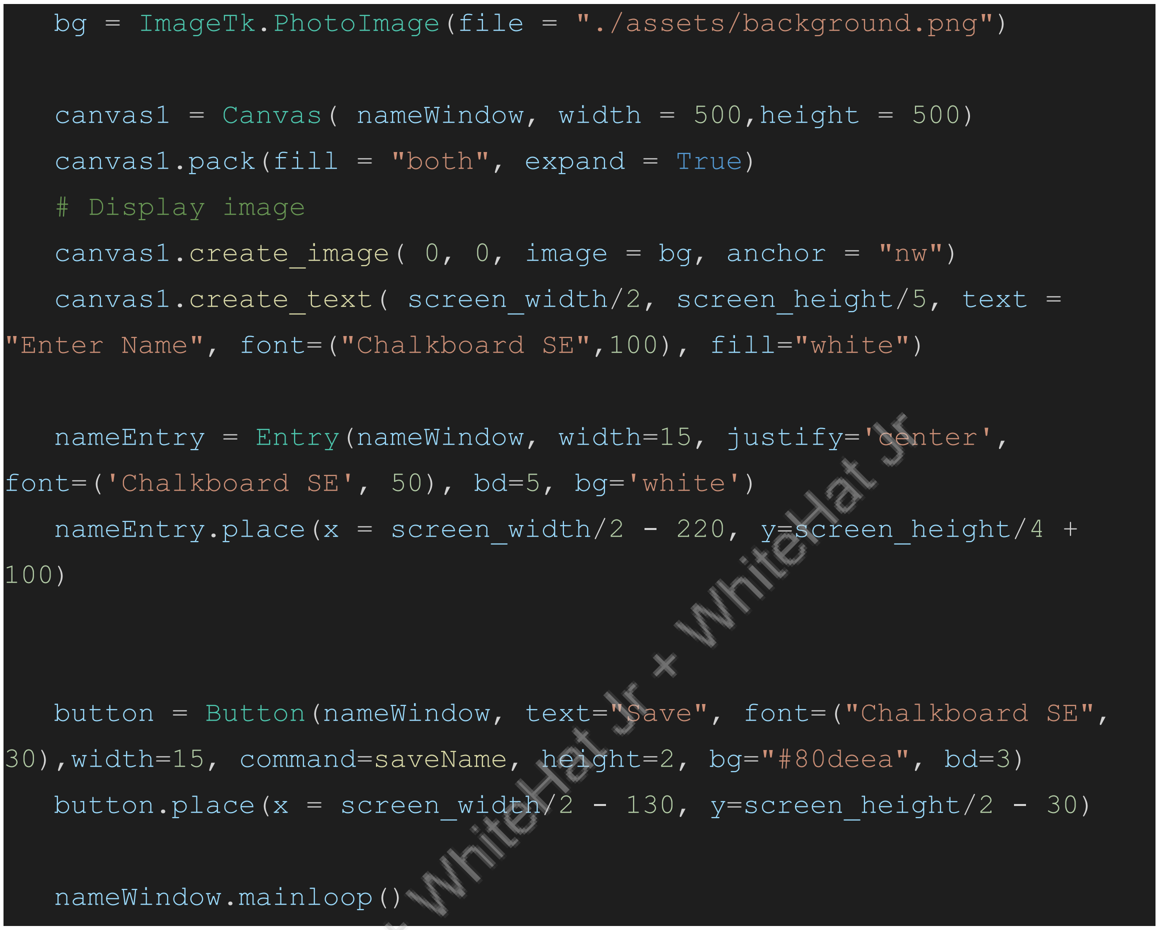


```
def setup():
    global SERVER
    global PORT
    global IP_ADDRESS

    PORT  = 5000
    IP_ADDRESS = '127.0.0.1'

    SERVER = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    SERVER.connect((IP_ADDRESS, PORT))
```

2. Create a function called **askName()**. Inside this function create a GUI using **tkinter** which will take the name from the user.

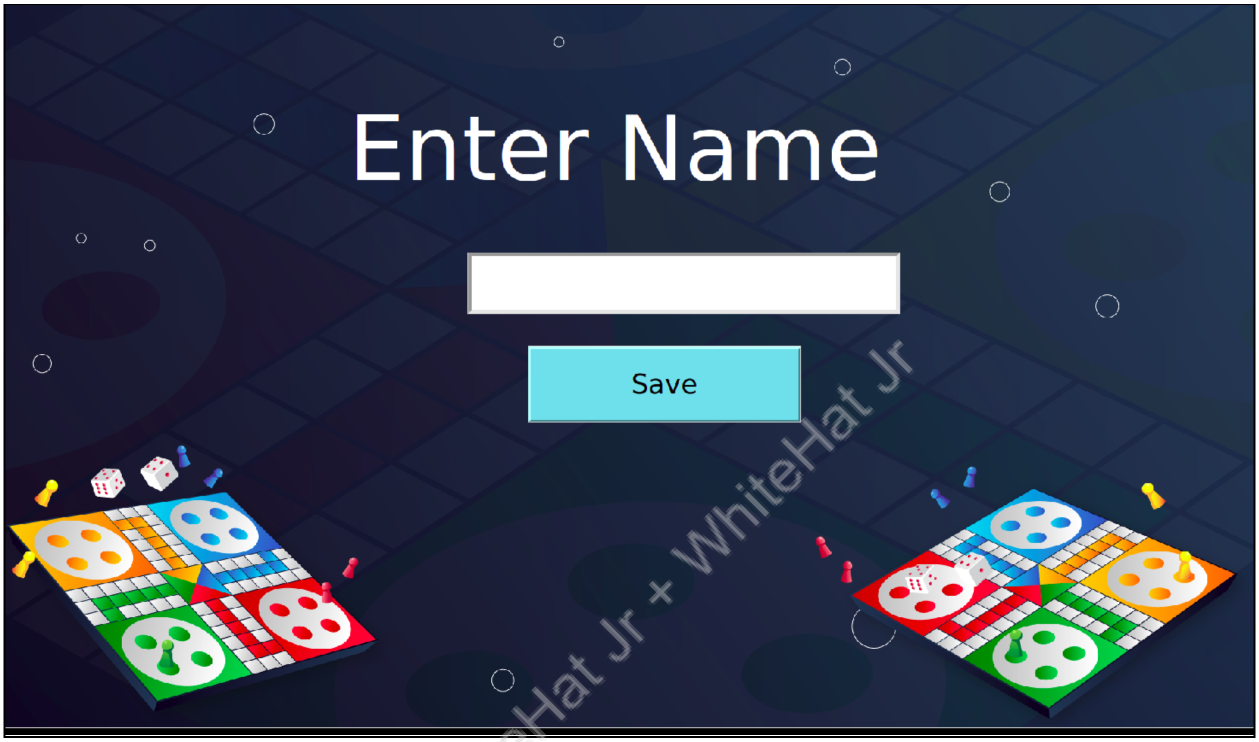


```
def askPlayerName():
    global playerName
    global nameEntry
    global nameWindow
    global canvas1

    nameWindow  = Tk()
    nameWindow.title("Ludo Ladder")
    nameWindow.attributes('-fullscreen',True)


    screen_width = nameWindow.winfo_screenwidth()
    screen_height = nameWindow.winfo_screenheight()
```



```
    bg = ImageTk.PhotoImage(file = "./assets/background.png")

    canvas1 = Canvas( nameWindow, width = 500,height = 500)
    canvas1.pack(fill = "both", expand = True)
    # Display image
    canvas1.create_image( 0, 0, image = bg, anchor = "nw")
    canvas1.create_text( screen_width/2, screen_height/5, text =
"Enter Name", font=("Chalkboard SE",100), fill="white")

    nameEntry = Entry(nameWindow, width=15, justify='center',
font=('Chalkboard SE', 50), bd=5, bg='white')
    nameEntry.place(x = screen_width/2 - 220, y=screen_height/4 +
100)


    button = Button(nameWindow, text="Save", font=("Chalkboard SE",
30),width=15, command=saveName, height=2, bg="#80deea", bd=3)
    button.place(x = screen_width/2 - 130, y=screen_height/2 - 30)

    nameWindow.mainloop()
```

3. Test the code to see the output.



Until now, you have created the server and added the GUI to accept the username from the user. Next, we'll create a function to take that we got from UI and send it to the server.

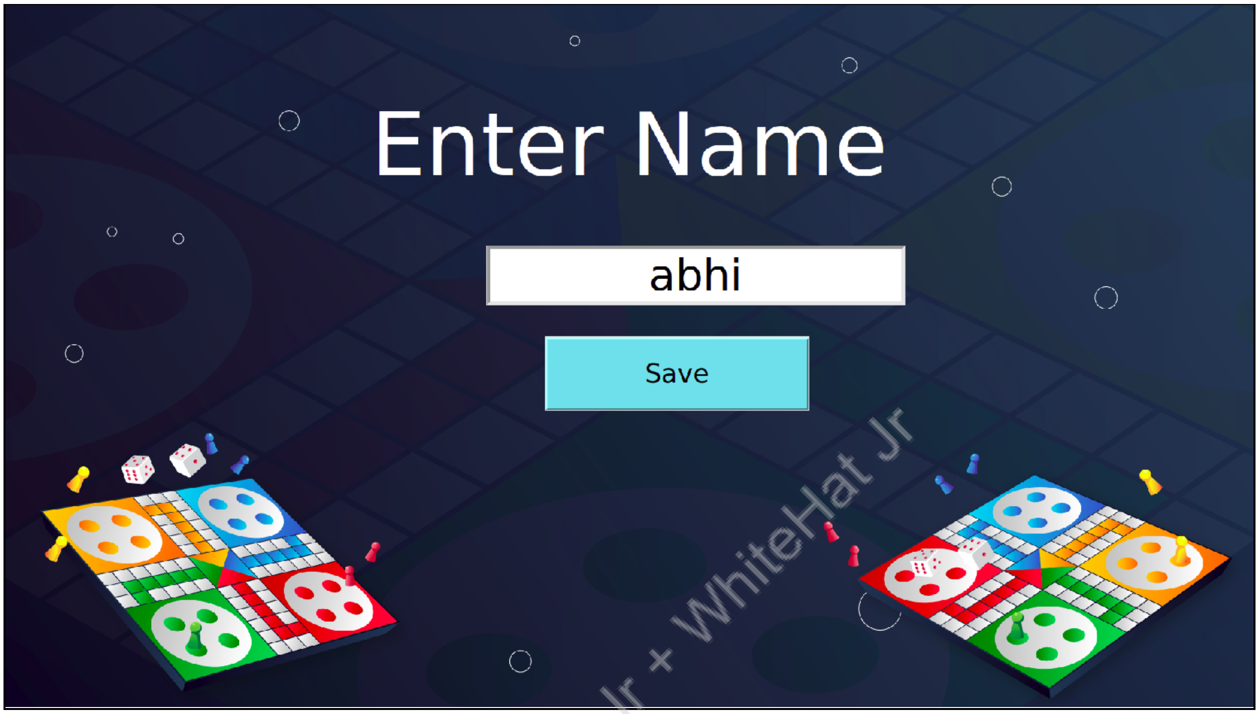
4. Now, write a **saveName()** function which will take the entered name and send it to the server by encoding it.

```
def saveName():
    global SERVER
    global playerName
    global nameWindow
    global nameEntry

    playerName = nameEntry.get()
    nameEntry.delete(0, END)
    nameWindow.destroy()

    SERVER.send(playerName.encode())
```

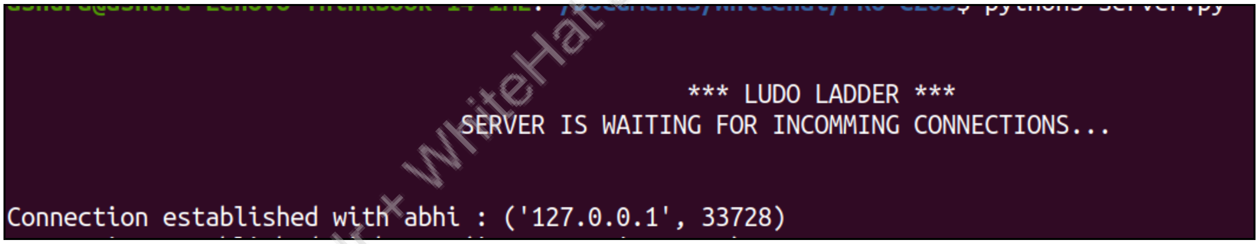
5. Write a **acceptConnection()** function that will accept the connection requests to connect with the server. Store the data of the client in a clients dictionary. And show the message of connection established with the clients.

```
def acceptConnections():
    global CLIENTS
    global SERVER

    while True:
        player_socket, addr = SERVER.accept()

        player_name = player_socket.recv(1024).decode().strip()

        if(len(CLIENTS.keys()) == 0):
            CLIENTS[player_name] = {'player_type' : 'player1'}
        else:
            CLIENTS[player_name] = {'player_type' : 'player2'}


        CLIENTS[player_name]["player_socket"] = player_socket
        CLIENTS[player_name]["address"] = addr
        CLIENTS[player_name]["player_name"] = player_name
        CLIENTS[player_name]["turn"] = False

        print(colored(f"Connection established with {player_name} :
{addr}", "cyan"))
```

6. Run and test the code.



abhi



```
                                  *** LUDO LADDER ***
                      SERVER IS WAITING FOR INCOMMING CONNECTIONS...


Connection established with abhi : ('127.0.0.1', 33728)
```

What's NEXT?

In the next class, we will be working on creating the game window using tkinter and also send and receive messages between the clients using the server.

Expand Your Knowledge:

Explore more about the python sockets through this link :
https://realpython.com/python-sockets/