

Afternoon Session:-

Date: / /

Date: 24th May

Name: Rakshita . C-U

Course: python

USN: 4AM18EC042.

Topic: Day 9 (Build a desktop Database

Sem & Sec: IV Sem & 'A' sec

GitHub Repository: Rakshita . Application

Report:-

```
from tkinter import *
window = Tk()
11. Label(window, text="Title")
11. Grid(row=0, column=0)
12. = Label(window, text="Author")
12. Grid(row=0, column=2)
13 = Label(window, text="year")
13. Grid(row=1, column=0)
14. = Label(window, text="ISBN")
14. Grid(row=1, column=2)
window.mainloop()
e1 = Entry(window, textvariable=
```

Frontend interface:-

Connecting the Frontend to Backend:

Fixing the Bug:

Exercise:

If you haven't already noticed, the program has a bug. When the listbox is empty and the user clicks the listbox, an Index error is generated in the terminal:

Why does this error happen?

Well, everything starts with the user clicking on the listbox. Clicking the listbox executes the following code:

list 1.bind('< list box select >', get_selected_row).
 The code calls the get_selected_row function:

Since the listbox is empty, list 1.curselection() will be an empty list with no items. Trying to access the first item on the list with [0] in line 3 will throw an error, because there is no first item in the list.

```
def get_selected_row(event):
```

```
    try:
```

```
        global selected_tuple
```

```
        index = list 1.curselection()[0]
```

```
        selected_tuple = list 1.get(index)
```

```
        e1.delete(0, END)
```

```
        e1.insert(END, selected_tuple[1])
```

```
        e2.delete(0, END)
```

```
        e2.insert(END, selected_tuple[2])
```

```
        e3.delete(0, END)
```

```
        e3.insert(END, selected_tuple[3])
```

```
        e4.delete(0, END)
```

```
        e4.insert(END, selected_tuple[4]).
```

Since the listbox is empty,
 list 1.curselection().

Creating a Standalone Executable Version of the program: