CT-2 Assignment

Name :- Shaik Afsar Basha

Reg no :- RA2111003010452

Section :- G1

6.) Write a SQL lite3 statement to create a table named as job including columns job-id, job-title, Min-salary, Max-salary, job-id column does not contain any duplicate value at time of insertion

A.)
```
import sqlite 3
con = sqlite3. connect ("Job.db")
con execute (" Create Table IF NOT EXISTS Jobs(JOB_ID varchar
        NOT NULL, JOB-TITLE varchar(35) NOTNULL, MIN-
        SALARY decimal (6,0), MAX-SALARY (6,0));" )

con. execute ("insert into job values (101, 'Manager', 15000,25000)")
```

7.) What is Event Driven Programming ?

A.) In programming, an event handler. is a callback routine that operates asynchronously once an event takes place. It dictates the action that follows the event. The programmer writes a code for action to take place. An event is an action that takes place when a user interacts with program.

8.) Write python code to print even numbers, odd numbers from list of given number using Imperative approach
[2, 5, 8, 23, 14, 37, 47, 18, 20, 36, 29]

A.)
```
num = [2, 5, 8, 23, 14, 37, 47, 18, 20, 36, 29]
odd = []
even = []
```

```
for i in num:
        if i%2==0:
                even.append(i)
        else:
                odd.append(i)

print(odd)
print(even)
```

9.) Explain implicit Parallelism

A.) -> Allow programmer to write their programs without concern about the exploitation of parallelism. Exploitation of parallelism is instead automatically performed by computer and/or the runtime system. In this way the parallelism is transparent to programmer maintaining the complexity of software development at same level of standard sequential programming.

-> Extracting parallelism implicitly is not an easy task. For imperative programming language, the complexity of program is almost prohibitively and allows positive results only for restricted sets of application.

10.) Find out geometry managers for follow options

a.) I have set of radio widgets and want place it next to each other along horizontal axis.

b.) I have button widget which I want to place it on specify coordinates denoting the screen

A.)
```
rb1 = tk.Radiobutton(root,text = "option 2", variable=var,value =1)
rb1.grid(row=0,column =0
rb1. grid (row=0, column =1)
~~b1.place (x=50, y=50~~ b1=tk.button(root, text="submit",
                                     command= root.destro
b1.place (x=50, y=50)
```

11.) Create table for student with following fields
a.) (Reg-no, student-name, sex, create table Dept with following fields (dept.no primary key, dept-name)
a.) Insert sample records
b.) Display student reg no, name, dept. name.
c.) Display name ending with ka
d.) Display female name students
e.) Display names in descending order.

A.) Import sqlite 3, con = sqlite3.connect("collegdb")
st = "create table (rno number(3)), name char(50), dept number(5),
gender char(20));"

```
con.execute(st)
st. "insert into student values (101,'ajay',1,'male');"
con.execute (st)
st = "create table dept (dno number(3) primary key, dname
        char(50));"
con.execute(st)
st. "insert into dept value(1,'cse');"
con.execute(st)
st = "select student.name, dept.name from student INNER JOIN
        dept ON S.dno = d.dno;"
con.execute(st);
st = "select rno, name, dept, spl from student;"
con.execute(st);
st = "select name from student where name like '%ka';"
con.execute(st)',
st = "select * from students where gender = 'female';"
con.execute(st)
st = "select * from students order by name desc;"
con.execute(st)
con.close()
```

11.) Explain Parallel programming and write multithreaded
b) application python

A.) Multitasking in general is capability of performing multiple tasks simultaneously. Multithreading refers to concurrently executing multiple threads by rapidly switching the control of CPU b/w threads. Python Global Interpreter Lock limits one thread at a time even if machine contain multiple processors.

Application:-

```
import threading
from threading import *
def  calculate_square(num):
     print("Calculate square of given number")
     for n in num:
          print(f'The Square of {n} is:' n*n)

def  calculate_cube(num):
     print("Calculate cube of given number")
     for n in num:
          print(f"The cube of {n} is:' n*n*n)

lst = [2,4,6,8,10]

thread1=
threading.Thread(target=calculate_square,args=(lst,))
threading.Thread(target=Calculate_cube, args=(lst,))
thread1.Start()
thread1.Start()
thread 1.join()
thread 2.join()
```

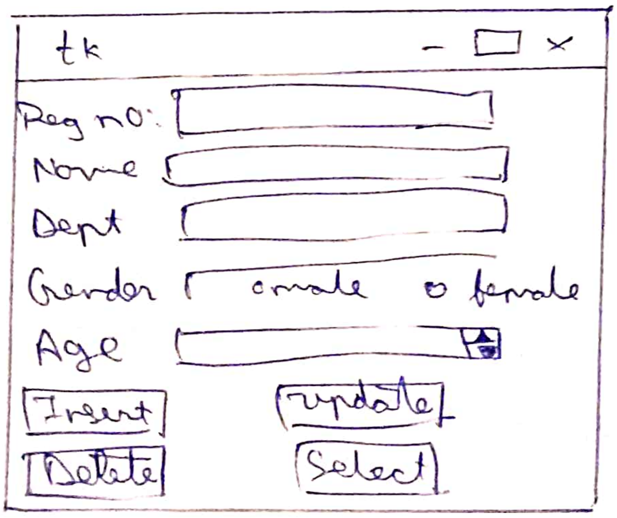
12.) Create calender using Tkinter by showing data month
a.) and year with scroll down menu fix the particular
data month and year and press the click button
to show message of clicked date, month, year.

A.) from tkinter import *

```
import calendar
from datetime import date
def printCalender():
    month=int(month_box.get())
    year=int(year_box.get())
    output_calender=calendar.month(year,month)
    calendar field.delete(1.0,'end')
    calendar_field.insert('end',output_calendar)

def reset():
    calendar_field.delete(1.0,'end')
    month_var.set(current_month)
    year_var.set(current_year)
    month_box.configure(textvariable=month_var)
    year_box.configure(textvariable=year_var)

def close():
    gui Window.destroy()

header_frame=Frame(guiWindow)
entry.frame=Frame(gui Window)
result frame=Frame(guiWindow)
button_frame=Frame(gui Window)
headerframe.pack(expand="True, fill="both")
entry_frame.pack(expand=True, fill="both")
result.frame.pack(expand=True, fill="both")
button_frame.pack(expand=True, fill="both")
```

```
header_label =Label(header_frame, text="CALENDAR")
header_label.pack(expand=True ,fill ="both")
month_label= Label(entry_frame,text="Month")
year_label = Label(entry_frame, text="year",
              font =("arial", "20", "bold"), fg="#000000")
month_label.place(x=30,y=0)
year_label.place(x=275,y=0)
month_var = IntVar(entry_frame)
year_var= Intvar(entry_frame)
current_month = date.today().month
current_year=date today() year
month_var.set(current_month)
year_var set(current_year)
month_box =spinbox(entry_frame, from 1,to 12, width="10",
textvariable=month_var ,font=('arial','15'))
if _name__ == "_main_"
    guiwindow =TK()
    guiwindow.title("GUI Calendar")
    guiWindow.geometry('500x550')
    ~~guiWindow.geometry('~~
    gui.Window.resizable (0,0)
```

12 b.)



A)
```
from tkinter import *
root = Tk()
def insert():
        pass
l1 =Label(root ,text= "reg no").grid(row=0,column=0)
eid=StringVar()
e1 = Entry(root, textvariable= eid).grid(row=0 ,column=1)
l2 = Label(root, text="Name").grid(row=1, column=0)
ename =StringVar()
e2 = Entry (root,text Variable= ename). grid(row=1,Column=1)
l3 = Label (root, text = "Dept").grid (row=2, Column=0)
job = String Var( )
e3= Entry (root, textVariable= job). grid (row=2, Column=1)
l4 =Label(root, text="Gender").grid (row= 3, column=0)

v= IntVar()
r1=Radio button(root, text= "Male", value=1,Variable=v).
    grid(row=3, Column=1)
r2= Radiobutton (root, text ="Female", Value = 2, Variable= v)
    grid (row=3, column= 0 )

S= Int Var()
S1 = Spinbox(root, from_ =18, to=50 ,textvariable=s).grid
        (row 4, column=1)
b1 = Button(root,text="Insert",command=insert). grid
        (row=5, column=0)
b2 = Button( root,text = "update", command= insert )
b3 = Button(root, text="select", command=insert )
b4 =Button (root,text="delete", command=insert )
root. mainloop()
```

MCQs

1.) The idea of imperative programming paradigm imitates
Object oriented programming

2.) which is false regarding dependent types.
They allows us to write programs and know they are correct only after ~~p~~ running them

3.) If "wait for graph" for a set of processes contain a cycle, so it means
There is a chance for deadlock to occur.

4.) >>> P = Pool(5) - pool of five worker process.

5.) "Spinbox()" provides range of values to user ~~out of~~

6.) "Trigger Function" decides what code to run when there are specific event occurs, which are used to select which event handler to use for event

7.) Creating "submit" syntax

```
Parent = TK()
submit = Button(parent, text="submit").grid(row=2, column=c
```

8.) "<thread> <threading>" are two modules offers to implement in python programming

9.) "expand" property of Geometry manager pack allows the widget to fill any space not otherwise used in widget's parent.

10.) Syntax which does not insert new record into sqlite3
Insert into phonebook(phoneno, fname, lname, email) values (123, 'x'; G', 'xyz@gmail.com');

Name: SHAIK AFSAR BASHA
Reg no:- RA2111003010452
Section :- G1.