



NAME	SRN
MEELA DEEPTI	PES2UG19CS227
M SATVIKA	PES2UG19CS207
LAKSHMI NARAYAN P	PES2UG19CS200

## Simple User Interface for Front End

There were two reasons to choose python as the frontend choice.

1. It is very easy to connect to a MySQL database. (read below)
2. There are many ways to create python GUI like tkinter,wxpython and Jpython out of which we chose tkinter It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with

tkinter is the fastest and easiest way to create GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter GUI, we need only four steps:

- a.Importing the module – tkinter
- b.Create the main window (container)
- c.Add any number of widgets to the main window
- d.Apply the event Trigger on the widgets.

## DOCTOR

tk

**WELCOME, WHO ARE YOU?**

DOCTOR

PATIENT

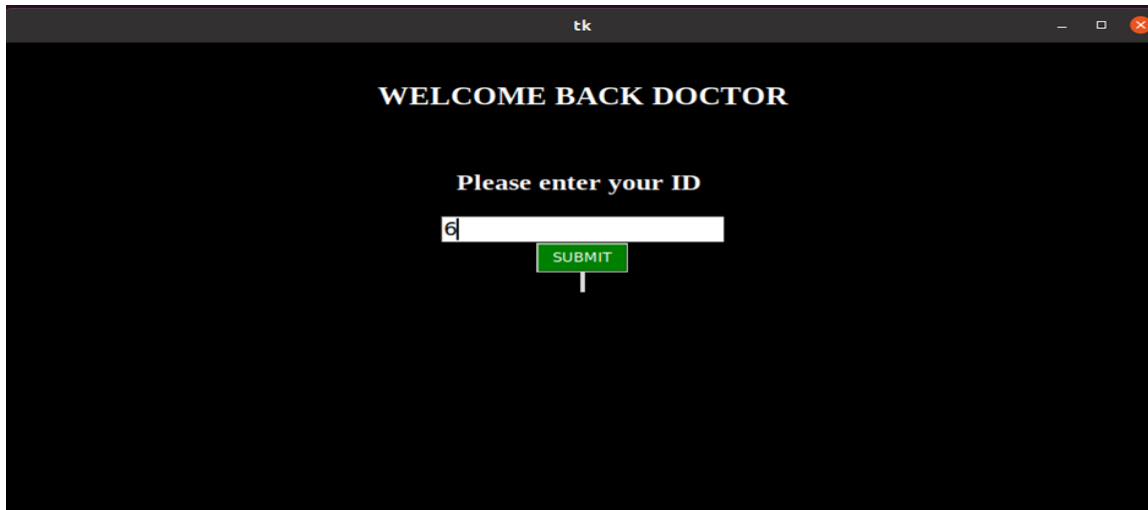
tk

**WELCOME DOCTOR**

DOCTOR LOGIN

DOCTOR REGISTER

## DOCTOR LOGIN

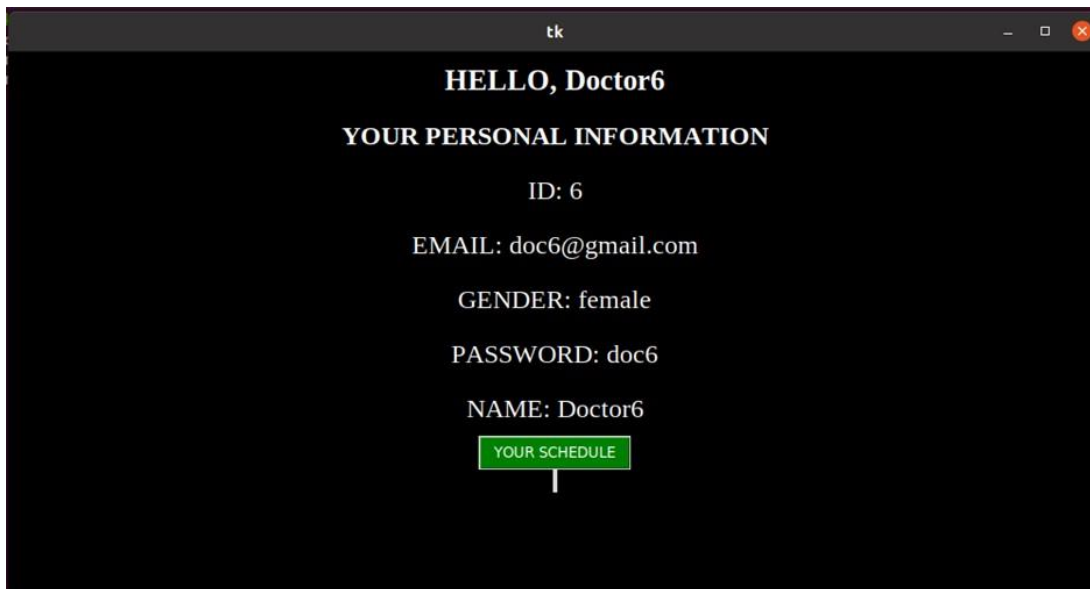


tk

**WELCOME BACK DOCTOR**

Please enter your ID

SUBMIT



tk

**HELLO, Doctor6**

**YOUR PERSONAL INFORMATION**

ID: 6

EMAIL: doc6@gmail.com

GENDER: female

PASSWORD: doc6

NAME: Doctor6

YOUR SCHEDULE

tk

## YOUR SCHEDULE, Doctor6

Start time: 8:00:00

End time: 17:00:00

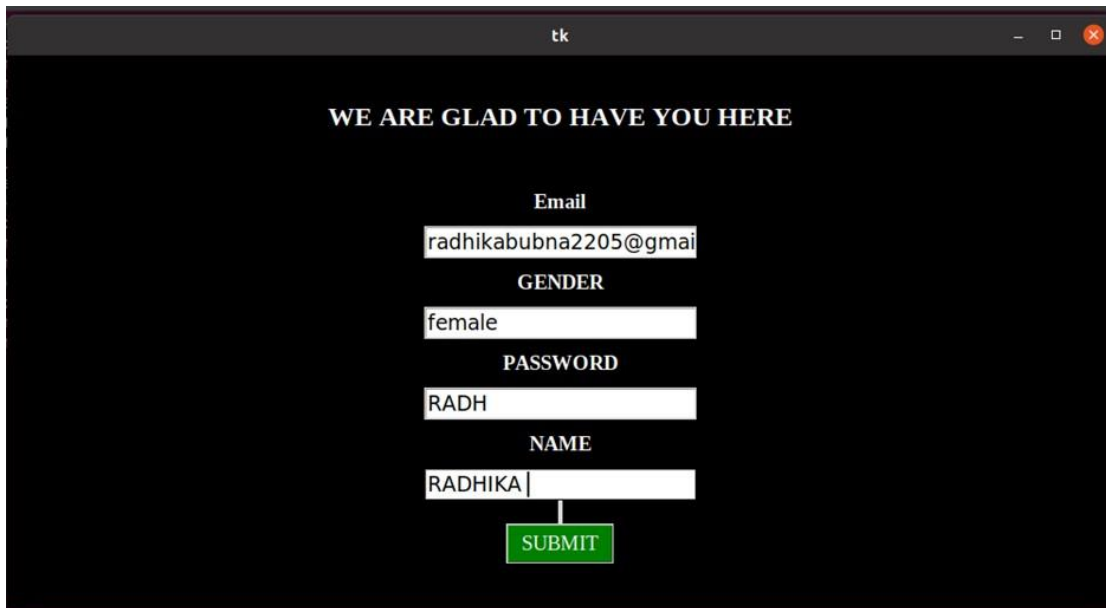
Break time: 12:00:00

Dn: Wednesday

PATIENT DETAILS

## YOUR PATIENT DETAILS

## DOCTOR REGISTER



tk

WE ARE GLAD TO HAVE YOU HERE

Email

radhikabubna2205@gmai

GENDER

female

PASSWORD

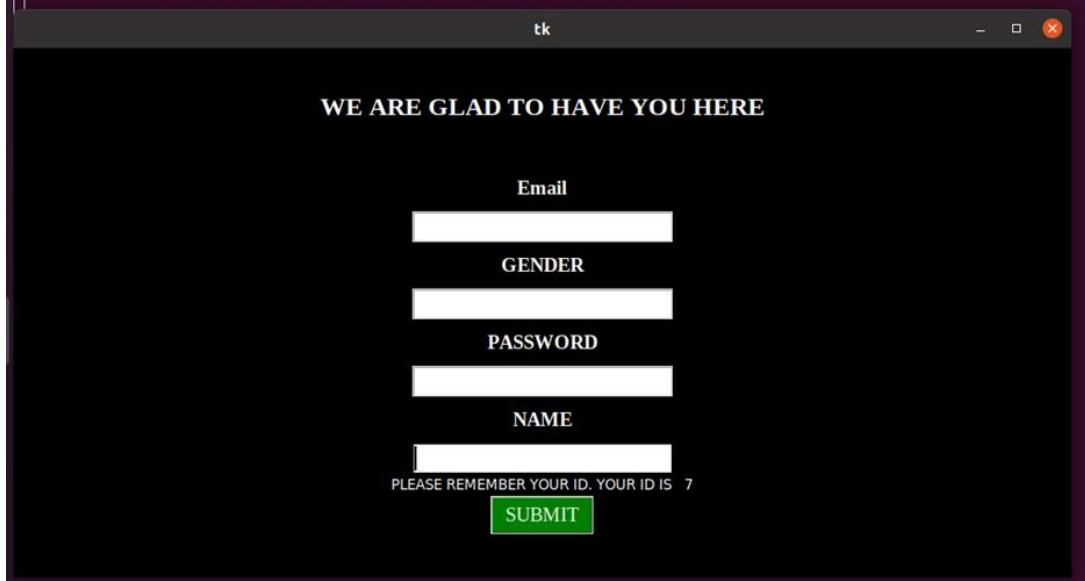
RADH

NAME

RADHIKA

SUBMIT

```
(1, 'doc1@gmail.com', 'female', 'doc1', 'Doctor1')
(2, 'doc2@gmail.com', 'male', 'doc2', 'Doctor2')
(3, 'doc3@gmail.com', 'male', 'doc3', 'Doctor3')
(4, 'doc4@gmail.com', 'female', 'doc4', 'Doctor4')
(5, 'doc5@gmail.com', 'male', 'doc5', 'Doctor5')
(6, 'doc6@gmail.com', 'female', 'doc6', 'Doctor6')
(7, 'radhikabubna2205@gmail.com', 'female', 'radhika123', 'RADHIKA')
```



tk

WE ARE GLAD TO HAVE YOU HERE

Email

GENDER

PASSWORD

NAME

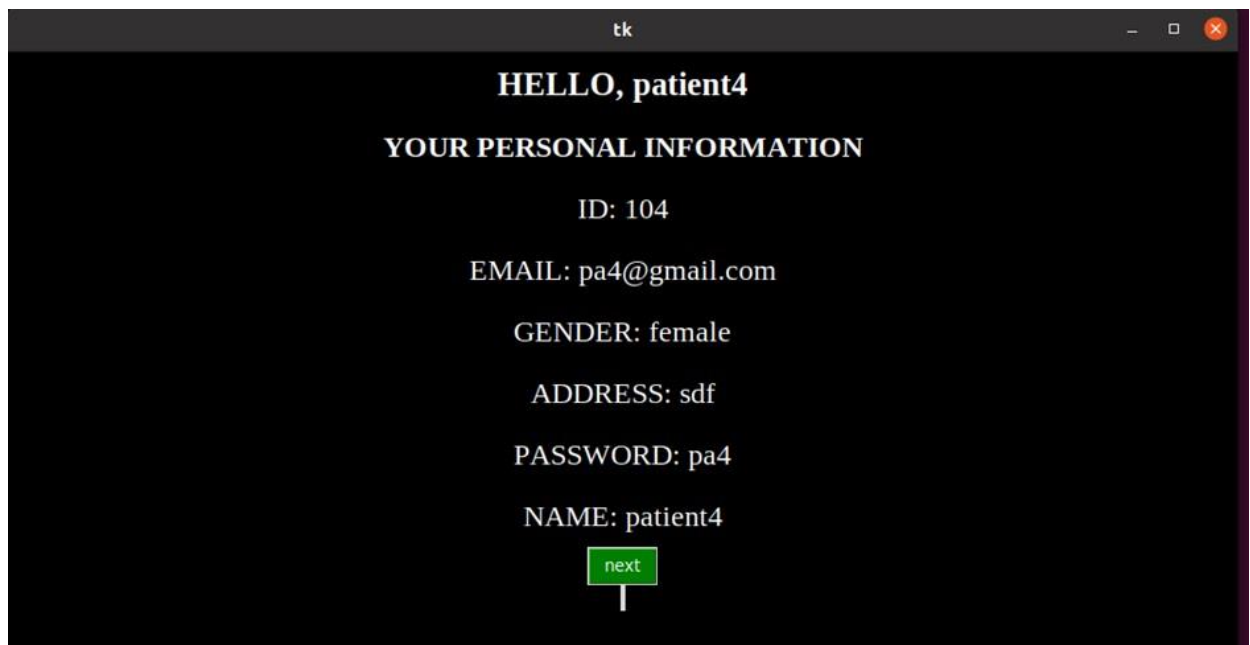
PLEASE REMEMBER YOUR ID. YOUR ID IS 7

SUBMIT

## PATIENT



## PATIENT LOGIN



WELCOME BACK !

Please enter your ID

SUBMIT

## YOUR APPOINTMENT DETAILS

DOCTOR ATTENDING YOU: Doctor1

START TIME: 22:00:00

END TIME: 23:30:00

STATUS: Done

tk

## YOUR APPOINTMENT DETAILS

DOCTOR ATTENDING YOU: Doctor1

START TIME: 22:00:00

END TIME: 23:30:00

STATUS: Done

|

## PATIENT LOGIN

```
(101, 'pa1@gmail.com', 'pa1', 'Patient1', 'Andhra Pradesh', 'Female')
(102, 'pa2@gmail.com', 'pa2', 'Patient2', 'ghsd', 'male')
(103, 'pa3@gmail.com', 'pa3', 'patient3', 'sdjfb', 'male')
(104, 'pa4@gmail.com', 'pa4', 'patient4', 'sdf', 'female')
(105, 'pa5@gmail.com', 'pa5', 'patient5', 'dfg', 'male')
(106, 'pa6@gmail.com', 'pa6', 'patient6', 'ddfs', 'female')
(107, 'tina@gmail.com', 'tina123', 'TINA', 'mumbai', 'female')
```

tk

## WELCOME

Email

GENDER

PASSWORD

NAME

ADDRESS

PLEASE REMEMBER YOUR ID. YOUR ID IS 107

SUBMIT



## CASE OF WRONG ID



tk

WELCOME BACK DOCTOR

Please enter your ID

995

SUBMIT

WRONG ID. QUITTING NOW

## Schema Changes

```
mysql> ALTER TABLE Patient ADD age int NOT NULL;
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Patient;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	
email	varchar(50)	NO		NULL	
password	varchar(30)	NO		NULL	
name	varchar(50)	NO		NULL	
address	varchar(60)	NO		NULL	
gender	varchar(20)	NO		NULL	
age	int(11)	NO		NULL	

7 rows in set (0.00 sec)

```
mysql> ALTER TABLE Appointment ADD fee int NOT NULL;
Query OK, 0 rows affected (0.07 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Appointment;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	
date	date	NO		NULL	
starttime	time	NO		NULL	
endtime	time	NO		NULL	
status	varchar(15)	NO		NULL	
fee	int(11)	NO		NULL	

6 rows in set (0.00 sec)

```
mysql> ALTER TABLE Doctor ADD salary int NOT NULL;
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Doctor;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	
email	varchar(50)	NO		NULL	
gender	varchar(20)	NO		NULL	
password	varchar(30)	NO		NULL	
name	varchar(50)	NO		NULL	
salary	int(11)	NO		NULL	

6 rows in set (0.01 sec)

## **Database Migration And Support**

If due to performance issues, or any other requirements, if we need to migrate current database then we would prefer Neo4j.

We can get the best out of graph database Neo4j when

1. large amounts of data. According to a survey Hospitals produce 50 petabytes of data per year.
2. has intricately structured high value relationships
3. and is constantly evolving.

Since our data follows all the three points, we would probably migrate it to the neo4j graph database.

For database connectivity, the module installed was MYSQL connector. We use connect function to connect

with the database and creating a cursor. A cursor is then further used for executing MYSQL queries.