

GUESSER- A FINE DOCUMENTATION

We are so happy. Since we are learning a crispy clear and shorter style of programming. Wondering what is it? . It's none other than our fast-emerging programming language --> "**Python**"

Got a good opportunity to take part in Python-75 hackathon organized by Talent Accurate that is being leaded by Raja sir.

We have done a small project called "Department Guesser". Anyone may wonder because just by using a single module in python, one can do a mini project that assures value to his/her knowledge.

SCENARIO:

- Nothing serious, it's just like a game
- Imagine 2 members are meeting suddenly
- They are Angeline and Jacqueline
- They are school friends
- From school days, both of them are so playful
- Now they are studying in different colleges but they are doing the same historic-choice of study .i.e., **engineering**
- Angeline said about her details of alma mater
- Jacqueline bet Angeline to guess her department
- Meanwhile they played this guessing game with their other school friends too and enjoyed a lot
- That's what the play here

CONSIDERATIONS:

- School friends list with names and their departments as base data in DB
- Users names are recorded along with the guesses
- **Input:** random name among the one of the friends
- **Output:** guessed department with chances used

Our core goal is achieved by python's most interesting module random. It comprises very valuable functions. Some of them are,

- randint(start_value, stop_value)
--> Eg: randint(3, 16) outputs randomly the numbers between 3 and 16

- `randrange(start_value,stop_value,skip_value)`
-->Eg: `randrange(3,16,2)` outputs randomly the odd numbers between 3 and 16
- `random.choice(list)`
-->Eg: from the list,randomly outputs something

We used `random.choice()` alone.

STEPS WE FOLLOWED:

- Import random module so that we can access the functions of random module.
`import random`
- We have created sqlite database that contains a table containing some of your college friends and their respective departments. (*As a start, you can also use dictionary to store this base data*)

Even sqlite is not that much as an herculean task. DB connectivity is also so simple in python.Let's walkthrough on that also in detail,

--> STEPS FOR SQLITE:

For using SQLITE database in python, first we are importing sqlite3 module.

`import sqlite3`

- Then connect to a database- if already existing it will be connected to that DB else a new DB is created.

`con.connect('test.db')`

`#con --> connection object`

`#test.db is the DB file created or will be created`

- Then a table named students is created by executing the query in the python program.

`con.execute('CREATE TABLE STUDENTS(name text not null, dept text not null);')`

`#execute is the function that holds the query to be passed as it's parameter.`

- Then insertion can be done by executing the insert query.

`con.execute('INSERT INTO STUDENTS(name,dept) VALUES('Ram','CSE');')`

Thus, by using sqlite now you can created your database with the base data of records containing your college mates' names and their respective departments.

- After base data,the process is simple.
- Using the connection object, fetch the data to be cross-checked as a dictionary in python.

```
c=con.execute("select * from students")  
#selecting the data
```

- Dictionary is just a data structure that maps key to it's value , which is more important in our case's mapping too.

```
dict1={} #dictionary  
for i in c:  
    dict1[i[0]]=i[1]  
#translocating the data as a key-value pair in dictionary
```

- Then after creating a dictionary, split that into a list of keys and to a list of values, so that the random choice can be displayed from the list.

```
key = list(dict1.keys()) #keys of dictionary in 'key' list  
val = list(dict1.values()) #values of dictionary in 'val' list
```

- Then use the main function of **random.choice(list)** outputs the random value among the names of your friend which are the keys.
- Find it's index then match it with it's value.
- Get user input - user inputs the department as per their guess.
- The user input is mapped with the value list that is the department list.
- If they matches, then the guess is correct.
- If not,then they can try but their tries would be counted.

