



# **Prepared Statement and Input Validation using Regular Expression**

Sumaya Altamimi  
442203026

---

## Introduction

I wrote a program in **Python** that stores valid contact information (Name, Phone number, and Address). The code and all files are attached. Also, the guidelines for installing and running the project is described in the next section. For input validation, I used Regular Expression. If the contact information is valid, I store it in a **MySQL** DB using prepare statement. You as a user will be able to ADD (insert valid contact in DB), DEL (delete contact either by phone number or name), and LIST (List all contacts in DB).

## Prepared Statement

Using question marks instead of actual values like this:

```
person_insert_query = "INSERT INTO Person  
(name,phone,address) VALUES(?,?,?);"
```

```
delete_query = """DELETE FROM Person WHERE phone  
= ? ; """
```

```
delete_query = """DELETE FROM Person WHERE name  
= ? ; """
```

And then sending the query along with the variables from user input after they are validated using regular expression.

```
Database.execute_add_query(connection,  
person_insert_query,(name, phone, address))
```

```
Database.execute_add_query(connection,  
delete_query,(to be deleted,))
```

## Regular Expressions

The regular expressions used in this project are listed below for each contact information item:

```
name_regex = r'((([a-zA-Z]{2,})?(\s?[a-zA-Z]\' {1}[a-zA-Z]{2,})?\,\s?([a-zA-Z]{2,})?(\-|\s)?[a-zA-Z]{0,})?\.\?)$'
```

```
phone_regex = r'^[1-3]{0,1}([+][1-3]{1,2})?\s{0,1}((([()]{0,1}(703){0,1}[)])|([()]{0,1}(212){0,1}[)])\s{0,1}[-\s\.0-6]{1}[-\s\.0-9]{4,}$'
```

```
address_regex = r'^[0-9]{3,4}[a-zA-Z]?\s[0-9]?[a-zA-Z]+\s[a-zA-Z]+\,\s[a-zA-Z]*[#\s\d]?\.\s[a-zA-Z]+\,\s[a-zA-Z]+\s[0-9]{5}\-?[0-9]{0,4}'
```

## Assumptions

Based on the given examples and test cases in the assignment problem, I have made some assumptions for each contact information item as follow:

### Name

- 1- No numbers allowed.
- 2- No special characters allowed other than: dot, comma, hyphen, and single quote.
- 3- The name starts with letters only.
- 4- The first name is at least 2 letters.
- 5- The first name has no special characters.
- 6- The name is at most four words/parts.
- 7- Dot is only allowed at the last part of the name.
- 8- Comma is only allowed after the first name.

- 9- Hyphen is only allowed after the first name.
- 10- Quote is only allowed after the first letter of the second name.

## Phone

- 1. No letters allowed.
- 2. No special characters allowed other than dot, hyphen, parentheses, and plus sign.
- 3. The phone number has at most five parts.
- 4. Phone number is at least 5 in length.
- 5. If phone number has plus sign, then it should be at the beginning.
- 6. If phone number has plus sign, then it should be followed by at most two numbers and space.
- 7. If phone number has plus sign, then it should be followed by either 1, 2, or 3.
- 8. If the phone number starts with digit, it should be less than 7.
- 9. If phone number has parenthesis, they should include either 703 or 21.
- 10. If phone number has parenthesis, and there is a number before the parentheses, it should be less than four.
- 11. If phone number has dot, then it should be after the first five numbers.

## Address

- 1. The address should start with 3 or 4 digits only.
- 2. The address starts with digits that can be followed by one character only in the same word (No space).
- 3. The second word might begin with a number.
- 4. Type of special characters allowed are limited only to dot, hash, comma, and hyphen.
- 5. Dot and hash are allowed only in specific locations: after the second/third word.
- 6. The address should end with 5 digits that might be followed by hyphen and four digits.
- 7. There should be at least four words.

## Project Guidelines

In this section I provide step by step in how to run the code.

### Requirement before running the project

- You should have **Python** installed
- You should have **MySQL** installed
- You should replace the **password** of the localhost to your own computer password as follow:
  1. Unzip the project folder named pythonProject
  2. Open src folder
  3. Open main.py file
  4. In the fourth line, write your password instead of 'mypassword'.
  5. Save and close the file.

```
1 from database import Database
2 import re
3 # Pass your computer password to the pw variable
4 pw = 'mypassword'
5
```

### Quick Start

- Open The terminal/ Command Line
- Go to the same path of the project, and open the src file by typing the following command:  
`cd <your path>/pythonProject /src/`
- Type the following command to run the project:  
`python3 test.py`
- The result should look like this:

```
MySQL Database connection successful
Query successful
MySQL Database connection successful
Database created successfully
MySQL Database connection successful
Query successful
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
```

- Select any number from 1-4 and start testing the project

## Test Cases

In this section, I provide screenshots of the code while it is running with some examples of accepted input and other examples of invalid input for each contact information item.

### Case1: Add a valid contact information

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
1
MySQL Database connection successful
Please enter your name: Sumaya
Please enter your phone: 12345
Please enter your address: 795 E DRAGRAM Drive, TUCSON, AZ 85705-7598
Query successful
after execute
```

#### Case2: Add an Invalid name

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
1
MySQL Database connection successful
Please enter your name: Ron O''Henry
Oops! Please enter a valid name: Bruce Schneier
Please enter your phone: (703)111-2121
Please enter your address: 795 E DRAGRAM Dr., TUCSON, AZ 85705
Query successful
after execute
```

#### Case3: Add an invalid phone number

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
1
MySQL Database connection successful
Please enter your name: Schneier, Bruce
Please enter your phone: 123
Oops! Please enter a valid phone number: 123-1234
Please enter your address: 770W DRAGRAM Drive # 321, TUCSON, AZ 85703
Query successful
```

#### Case4: Add an invalid address

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
1
MySQL Database connection successful
Please enter your name: Schneier, Bruce Wayne
Please enter your phone: +1(703)111-2121
Please enter your address: E DRAGRAM Drive, TUCSON, AZ 85705-7598
Oops! Please enter a valid address: 2430 7th St, Washington, DC 20521-2430
Query successful
```

### Case5: List all

\*note: Because of long values it is compressed. However, if you are interested to see the full data you can run the project in other platform.

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
3
In listAll
MySQL Database connection successful
  id  ...                                address
0   1  ...  795 E DRAGRAM Drive, TUCSON, AZ 85705-7598
1   2  ...                795 E DRAGRAM Dr., TUCSON, AZ 85705
2   3  ...  770W DRAGRAM Drive # 321, TUCSON, AZ 85703
3   4  ...      2430 7th St, Washington, DC 20521-2430
```

### Case6: Delete by name

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
2
Do you prefer to delete by phone press 1, other wise press 2:
2
In delPerson
MySQL Database connection successful
Please enter your name: Sumaya
Query successful
  id  ...                                address
0   2  ...                795 E DRAGRAM Dr., TUCSON, AZ 85705
1   3  ...  770W DRAGRAM Drive # 321, TUCSON, AZ 85703
2   4  ...      2430 7th St, Washington, DC 20521-2430

[3 rows x 4 columns]
```



## Case7: Delete by phone

```
Please Select one of the following:
    1. Add Person
    2. Delete Person
    3. List all persons
    4. Exit
2
Do you prefer to delete by phone press 1, other wise press 2:
1
In delPerson
MySQL Database connection successful
Please enter your phone: 123-1234
Query successful
    id  ...                                address
0     2  ...      795 E DRAGRAM Dr., TUCSON, AZ 85705
1     4  ...  2430 7th St, Washington, DC 20521-2430

[2 rows x 4 columns]
```

## References

- [1] <https://www.freecodecamp.org/news/connect-python-with-sql/>
- [2] <https://regex101.com/>
- [3] <https://stackoverflow.com/questions/33881152/validate-user-input-using-regular-expressions>