

Adhish Bahl

2347203

5MCA B

Lab 10: Apply regular expression for form validation. Create your domain-form using Tkinter Module.

1. Form should contain Text box [For Name, Email Id, Phone number], Dropdown [for Gender], Spinbox [for Year/DoB] and other necessary widgets required for your domain.

2. Validate Your Name, Email Id, Phone number in the form.

```
import tkinter as tk
import re

def submit_form():
    name = name_entry.get()
    email = email_entry.get()
    phoneNo = phoneNo_entry.get()
    role = role_var.get()
    age = age_spinbox.get()
    gender = gender_var.get()
    policies = policies_var.get()

    flag = 0
    error = ""

    if re.fullmatch('[A-Za-z]{2,25}([A-Za-z]{2,25})?', name):
        print("")
    else:
        flag = 1
        error = error + "\nInput a Valid Name!"

    if re.fullmatch('([A-Za-z0-9]+[._-])*[A-Za-z0-9]+@[A-Za-z0-9-]+(\.[A-Z|a-z]{2,})+', email):
        print("")
    else:
        flag = 1
        error = error + "\nInput a Valid E-mail!"

    if re.fullmatch('^[6-9]\d{9}$', phoneNo):
        print("")
    else:
        flag = 1
        error = error + "\nInput a Valid Phone Number!"

    if(role == ""):
        flag = 1
        error = error + "\nInput a Role"
```

```

if(age == ""):
    flag = 1
    error = error + "\nEnter your Role"

if(gender != "Male" and gender != "Female"):
    flag = 1
    error = error + "\nSelect your Gender"

if(policies == 1):
    if(flag==1):
        print("User wasn't able to register. The input vales are
invalid.\n", error)
    else:
        print("User registered Successfully!")
        print(f"\nName: {name}")
        print(f"E-mail: {email}")
        print(f"Phone Number: {phoneNo}")
        print(f"Role: {role}")
        print(f"Age: {age}")
        print(f"Gender: {gender}")
        print(f"Policies: {policies}")
else:
    print("Accept the privacy policies!")

root = tk.Tk()
root.geometry("400x400")
root.title("New User Form")

name_label = tk.Label(root, text="Name:", font=("Arial", 15))
name_label.place(x=20, y=20)
name_entry = tk.Entry(root, font=("Arial", 15))
name_entry.place(x=140, y = 20)

email_label = tk.Label(root, text="E-mail:", font=("Arial", 15))
email_label.place(x=20, y=60)
email_entry = tk.Entry(root, font=("Arial", 15))
email_entry.place(x=140, y = 60)

phoneNo_label = tk.Label(root, text="Phone No:", font=("Arial", 15))
phoneNo_label.place(x=20, y = 100)
phoneNo_entry = tk.Entry(root, font=("Arial", 15))
phoneNo_entry.place(x=140, y = 100)

role_label = tk.Label(root, text="Role:", font=("Arial", 15))
role_label.place(x=20, y = 140)
role_var = tk.StringVar()
role_options = ["Customer", "Admin", "Delivery Boy", "Supplier"]
role_menu = tk.OptionMenu(root, role_var, *role_options)
role_menu.config(width=16, height=1, font=("Arial", 15))

```

```

role_menu.place(x=140, y = 135)

age_label = tk.Label(root, text="Age:", font=("Arial", 15))
age_label.place(x=20, y = 180)
age_spinbox = tk.Spinbox(root, from_ = 16, to = 90, font=("Arial", 15),
width=18)
age_spinbox.place(x=140, y = 180)

gender_label = tk.Label(root, text="Gender:", font=("Arial", 15))
gender_label.place(x=20, y = 220)
gender_var = tk.StringVar()
male_radio = tk.Radiobutton(root, text="Male", variable=gender_var,
value="Male", font=("Arial", 15))
female_radio = tk.Radiobutton(root, text="Female",
variable=gender_var, value="Female", font=("Arial", 15))
male_radio.place(x=140, y = 220)
female_radio.place(x=230, y = 220)

policies_var = tk.IntVar()
policies_button = tk.Checkbutton(root, text = "By registering, I accept
all the rules and policies of the company.", variable = policies_var,
onvalue = 1, offvalue = 0, height = 2, width = 50)
policies_button.place(x=20, y=270)

submit_button = tk.Button(root, text="Submit", command=submit_form,
font=("Arial", 15))
submit_button.place(x=140, y=330)

root.mainloop()

```

Accept the privacy policies!

User wasn't able to register. The input vales are invalid.

Input a Valid Name!

Input a Valid Phone Number!

User registered Successfully!

Name: Adhish

E-mail: adhish@gmail.com

Phone Number: 9462849725

Role: Admin

Age: 21

Gender: Male

Policies: 1

Lab 11: Perform the Exploratory Data Analysis on your domain-based dataset and demonstrate the retrieved insights using “Matplotlib” modules. hidden insights using appropriate plots (graphs) [Usage of line plot and scatter plot are mandatory]

Import Libraries

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Importing Dataset

```
df = pd.read_csv("./Nike_shoes.csv")
```

Displaying Dataset Basic Information

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 259 entries, 0 to 258
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   Unnamed: 0            259 non-null   int64
 1   title                 259 non-null   object
 2   sub_title            259 non-null   object
 3   brand                259 non-null   object
 4   color_breif         259 non-null   object
 5   fullPrice            259 non-null   float64
 6   currentPrice         259 non-null   float64
 7   country              259 non-null   object
 8   availability          259 non-null   bool
 9   publish_date         259 non-null   object
10   created_date         259 non-null   object
11   discount_amount      259 non-null   float64
12   asof_date            259 non-null   object
dtypes: bool(1), float64(3), int64(1), object(8)
memory usage: 24.7+ KB
```

```
print(df.head(10))
```

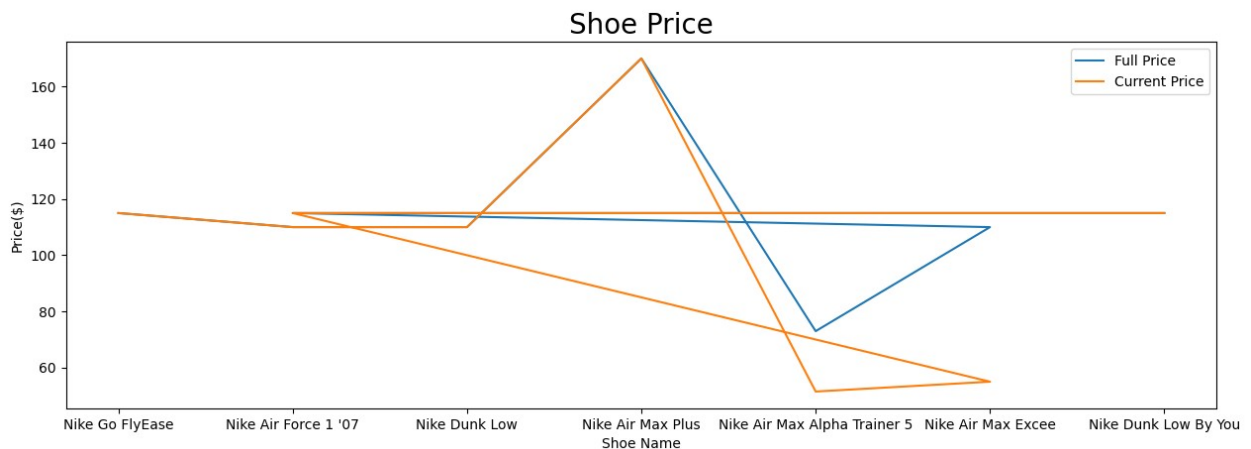
	Unnamed: 0	title	sub_title \
0	0	Nike Go FlyEase	Easy On/Off Shoes
1	1	Nike Air Force 1 '07	Men's Shoes
2	2	Nike Air Force 1 '07	Men's Shoes
3	3	Nike Dunk Low	Men's Shoes
4	4	Nike Air Max Plus	Men's Shoes
5	5	Nike Air Max Alpha Trainer 5	Men's Training Shoes
6	6	Nike Air Max Excee	Men's Shoe
7	7	Nike Air Force 1 '07	Men's shoes

8	8	Nike Dunk Low By You	Custom Men's Shoes
9	9	Nike Dunk Low By You	Custom Men's Shoes
brand			color_breif
\			
0	Nike Sportswear	Black/White	
1	Nike Sportswear	White/White	
2	Nike Sportswear	White/White/Wolf Grey	
3	Nike Sportswear	Midnight Navy/Summit	White/White/Light Smoke Grey
4	Nike Sportswear	Black/Black/Black	
5	Nike	Black/Black/White	
6	Nike Sportswear	Black/Dark Grey/Black	
7	Nike Sportswear	White/Picante Red/Wolf Grey	
8	Nike Sportswear	Multi-Colour/Multi-Colour	
9	Nike Sportswear	Multi-Colour/Multi-Colour	
fullPrice	currentPrice	country	availability
publish_date	\		
0	114.95	114.95	GB True 2022-09-13T08:44:00.000Z
1	109.95	109.95	GB True 2020-07-20T22:00:00.000Z
2	109.95	109.95	GB True 2022-12-16T08:00:00.000Z
3	109.95	109.95	GB True 2023-01-30T15:26:00.000Z
4	169.95	169.95	GB True 2012-06-30T22:00:00.000Z
5	72.95	51.47	GB True 2022-06-29T12:48:00.000Z
6	109.95	54.97	GB True 2022-11-01T08:00:00.000Z
7	114.95	114.95	GB True 2023-01-03T08:00:00.000Z
8	114.95	114.95	GB True 2022-03-01T17:00:00.000Z
9	114.95	114.95	GB True 2022-03-01T17:00:00.000Z
created_date	discount_amount	asof_date	

0	2023-04-13T15:10:39.580Z	0.00	2023-04-16
1	2023-04-06T05:41:20.507Z	0.00	2023-04-16
2	2023-04-15T07:31:47.750Z	0.00	2023-04-16
3	2023-04-06T06:56:18.880Z	0.00	2023-04-16
4	2023-04-11T21:39:18.716Z	0.00	2023-04-16
5	2023-04-15T08:05:45.473Z	21.48	2023-04-16
6	2023-04-13T18:24:45.316Z	54.98	2023-04-16
7	2023-04-13T11:22:19.952Z	0.00	2023-04-16
8	2023-04-12T22:38:03.863Z	0.00	2023-04-16
9	2023-04-12T22:38:04.012Z	0.00	2023-04-16

Line Graph

```
df2 = df.head(10)
plt.figure().set_figwidth(15)
plt.plot(df2["title"], df2["fullPrice"], label="Full Price")
plt.plot(df2["title"], df2["currentPrice"], label = "Current Price")
plt.xlabel('Shoe Name')
plt.ylabel('Price($)' )
plt.title('Shoe Price', fontsize = 20)
plt.legend()
plt.show()
```

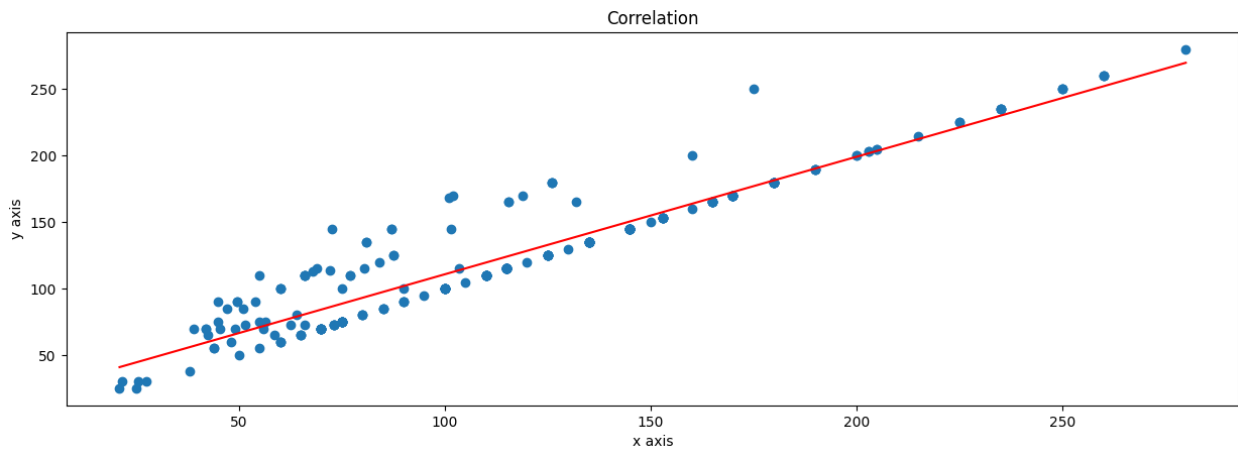


Observation: Here, in the above line graph, we can observe the difference between the Full Price and the Current Price. So, we can observe that the difference between the Full price and the Current price is maximum for "Nike Air Max Alpha Trainer 5", which shows that the price has gone down by \$20 for that shoe. However we can observe that the Current Price and the Full Price are same for "Nike Go FlyEase", "Nike Air Force 1 '07", "NiKe Dunk Low" and "Nike Air Max Plus". Therefore, we can observe that the their prices have not chnaged.

Scatter Plot and Correlation between Current Price and Full Price

```
plt.figure().set_figwidth(15)
plt.title('Correlation')
x = df["currentPrice"]
y = df["fullPrice"]
```

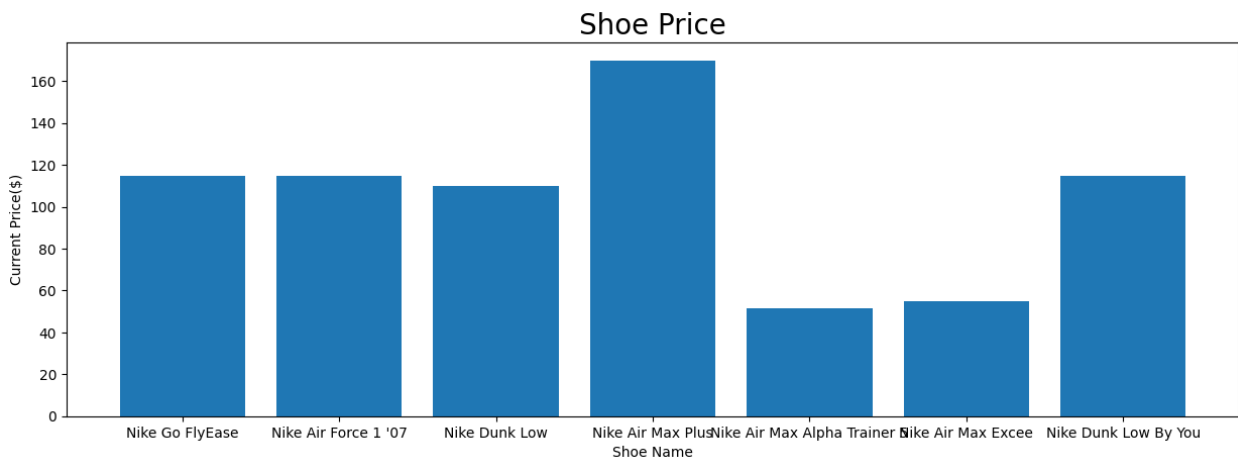
```
plt.scatter(x,y)
plt.plot(np.unique(x),
         np.poly1d(np.polyfit(x, y, 1))
         (np.unique(x)), color='red')
plt.xlabel('x axis')
plt.ylabel('y axis')
Text(0, 0.5, 'y axis')
```



Observation: This the above correlation Graph which is a combination of scatter plot and a line graph, we can observe that the corelation between current price and full price, is positive.

Bar Graph

```
plt.figure().set_figwidth(15)
plt.bar(df2["title"], df2["currentPrice"])
plt.xlabel('Shoe Name')
plt.ylabel('Current Price($)'')
plt.title('Shoe Price', fontsize = 20)
plt.show()
```



Observation: The above bar graph shows the current price of the first 7 shoes in the dataset. We can observe that the most expensive shoe in above list is "Nike Air Max Plus" which is currently selling at \$165. The cheapest shoe in above list is "Nike Air Max Alpha Trainer" which is currently selling at \$42. We can also get to know the price of any shoe in he list.