**Q1.**

#Using tkinter add filter ,lambda and other advance concept to display once click on the button

import tkinter as tk

def filterEvenNumbers():

numbers = [1,2,3,4,5,6,7,8,9,10]

evenNumbers = list(filter(lambda x: x % 2 == 0, numbers))

evenResult.config(text = f"Even Numbers: {evenNumbers}")

#print("Even Number", evenNumbers)

def filterOddNumbers():

numbers = [1,2,3,4,5,6,7,8,9,10]

oddNumbers = list(filter(lambda x: x % 2 != 0, numbers))

oddResult.config(text = f"Odd Numbers: {oddNumbers}")

root = tk.Tk()

root.title("Filter Numbers using lambda")

filterEvenButton = tk.Button(root, text = "Filter Even Button", command = filterEvenNumbers)

filterEvenButton.grid(row = 0, column = 0, padx = 10, pady = 10)

filterOddButton = tk.Button(root, text = "Filter Odd Button", command = filterOddNumbers)

filterOddButton.grid(row = 0, column = 1, padx = 10, pady = 10)

evenResult = tk.Label(root, text = "Even Numbers: ")

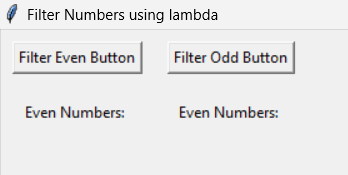
evenResult.grid(row = 1, column = 0, padx = 10, pady = 10)

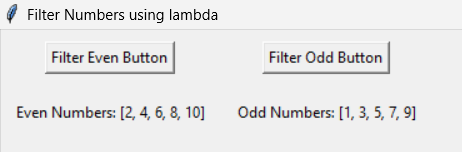
oddResult = tk.Label(root, text = "Even Numbers: ")

oddResult.grid(row = 1, column = 1, padx = 10, pady = 10)

root.mainloop()

**Q1\_Ouput**





**Q2.**

#create a school registeration form

import tkinter as tk

from tkinter import messagebox

def Form():

name = nameEntry.get()

age = ageEntry.get()

sclass = classEntry.get()

contact = contactEntry.get()

if not name or not age or not sclass or not contact:

messagebox.showwarning("Input Error", "All fields must be filled out")

return

print(f"Name: {name}")

print(f"Age: {age}")

print(f"Class: {sclass}")

print(f"Contact: {contact}")

# Clear the entry fields

nameEntry.delete(0, tk.END)

ageEntry.delete(0, tk.END)

classEntry.delete(0, tk.END)

contactEntry.delete(0, tk.END)

# Optionally show a confirmation message

messagebox.showinfo("Registration Details", "Form submitted successfully!")

# Create the main window

root = tk.Tk()

root.title("School Registration Form")

# Create and place labels and entry fields

tk.Label(root, text="Name:").grid(row=0, column=0, padx=10, pady=5, sticky="e")

tk.Label(root, text="Age:").grid(row=1, column=0, padx=10, pady=5, sticky="e")

tk.Label(root, text="Class:").grid(row=2, column=0, padx=10, pady=5, sticky="e")

tk.Label(root, text="Contact:").grid(row=3, column=0, padx=10, pady=5, sticky="e")

nameEntry = tk.Entry(root)

nameEntry.grid(row=0, column=1, padx=10, pady=5)

ageEntry = tk.Entry(root)

ageEntry.grid(row=1, column=1, padx=10, pady=5)

classEntry = tk.Entry(root)

classEntry.grid(row=2, column=1, padx=10, pady=5)

contactEntry = tk.Entry(root)

contactEntry.grid(row=3, column=1, padx=10, pady=5)

# Create and place the submit button

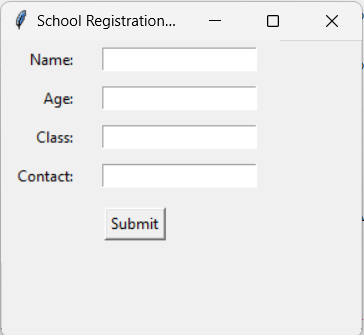
submitButton = tk.Button(root, text="Submit", command=Form)

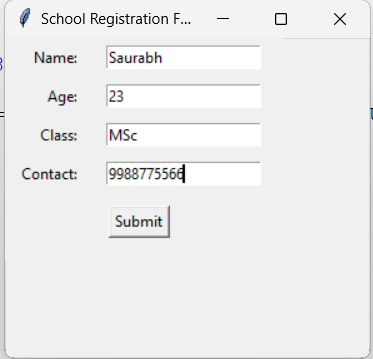
submitButton.grid(row=4, column=0, columnspan=2, pady=10)

# Run the application

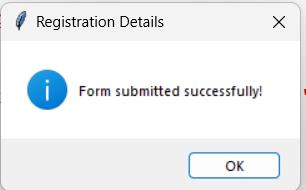
root.mainloop()

Q2\_Output





Message box will pop out after hitting submit



And data will be shown on the terminal

