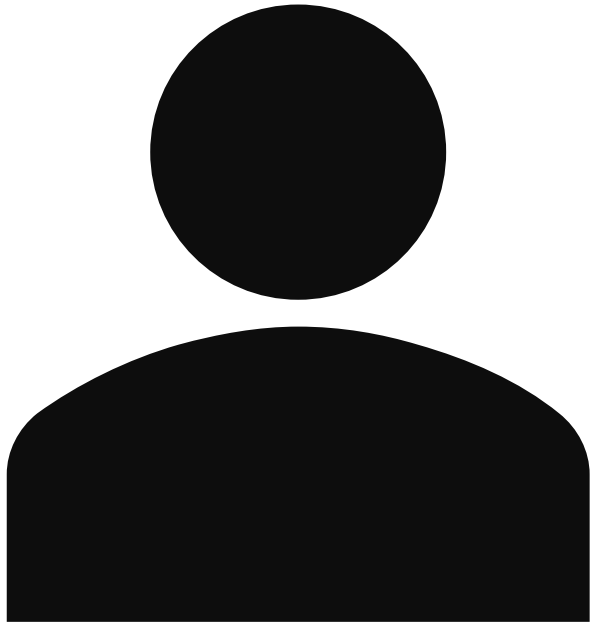
Numerical Analysis

Dr :Manal Elsaid

Lecture 3 : Special Functions (Gamma)



Team name : DataVerse

Made by :
Zeinab Talaat Antar Samaha

- Project Link : https://drive.google.com/file/d/1ICpAsUVGgpRegZKJ4dpfllTwEF5soGPl/view?usp=drivesdk

OutPut Code

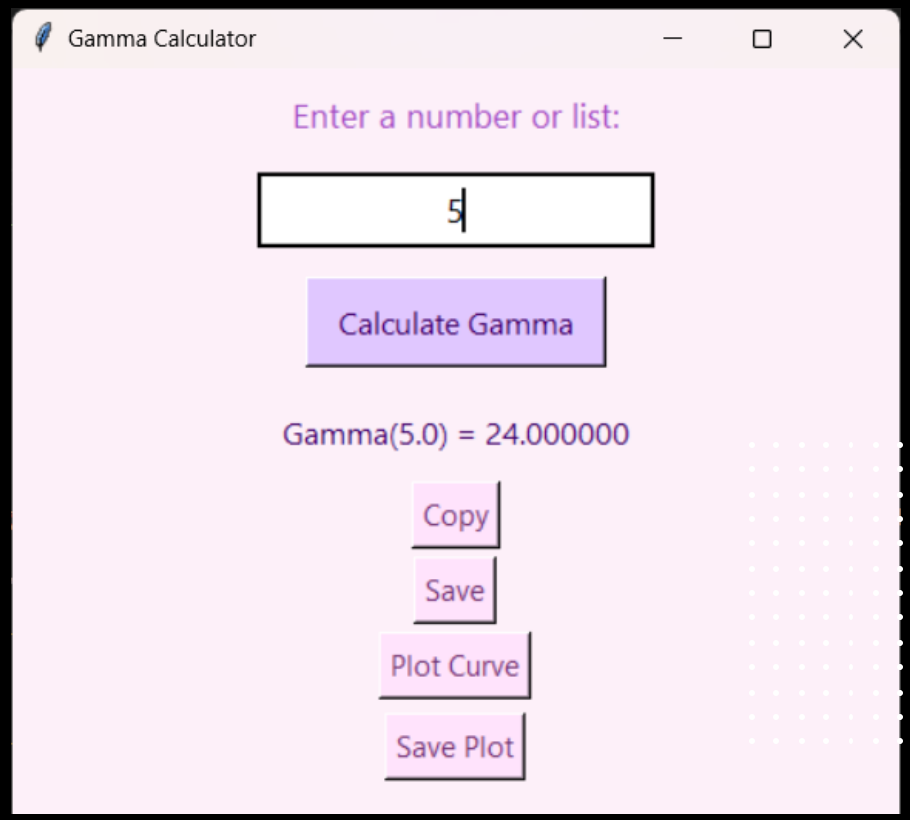
شاشة اختيار اللغة

```python
def language_selector():
    root = tk.Tk()
    root.title("Language / اللغة")
    root.geometry("300x200")
    root.configure(bg="#fdf0f9")

    tk.Label(root, text="اختاري اللغة / Choose Language", font=("Segoe UI", 13, "bold"), bg="#fdf0f9", fg="#a24ac3").pack(

    tk.Button(root, text="العربية", font=("Segoe UI", 12), bg="#e0c7ff", fg="#3d0066",
              command=lambda: [root.destroy(), arabic_frame()]).pack(pady=5, ipadx=10, ipady=5)

    tk.Button(root, text="English", font=("Segoe UI", 12), bg="#e0c7ff", fg="#3d0066",
              command=lambda: [root.destroy(), english_frame()]).pack(pady=5, ipadx=10, ipady=5)

    root.mainloop()

language_selector()
```

شاشة اللغة العربية

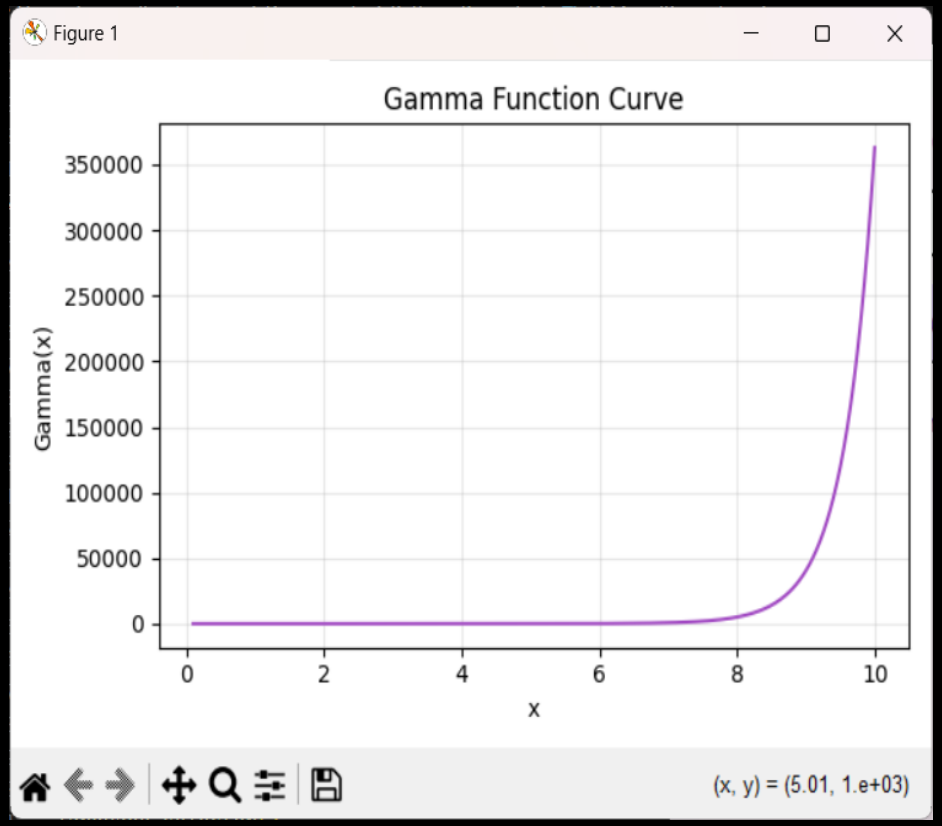
```python
def arabic_frame():
    def calculate_gamma():
        try:
            numbers = entry.get().split(",")
            results = []
            for num in numbers:
                x = float(num.strip())
                if x <= 0:
                    results.append(f"غير صحيح: {x}")
                else:
                    results.append(f"Gamma({x}) = {gamma(x):.6f}")
            result_text = "\n".join(results)
            result_label.config(text=result_text)
            global last_result
            last_result = result_text
        except ValueError:
            messagebox.showerror("خطأ", "أدخل رقمًا أو قائمة أرقام مفصولة بفواصل (مثال: 2.5, 4)")

    def copy_result():
        if last_result:
            pyperclip.copy(last_result)
            messagebox.showinfo("تم النسخ", "تم نسخ النتيجة.")
        else:
            messagebox.showwarning("تحذير", "لا توجد نتيجة لنسخها.")

    def save_result():
        if last_result:
            file_path = filedialog.asksaveasfilename(defaultextension=".txt",
                                                     filetypes=[("Text Files", "*.txt")])
            if file_path:
                with open(file_path, "w", encoding="utf-8") as f:
                    f.write(last_result)
                messagebox.showinfo("تم الحفظ", f"تم حفظ النتيجة في:\n{file_path}")
        else:
            messagebox.showwarning("تحذير", "لا توجد نتيجة لحفظها.")
```

اللغة الانجليزيه شاشة




```python
def english_frame():
    def calculate_gamma():
        try:
            numbers = entry.get().split(",")
            results = []
            for num in numbers:
                x = float(num.strip())
                if x <= 0:
                    results.append(f"Invalid: {x}")
                else:
                    results.append(f"Gamma({x}) = {gamma(x):.6f}")
            result_text = "\n".join(results)
            result_label.config(text=result_text)
            global last_result
            last_result = result_text
        except ValueError:
            messagebox.showerror("Error", "Enter a valid number or comma-separated list (e.g. 2.5, 4)")

    def copy_result():
        if last_result:
            pyperclip.copy(last_result)
            messagebox.showinfo("Copied", "Result copied to clipboard.")
        else:
            messagebox.showwarning("Warning", "No result to copy.")

    def save_result():
        if last_result:
            file_path = filedialog.asksaveasfilename(defaultextension=".txt",
                                                     filetypes=[("Text Files", "*.txt")])
            if file_path:
                with open(file_path, "w", encoding="utf-8") as f:
                    f.write(last_result)
                messagebox.showinfo("Saved", f"Result saved to:\n{file_path}")
        else:
            messagebox.showwarning("Warning", "No result to save.")

```

منحنی داله جاما

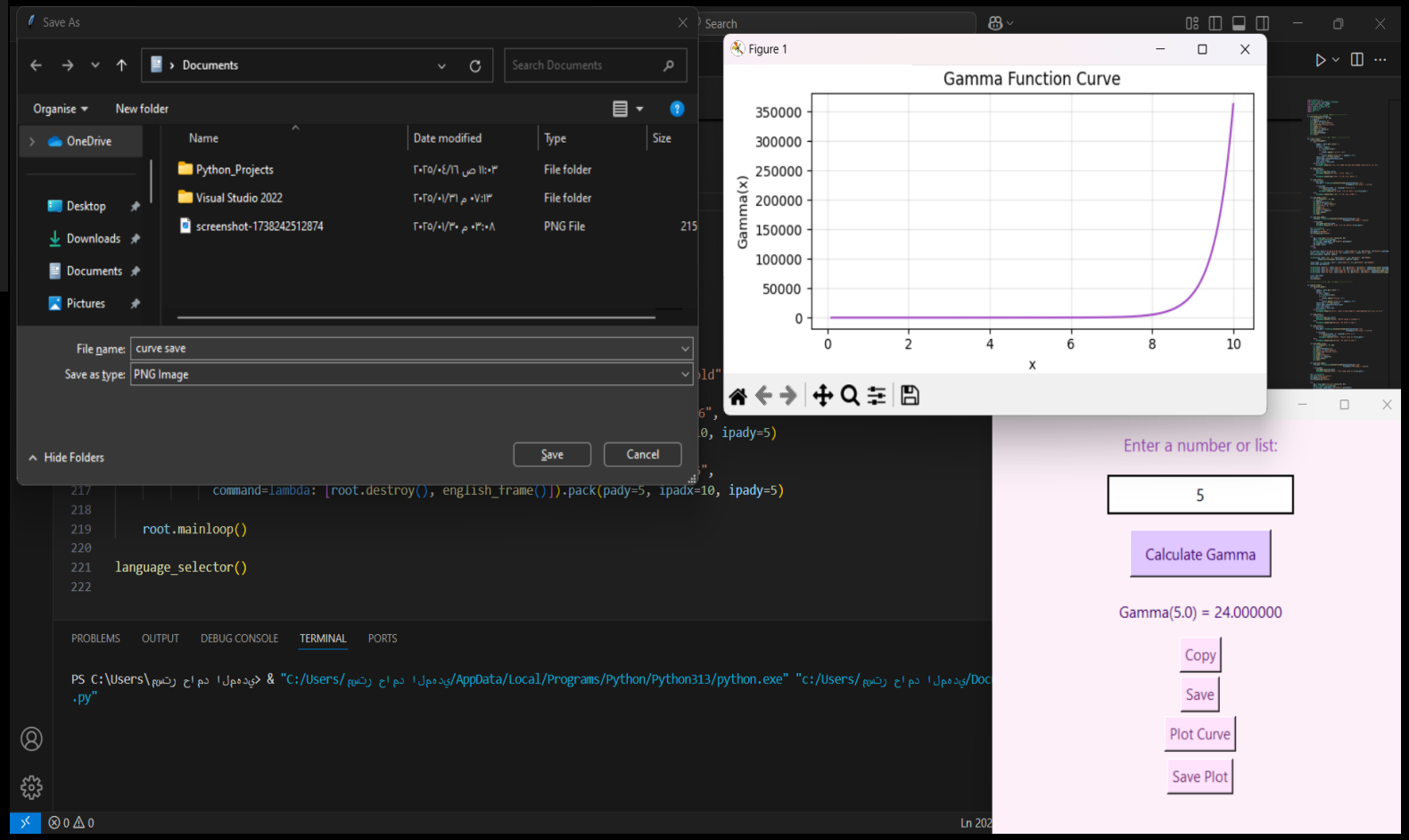



```python
    def plot_gamma_curve():
        x = np.linspace(0.1, 10, 400)
        y = gamma(x)
        plt.figure(figsize=(6, 4))
        plt.plot(x, y, color="#a24ac3")
        plt.title("Gamma Function Curve")
        plt.xlabel("x")
        plt.ylabel("Gamma(x)")
        plt.grid(True, alpha=0.3)
        plt.tight_layout()
        plt.show()

    def save_plot_image():
        file_path = filedialog.asksaveasfilename(defaultextension=".png",
                                                 filetypes=[("PNG Image", "*.png")])
        if file_path:
            save_gamma_plot(file_path)
            messagebox.showinfo("Saved", f"Plot image saved to:\n{file_path}")

    win = tk.Toplevel()
    win.title("Gamma Calculator")
    win.geometry("460x500")
    win.configure(bg="#fdf0f9")

    try:
        img = Image.open("icon.png").resize((60, 60))
        photo = ImageTk.PhotoImage(img)
        tk.Label(win, image=photo, bg="#fdf0f9").pack(pady=5)
        win.iconphoto(False, photo)
        win.image = photo
    except:
        pass

    tk.Label(win, text="Enter a number or list:", font=("Segoe UI", 13), bg="#fdf0f9", fg="#a24ac3").pack(pady=10)
    entry = tk.Entry(win, font=("Segoe UI", 13), justify="center", relief="solid", bd=2)
```

⚠ 0

حفظ منحنی داله جاما

```python
def save_gamma_plot(filename):
    x = np.linspace(0.1, 10, 400)
    y = gamma(x)
    plt.figure(figsize=(6, 4))
    plt.plot(x, y, color="#a24ac3")
    plt.title("Gamma Function Curve")
    plt.xlabel("x")
    plt.ylabel("Gamma(x)")
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.savefig(filename)
    plt.close()

```



Thank you

- Project summary
- In this project, a graphical user interface was developed using Python and the Tkinter library to calculate the Gamma function for a single number or a list of numbers, with features such as:

- Displaying results clearly.
- Copying or saving results to a file.

- Plotting the Gamma function curve
- Supporting both Arabic and English interfaces