EX.NO:	HOSPITAL PRIORITY QUEUE SYSTEM USING QUEUE
DATE: / /2025	

AIM

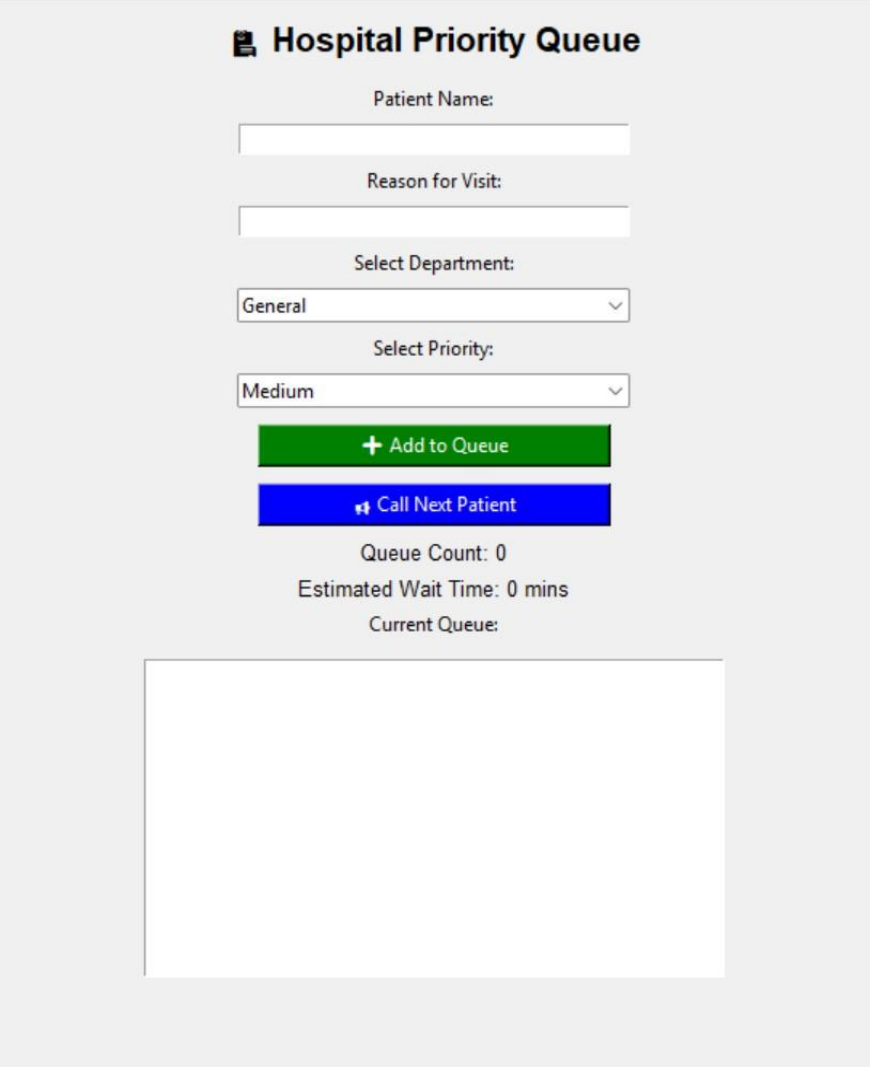
To Develop hospital priority queue system using queue in python.

SOURCE CODE:

```
import tkinter as tk
from tkinter import messagebox, tt
patient_queue = []
patient_id = 1000
priority_text = {1: "High", 2: "Medium", 3: "Low"}
departments = ["ICU", "General", "Lab"]
def add_patient():
    global patient_id
    name = entry_name.get().strip()
    reason = entry_reason.get().strip()
    department = dept_combo.get()
    priority = priority_var.get()
    if not name or not reason or department == "":
        messagebox.showwarning("Warning", "Fill all fields correctly")
        return
    patient_id += 1
    data = (priority, patient_id, name, reason, department)
    patient_queue.append(data)
    patient_queue.sort()
    entry_name.delete(0, tk.END)
    entry_reason.delete(0, tk.END)
    messagebox.showinfo(
        "Added",
        f"Patient Added:\n\nID: {patient_id}\nName: {name}\nDept: {department}\nReason:
{reason}\nPriority: {priority_text[priority]}"
    show_queue()
def call_next():
    if not patient_queue:
        messagebox.showwarning("Empty Queue", "No patients waiting")
        return
    priority, pid, name, reason, dept = patient_queue.pop(0)
    messagebox.showinfo(
        "Next Patient",
```

```
        f"Calling Patient:\n\nID: {pid}\nName: {name}\nDept: {dept}\nReason: {reason}"
    show_queue()
def show_queue():
    listbox.delete(0, tk.END)
    for priority, pid, name, reason, dept in patient_queue:
        listbox.insert(tk.END, f"ID: {pid} | {name} | {dept} | {priority_text[priority]}")
    count = len(patient_queue)
    lbl_count.config(text=f"Queue Count: {count}")
    lbl_wait.config(text=f"Estimated Wait Time: {count * 5} mins")
def update_priority(event):
    selected = priority_combo.get()
    if selected == "High":
        priority_var.set(1)
    elif selected == "Medium":
        priority_var.set(2)
    else:
        priority_var.set(3)
root.title("Hospital Enhanced Priority Queue System")
tk.Label(root, text="Hospital Priority Queue", font=("Arial", 16, "bold")).pack(pady=10)
tk.Label(root, text="Patient Name:").pack()
entry_name = tk.Entry(root, width=40)
entry_name.pack(pady=5)
tk.Label(root, text="Reason for Visit:").pack()
entry_reason = tk.Entry(root, width=40)
entry_reason.pack(pady=5)
tk.Label(root, text="Select Department:").pack()
dept_combo = ttk.Combobox(root, values=departments, state="readonly", width=37)
dept_combo.pack(pady=5)
dept_combo.set("General")
tk.Label(root, text="Select Priority:").pack()
priority_var = tk.IntVar(value=2)
priority_combo = ttk.Combobox(root, values=["High", "Medium", "Low"],
state="readonly", width=37) priority_combo.pack(pady=5)
priority_combo.set("Medium")
priority_combo.bind("<<ComboboxSelected>>", update_priority)
tk.Button(root, text="Add to Queue", width=30, bg="green", fg="white",
command=add_patient).pack(pady=5)
tk.Button(root, text="Call Next Patient", width=30, bg="blue", fg="white",
command=call_next).pack(pady=5)
lbl_count = tk.Label(root, text="Queue Count: 0", font=("Arial", 10))
lbl_wait = tk.Label(root, text="Estimated Wait Time: 0 mins", font=("Arial", 10))
lbl_wait.pack()
tk.Label(root, text="Current Queue:").pack()
listbox = tk.Listbox(root, width=60, height=12)
listbox.pack(pady=10)
root.mainloop()
```

OUTPUT:



RESULT:

The program has been Successfully Executed.