```python
import tkinter as tk
from tkinter import scrolledtext
import pyttsx3

# -------------------- Text-to-Speech Setup -------------------- #
engine = pyttsx3.init()
engine.setProperty('rate', 150)  # Adjust speaking speed

def speak(text):
    """Speak the given text aloud."""
    engine.say(text)
    engine.runAndWait()

# -------------------- Emergency Responder -------------------- #
def emergency_response(command):
    """Respond to emergency commands."""
    command = command.lower()
    if "accident" in command:
        speak("Accident detected. Stay calm. Check breathing and bleeding.")
        output_box.insert(tk.END, "рџљЁ Accident detected. Instructions spoken.\n")
    elif "fire" in command:
        speak("Fire emergency. Move away from fire and call emergency services.")
        output_box.insert(tk.END, "рџ”Ґ Fire emergency. Instructions spoken.\n")
    elif "medical" in command:
        speak("Medical emergency. Sit or lie down. Help is being alerted.")
        output_box.insert(tk.END, "рџЏҐ Medical emergency. Instructions spoken.\n")
    else:
        speak("Emergency not recognized. Please repeat.")
        output_box.insert(tk.END, "вќ“ Emergency not recognized.\n")
    output_box.see(tk.END)

def handle_emergency():
    """Fetch user input and respond to emergency."""
    command = emergency_input.get()
    emergency_input.delete(0, tk.END)
    emergency_response(command)

# -------------------- Offline Coding Tutor -------------------- #
ERROR_EXPLANATIONS = {
    "SyntaxError": "Check for missing colons, parentheses, or indentation.",
    "IndentationError": "Blocks are improperly indented.",
    "NameError": "Variable or function not defined.",
    "TypeError": "Incompatible types used.",
    "ZeroDivisionError": "Cannot divide by zero.",
    "AttributeError": "Non-existent attribute or method used."
}

def explain_error(error_type):
    """Return explanation for common Python errors."""
    return ERROR_EXPLANATIONS.get(error_type, "An error occurred. Check your code carefully.")

def run_code():
    """Execute Python code from the input box and show output/errors."""
    code = code_input.get("1.0", tk.END)
    output_box.config(state='normal')
    try:
        exec(code, {})
        output_box.insert(tk.END, "вњ… Code executed successfully!\n")
    except Exception as e:
        error_type = type(e).__name__
        explanation = explain_error(error_type)
        output_box.insert(tk.END, f"вќЊ {error_type}: {str(e)}\nрџ’Ў Explanation: {explanation}\n")
    output_box.config(state='disabled')
    output_box.see(tk.END)

# -------------------- GUI Setup -------------------- #
root = tk.Tk()
root.title("рџ’Ў Offline AI Assistant")
root.geometry("800x700")
root.resizable(False, False)
root.configure(bg="#f5f5f5")

# Title
title = tk.Label(root, text="рџ’Ў Offline AI Assistant", font=("Helvetica", 20, "bold"), bg="#f5f5f5")
title.pack(pady=10)

# --- Emergency Section --- #
emergency_frame = tk.LabelFrame(root, text="рџљЁ Emergency Responder", font=("Helvetica", 14, "bold"),
                                padx=10, pady=10, bg="#e3f2fd")
emergency_frame.pack(fill="x", padx=20, pady=10)

emergency_input = tk.Entry(emergency_frame, font=("Helvetica", 12), width=50)
emergency_input.pack(side="left", padx=(0,10))

emergency_button = tk.Button(emergency_frame, text="Trigger Emergency", command=handle_emergency,
                             bg="#ff5252", fg="white", font=("Helvetica", 12, "bold"))
emergency_button.pack(side="left")

# --- Coding Tutor Section --- #
coding_frame = tk.LabelFrame(root, text="рџ’» Offline Coding Tutor", font=("Helvetica", 14, "bold"),
                             padx=10, pady=10, bg="#fff3e0")
coding_frame.pack(fill="both", expand=True, padx=20, pady=10)

code_input = scrolledtext.ScrolledText(coding_frame, height=15, width=80, font=("Consolas", 12))
code_input.pack(pady=5)

run_button = tk.Button(coding_frame, text="Run Code", command=run_code,
                       bg="#4CAF50", fg="white", font=("Helvetica", 12, "bold"))
run_button.pack(pady=5)

# Output Box
output_label = tk.Label(coding_frame, text="рџ“ќ Output / Error Explanation:", font=("Helvetica", 12, "bold"),
                        bg="#fff3e0")
output_label.pack(pady=(10,0))

output_box = scrolledtext.ScrolledText(coding_frame, height=10, width=80, font=("Consolas", 12))
output_box.pack(pady=5)
output_box.config(state='disabled')  # Read-only

# -------------------- Example Inputs -------------------- #
# You can copy these into the code_input box to test:
# Example 1: Hello World
# print("Hello, Offline AI Assistant!")

# Example 2: Sum of two numbers
# a = 10
# b = 20
# print("Sum:", a + b)

# Example 3: Loop Example
# for i in range(5):
#     print("Counting:", i)
```

```
# Example 4: Function with error
# def divide(a, b):
#     return a / b
# print(divide(10, 0))  # Will trigger ZeroDivisionError

# Example 5: Undefined variable
# print(x)  # Will trigger NameError

# Run the GUI
root.mainloop()
```