LERATO
MOLOKOMME
ST10439626

PROG6221

PART 1

PROJECT OVERVIEW

- *Its about cybersecurity awareness and support by providing real-time assistance, and information about passwords , phinishing, and safe browsing guidance through an intelligent chatbot, helping users navigate security challenges efficiently."*

CODE STRUCTURE

- //start the program

  start

-

- Asking for user name //prompting user name
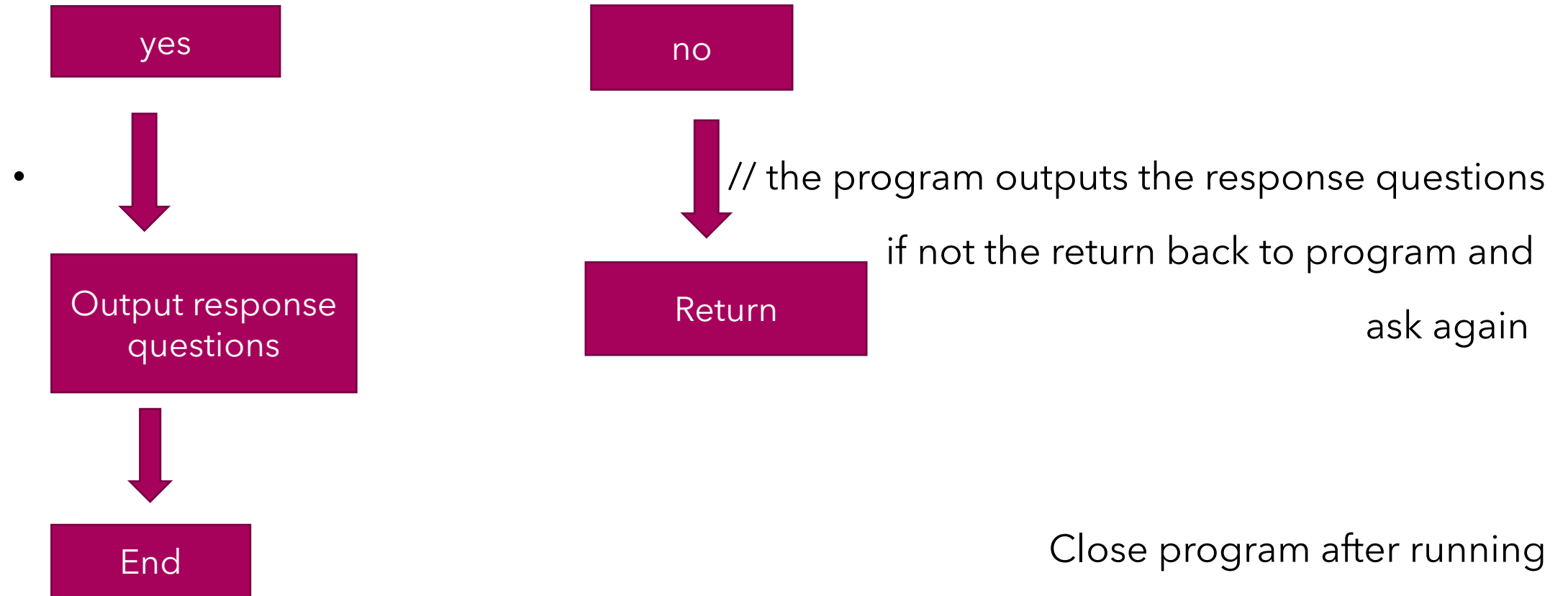
  // a basic response system (Boolean)

-

  Did the user input question

-

-

CODE STRUCTURE CONTINUED

yes

no

// the program outputs the response questions

•

Output response questions

Return

if not the return back to program and

ask again

End

Close program after running

CODE WALKTHROUGH

- **Program Start**:
- The program begins by prompting the user to input their name.
- Once the user provides their name,
the program (represented by the chatbox,)
greets the user personally by using their name.
For example: "Hello, [User's Name]! How can I assist you today?"

CODE WALKTHROUGH

- **Response System**:

- The program is designed with an automatic response.

- If the user did not ask question , the chatbox will send a message, such as: "Im still learning ask me anything about password , phinishing and safe browsing " .

- If the user inputs a question, the program processes the input and replies appropriately based on predefined responses.

CODE WALKTHROUGH

- The chatbox is programmed with a few basic responses to common questions
- or statements that the user might make.
- For example:
- **User Input:** "how are you?"
- **chatbox Response:** "im great."
- **User Input:** "hello"
- **Susan's Response:** "hello + [username] "

CODE FORMANTTING

- **Programming Language and Framework**:

- The program is developed using **C#** and runs on the **.NET Core** framework. This provides a cross-platform environment for running the application on different operating systems, such as Windows, Linux, and macOS.

- **Code Formatting According to C# Standards**:

- The code strictly adheres to **C# best practices** and is written following the official C# style guide to ensure readability, maintainability, and consistency.

- This includes:
    - Proper use of **indentation** (usually 4 spaces per indentation level).
    - Clear **variable naming conventions** (camelCase for variables and method parameters, PascalCase for methods, classes, and properties).

CODE FORMANTING

- **Visual Enhancement with Cyan Color**:

- The code is designed to display certain outputs or logs in the color **cyan** to improve readability and help distinguish important information from other outputs.

- **Code Readability and Maintenance**:

- The code will be clean and modular, with clear separation of concerns. This could include organizing code into **methods** for individual tasks (e.g., a method to handle user input, another to display a greeting, and another for handling exits).