TASK 1:

Create a Subsystem:

A *subsystem* is a set of blocks that are replaced by a single block called a Subsystem block.

You can create a Subsystem by using any of the following methods:
1. Add a Subsystem block to your model, then open that block and add the blocks that it contains to the subsystem window.
2. Add the blocks required, and then group them into a subsystem.

Create Subsystem from a Block:

Add a Subsystem to the model, and then add the blocks that make up the subsystem.
1. Copy the Subsystem block from the **Ports & Subsystems** library into your model.
2. Double-click on the block to open it.
3. Create the Subsystem in the empty subsystem window. Use Inport blocks to represent input from outside the subsystem and Outport blocks to represent external output.

For example, this subsystem includes a Sum block and Inport and Outport blocks to represent input to and output from the subsystem:
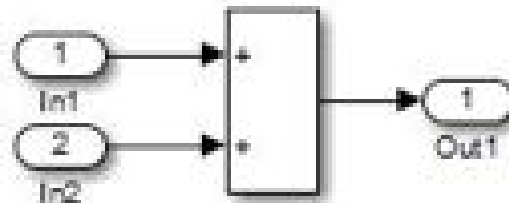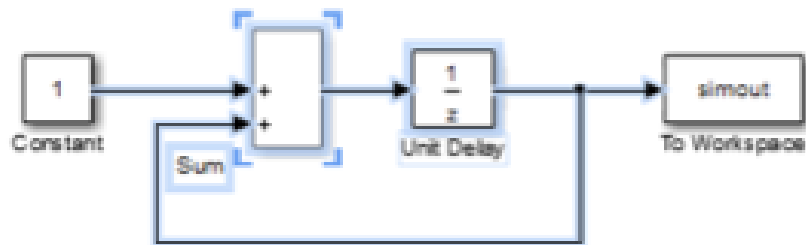

Create Subsystem from Selection:

If your model already contains the blocks that you want to convert to a subsystem, then create the subsystem by grouping those blocks.

1. Select individual blocks that you want to include in a subsystem. To select multiple blocks, drag a bounding box with your cursor over the blocks and connecting lines that you want to enclose in the subsystem.

    For example, this figure shows a model that represents a counter. The bounding box selects the Sum and Unit Delay blocks:

2. Select **Diagram** > **Subsystems & Model Reference** > **Create Subsystem from Selection**. A Subsystem block replaces the selected blocks.

3. Resize the Subsystem block so that the port labels are readable. If you open the Subsystem block, the underlying system opens.
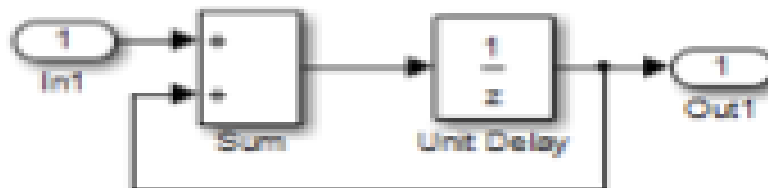
Simulink adds Inport and Outport blocks to represent input and output ports of the subsystem.

Conditional Subsystems Overview:

A *conditional subsystem*, also known as a *conditionally executed subsystem*, is a subsystem where its execution depends on the value of an input signal, called the *control signal*. The control signal enters a subsystem block at the *control input*. Conditional subsystems can be very useful when you are building complex models that contain components whose execution depends on other components.

Simulink® supports the following types of conditional subsystems:

- **Enabled Subsystem:**
  Executes while the control signal is positive. Execution starts at the time step where the control signal crosses zero (from the negative to the positive direction).

- **Triggered Subsystem:**
  Executes each time a *trigger event* occurs.
  Trigger Event:
  - Can occur on the rising or falling edge of a trigger signal, which can be continuous or discrete.

- **Triggered and Enabled Subsystem:**
  Executes once at the time step when the trigger event occurs and the enable control signal has a positive value.

- **Control Flow Subsystem:**
  Executes one or more times at the current time step when enabled by a *control flow block*.
  Control Flow Block:
  - Implements control logic similar to that expressed by control flow statements of programming languages (e.g., if-then, while, do, and for).

Conditional Execution Behavior:

To speed up the simulation of a model, by default the Simulink® software avoids unnecessary execution of conditionally executed blocks and blocks connected to Switch and Multiport Switch. This behavior is **conditional execution (CE)** behavior. You can either disable this behavior for all Switch and Multiport Switch blocks in a model, or for specific conditional subsystems.

TASK 2:

Questions:
1. How do the Subsystems procedures help create different types of subsystems (Eg: Conditional Subsystems)? Are they all available under the Ports and Subsystems library? Is there any information, like Library information, that can help the user determine where to find the right block?
2. Why would the user require a subsystem? This is required for the introduction to the Subsystem procedure.

3. What are ports and how do you use them?
4. Where are the examples to explain the conditional subsystems?
5. What are Switch and Multiport Switch?
6. Where are the examples for CE Behavior for the user to properly understand what it is?