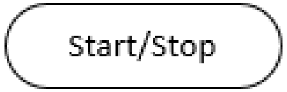
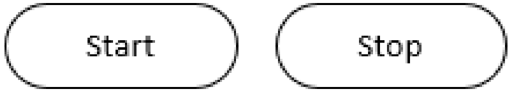
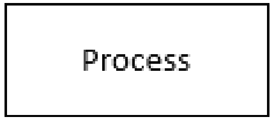
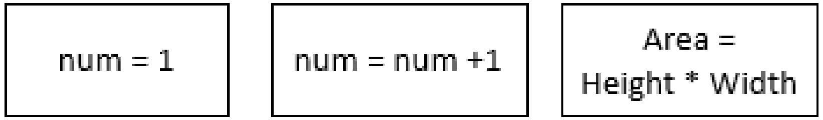
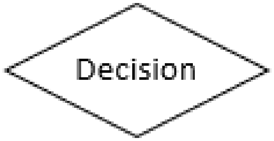
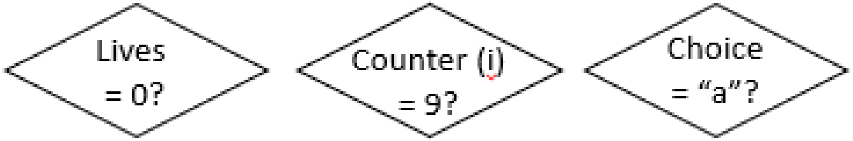
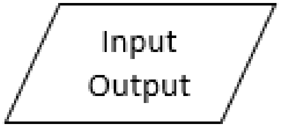
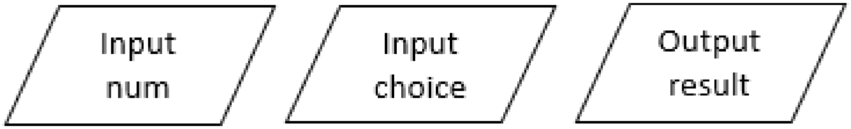
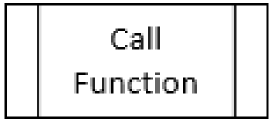
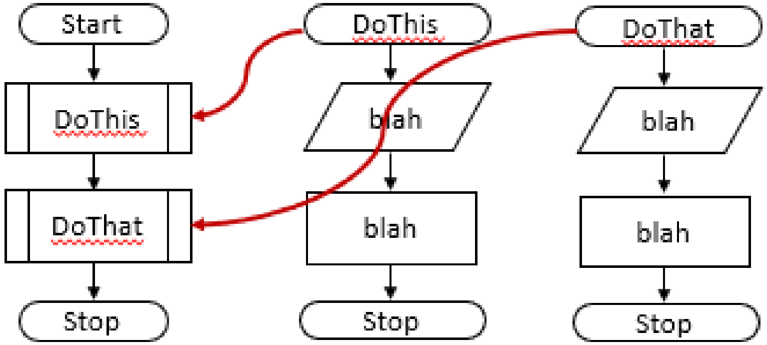
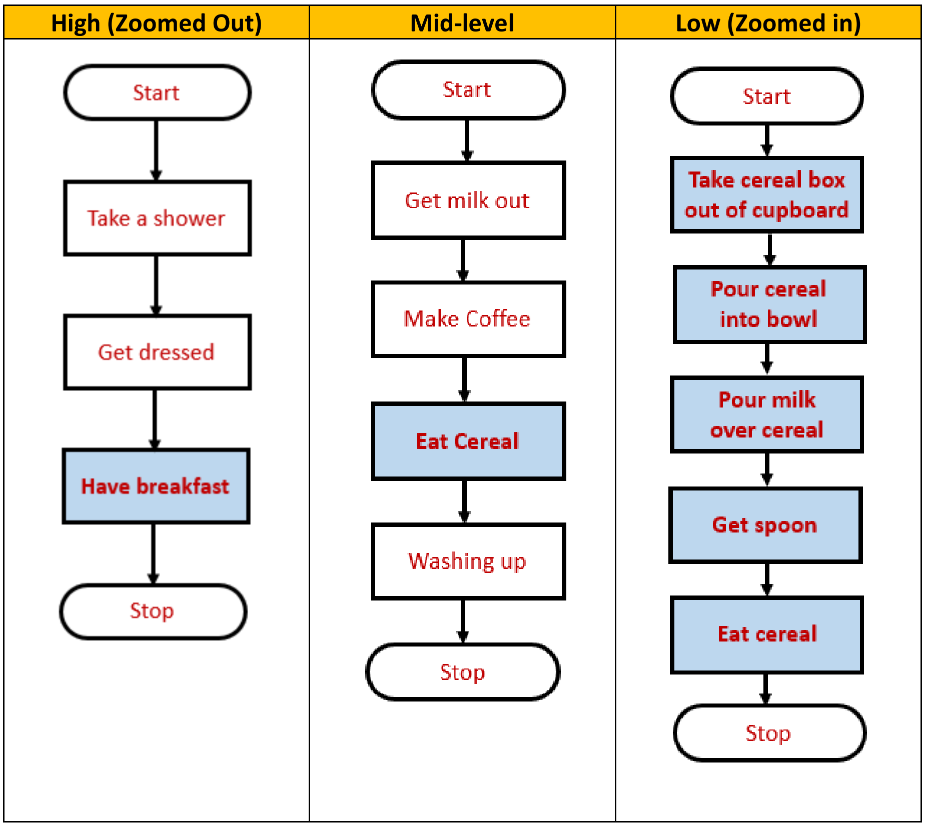
Flowchart Shapes

Shape	What's it for?	Notes	Examples
	- Begin a flowchart - End a flowchart	A flowchart can have: Only one 'Start', Multiple 'Stops'	
	- Setting the value of variables - Changing the value of variables - Performing calculations		
	- To make decisions/selections - To check the value of variables - IF & ELIF statements - Controlling loops (for & while)	Must contain a question with a 'yes' or 'no' answer	
	- User typing in information - To print information to the user		
	- To separate your program into logically organised sections e.g: - def MainMenu() - def Level1() - def Level2() - To call a function into action - To jump from one function into another	Multiple flowcharts required (*see examples*)	 Start DoThis DoThat Stop DoThis blah blah Stop DoThat blah blah Stop

Different levels of Flowchart

We can create flowcharts on many different levels of complexity…

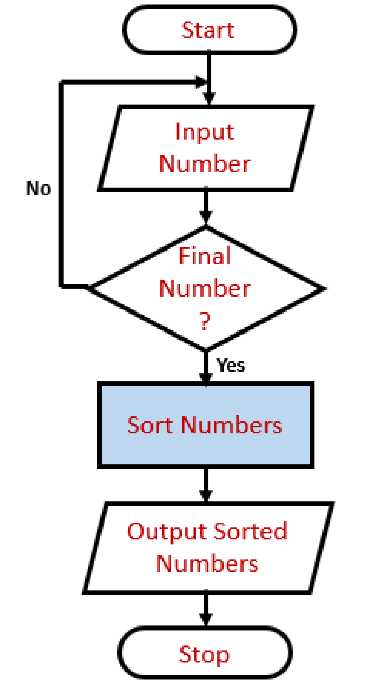


High ↑	**Python, Java Script, PHP**
	↑ less detail ↑ less depth more detail ↓ more depth ↓
↓	**Assembly Code**
Low	**Binary**

Notice how the higher-level flowcharts hide unnecessary detail. This is called 'Abstraction'.

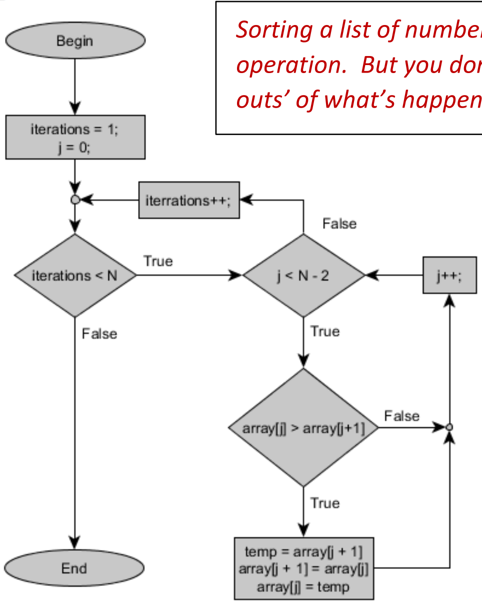
Abstraction - applied to sorting algorithms

High (Zoomed Out)	Low (Zoomed in)



```
MyList = [4, 26, 7, 22, 5, 2, 9]
MyList.Sort()
```

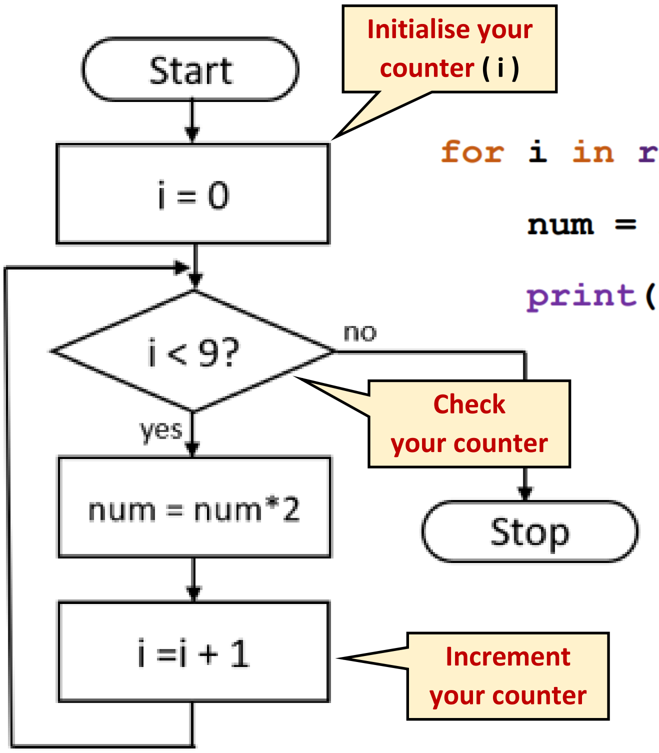
Sorting a list of numbers is actually quite a complex operation. But you don't need to know the 'ins and outs' of what's happening behind the scenes!



ⓘ *You need to demonstrate no more than* ***5*** *complex problems* ⓘ

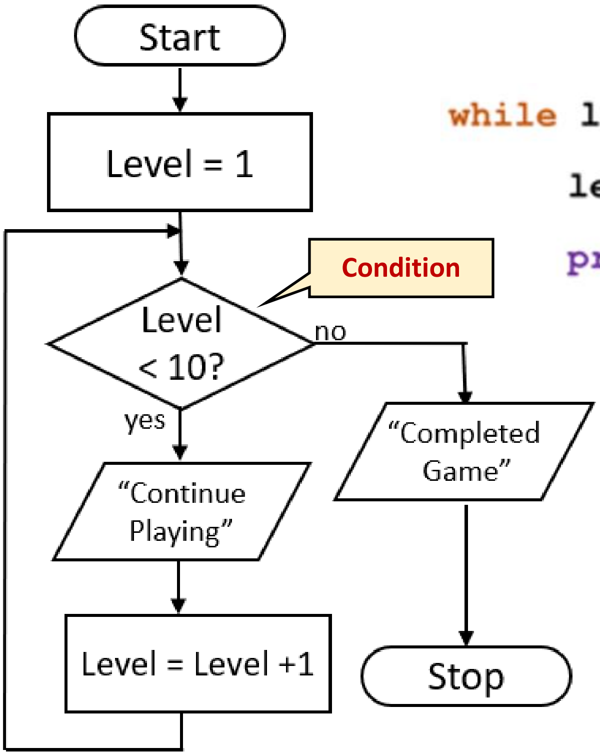
Loops

FOR Loops…



```
for i in range(0, 9):
    num = num * 2
    print("Multiplied by two is " + num)
```

WHILE Loops…

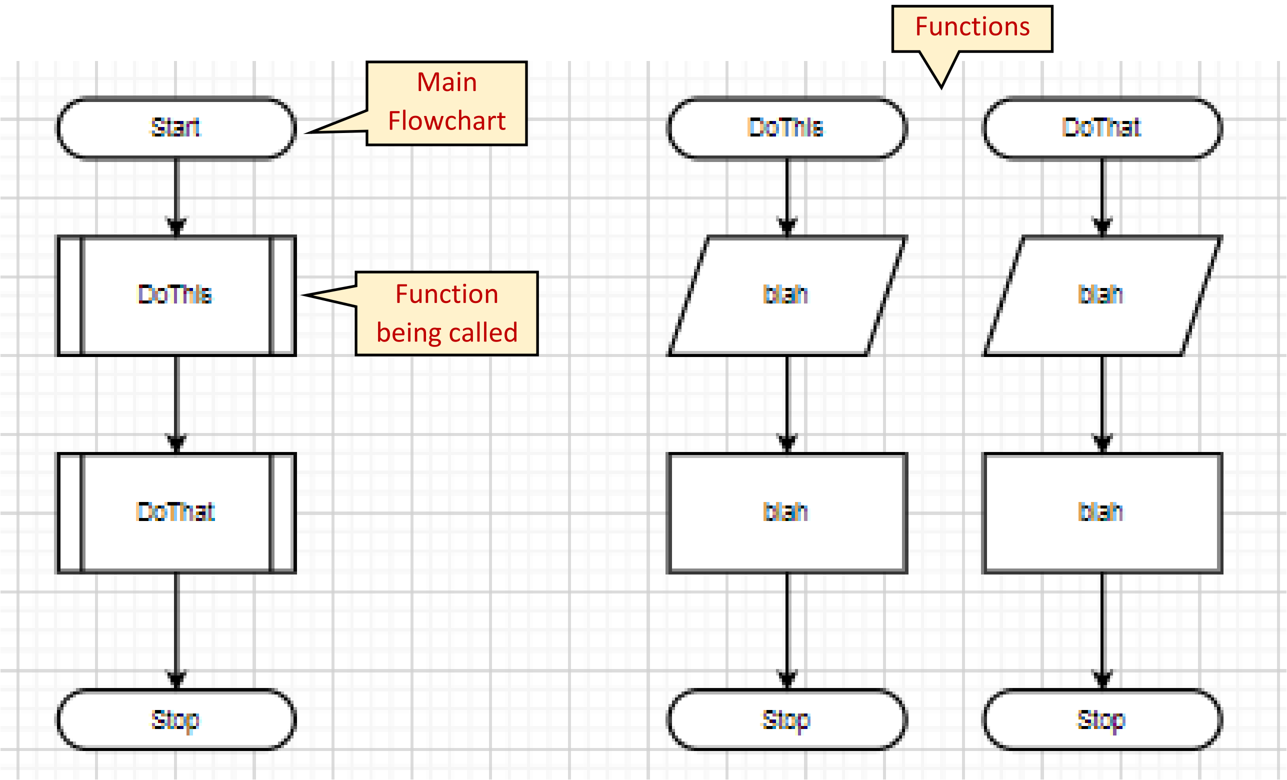


```
while level < 10:
    level = level + 1
    print("Completed Level " + level)
```

ⓘ *Loops are used to read lines in text files* ⓘ

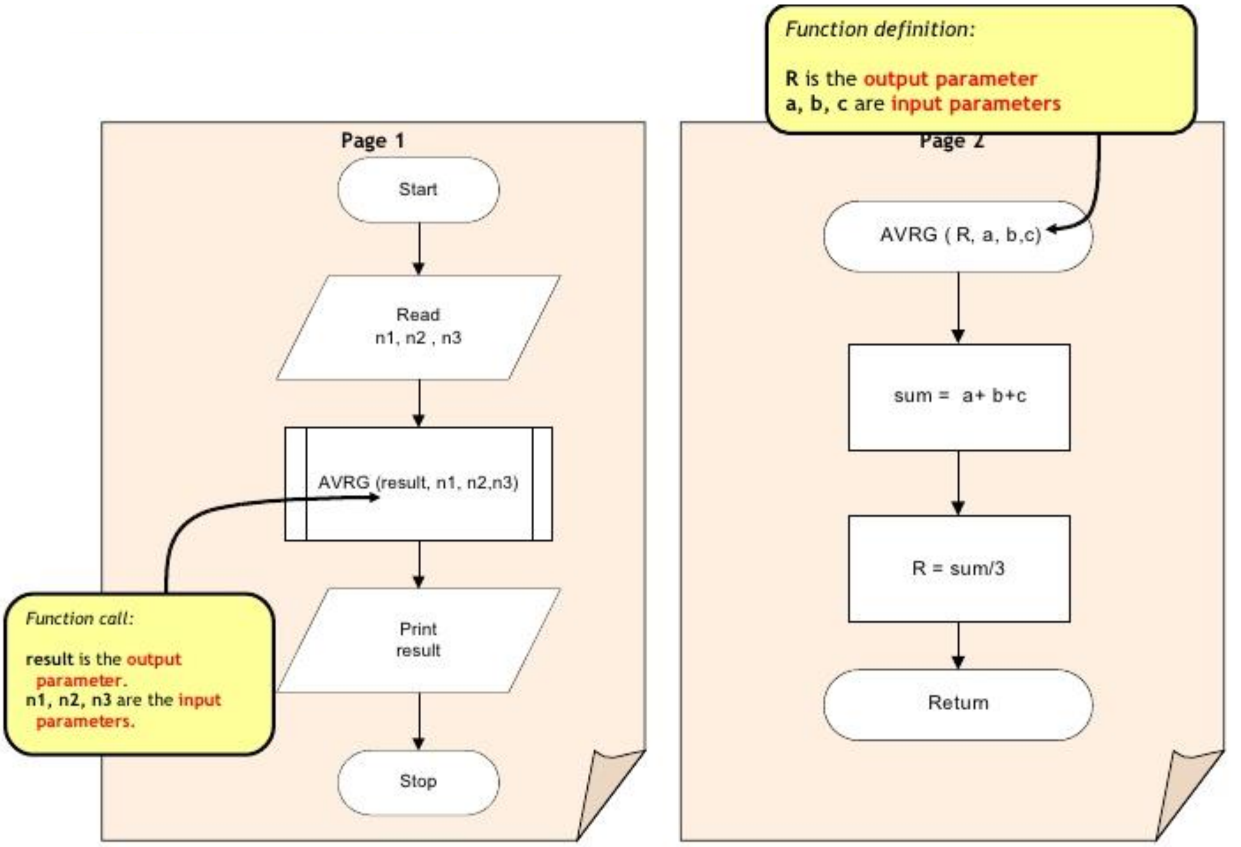
Functions

Functions are represented as separate flowcharts…



ⓘ *After a function has finished, the program jumps back to the main flowchart* ⓘ

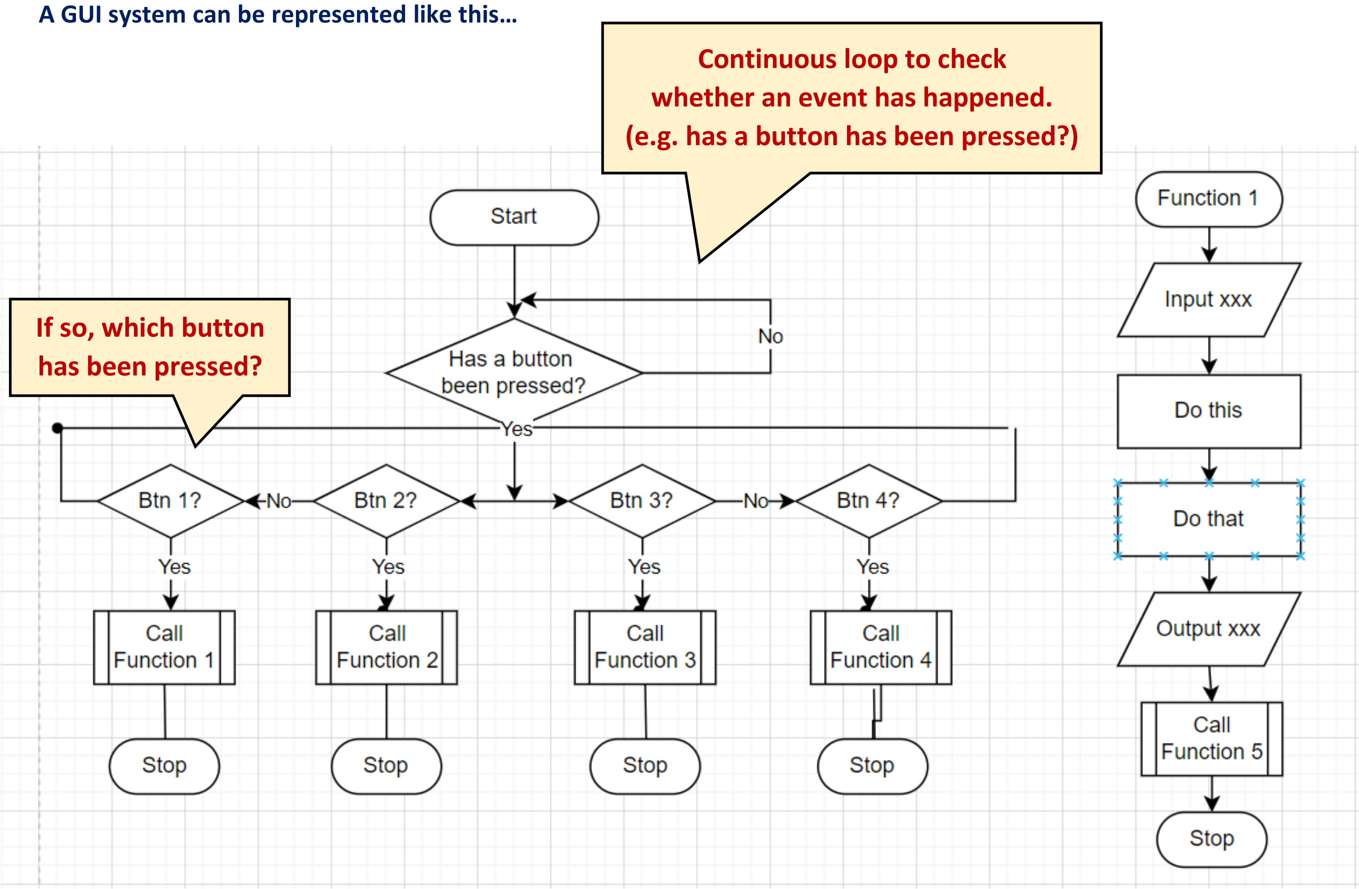
Functions with parameters…



This flowchart calculates the average of three numbers

GUIs (Graphical User Interfaces)

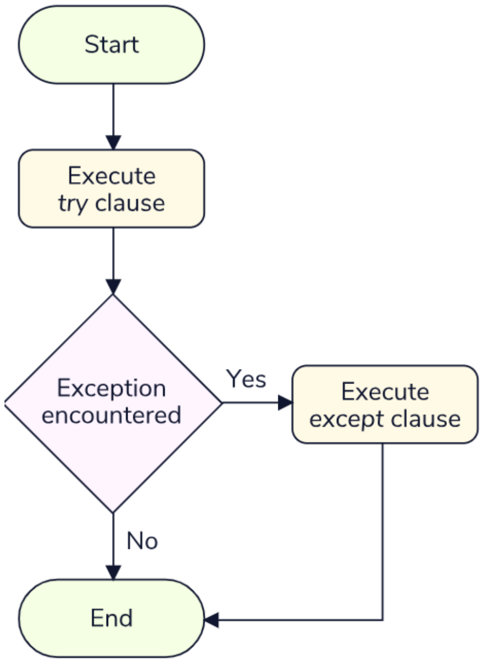
A GUI system can be represented like this…



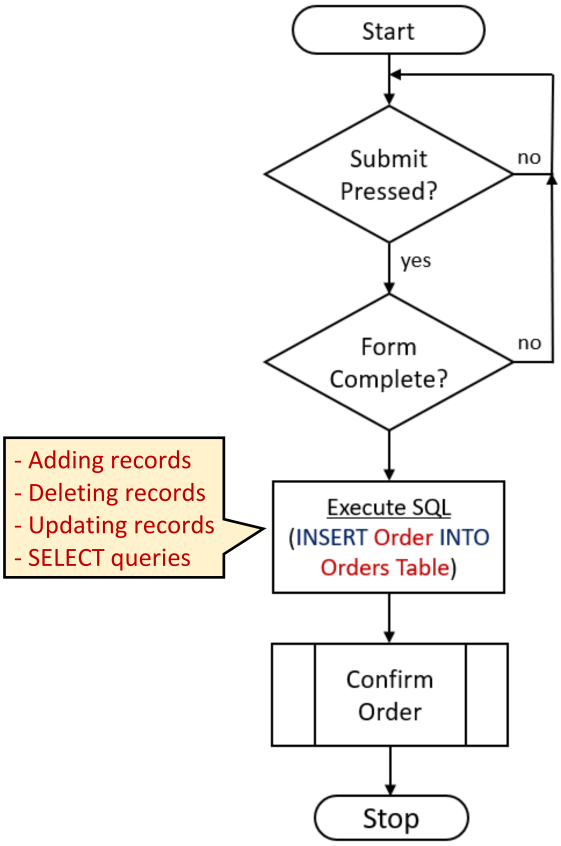
Continuous loop to check whether an event has happened. (e.g. has a button has been pressed?)

If so, which button has been pressed?

Try: Except: Error Handling



Executing SQL statements



- Adding records
- Deleting records
- Updating records
- SELECT queries