A Worked Example

Moving from the architecture developed in the lecture. In summary, we had a program to convert temperatures between different scales. We identified input functionality to include reading in the necessary numbers, validating that it was numeric, and that the input value wasn't lower than absolute 0. The output functionality was simply printing out the value with a label. The conversion functionality to apply the various formulas to make the appropriate conversion.

In the simplest case the input will be a single number. Most programming languages are typed, i.e. you must specify data types. In this case it is a numeric input. Should the input be int, float, or double? The question really is, do you need the decimal or fractional component. If you do the difference in precision between float and double probably won't be an issue. So, who is the intended user? If it's recreational use (such as travel or sporting events then integer values for input and output is probably sufficient. For scientific work a double might be required. Just use the same data type for all variables and return types.

Functional decomposition, now decomposing to programming language functions.

input() get the number to convert, using:
- menu()- offer choices to user
- readValue()- get the number
- validateInput()- is it numeric and greater than absolute 0

convert()- apply the correct formula for:
- F2C()
- C2F()
- And so on…

output() – print the result with appropriate labels

There are some details left to the user. ☺ For example you need the outer structure that passes the data between these parts. s

Yes, this is a simple problem and this may seem like overkill but it illustrates the steps more clearly due to the simplicity.