***Bottom Up vs. Top Down:***

* Bottom-Up: Starts with the most detailed and specific parts and gradually gets broader
* Top-Down: Starts with the broader parts and gets gradually more specific.
  + Requirements Elicitation: Learn what the client needs from the Software
  + Design: Figure out how the software is to be built
  + Development: Actually writing the code
  + Testing: Finding defects and fixing bugs
* Requirements Elicitation can be done both Top-Down and Bottom-Up
* “Precision questioning”: Asking questions that are increasingly more specific
* Bottom up Requirements elicitation is starting with very specific questions and getting gradually broader
* Use Either with the client depending on needs
* Top-Down designs tend to produce the most eloquent and cohesive designs
* Downside: You can paint yourself into a corner
* Development is normally bottom up, Top down normally runs into the problem that testing is postponed
* Top-Down normally goes from System Testing 🡪 Integration Testing 🡪 Unit Testing
* Bottom-Up is the opposite
* Bottom Up development

1. Design program
2. Stub out the program
3. Select a function
4. Review the Design
5. Implement the function
6. Test the function