

Recursion

- Recursion is nothing but using the iterations with functional calls
- To understand recursion prerequisites are
 - ★ functions &
 - ★ memory management (Stack, Heaps)
- Debug to understand the Recursion is one of the best way of understanding it.
- Execution flow of functions:
 - ★ If the function not finished the execution it will remain on stack
 - ★ After finishing function execution, whatever moved over stack will be removed. Flow of program will be stored where function called.
- Almost on same flow recursion works.
 - ★ Recursion should return something
 - ★ A return statement stops execution & avoids infinite loops
 - ★ These are called the base conditions
- There are two types of recursions
 - ★ Direct → 4 types
 - TAIL Recursion
 - HEAD Recursion
 - TREE Recursion
 - NESTED Recursion
- In direct recursion
- Things to remember working with these 3 variables:
 - Arguments
 - return types
 - Body of function
 - local, global variables
- Always takes down required variables