

C Language Tutorial

(Basic to Advanced)

Topics to be covered :

- Installation + Setup
- Chapter 1 - Variables, Data types + Input/Output
- Chapter 2 - Instructions & Operators
- Chapter 3 - Conditional Statements
- Chapter 4 - Loop Control Statements
- Chapter 5 - Functions & Recursion
- Chapter 6 - Pointers
- Chapter 7 - Arrays
- Chapter 8 - Strings
- Chapter 9 - Structures
- Chapter 10 - File I/O
- Chapter 11 - Dynamic Memory Allocation

Pointers

(Chapter 6)

1. Syntax

```
#include<stdio.h>

int main() {
    int age = 22;
    int *ptr = &age;
    int _age = *ptr;
    printf("%d\n", _age);

    //address
    printf("%p\n", &age);
    printf("%p\n", ptr);
    printf("%p\n", &ptr);

    //data
    printf("%d\n", age);
    printf("%d\n", *ptr);
    printf("%d\n", *(&age));
    return 0;
}
```

2. Pointers in Function call

```
# include <stdio.h>

void square(int n);
void _square(int* n);

int main() {
    int number = 4;

    //call by value
    square(number);
    printf("n is : %d\n", number);

    //call by reference
    _square(&number);
    printf("n is : %d\n", number);
    return 0;
}

void square(int n) {
    n = n * n;
    printf("square is : %d\n", n);
}

void _square(int* n) {
    *n = *n * *n;
    printf("square is : %d\n", *n);
}
```

3. Swap 2 numbers

```
# include <stdio.h>

void swap(int a, int b);
void _swap(int* a, int *b);

int main() {
    int x = 3, y = 5;

    //call by value
    swap(x, y);
```

```
printf("x = %d & y = %d\n", x, y);

//call by reference
_swap(&x, &y);
printf("x = %d & y = %d\n", x, y);
return 0;
}

void swap(int a, int b) {
    int t = a;
    a = b;
    b = a;
}

void _swap(int* a, int* b) {
    int t = *a;
    *a = *b;
    *b = *a;
}
```