

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

Strings (Chapter 8)

1. Strings

```
# include <stdio.h>
# include <string.h>

int main() {
    //declaration
    char name[] = "Shradha Khapra";
    char course[] = {'a','p', 'n', 'a', ' ', 'c', 'o', 'l', 'l', 'e', 'g', 'e',
'\0'};

    //printing string
    for(int i=0; name[i] != '\0'; i++) {
        printf("%c", name[i]);
    }
    printf("\n");

    //printing string with pointer
    for(char *ptr=name; *ptr != '\0'; ptr++) {
        printf("%c", *ptr);
    }
}
```

```
}  
  
printf("\n");  
  
//printing using format specifier  
printf("%s\n", name);  
  
//input a string  
char firstName[40];  
printf("enter first name : ");  
scanf("%s", firstName);  
printf("you first name is %s\n", firstName);  
    char fullName[40];  
printf("enter full name : ");  
scanf("%s", fullName);  
printf("you first name is %s\n", fullName);  
  
// gets & puts  
char fullName[40];  
printf("enter full name : ");  
fgets(fullName, 40, stdin);  
puts(fullName);  
  
//Library Functions  
char name[] = "Shradha";  
int length = strlen(name);  
printf("the length of name : %d\n", length);  
  
char oldVal[] = "oldValue";  
char newVal[50];  
strcpy(newVal, oldVal);  
puts(newVal);  
  
char firstStr[50] = "Hello ";  
char secStr[] = "World";  
strcat(firstStr, secStr);  
puts(firstStr);  
  
char str1[] = "Apple";  
char str2[] = "Banana";  
printf("%d\n", strcmp(str1, str2));  
  
//enter String using %c
```

```
printf("enter string : ");
char str[100];
char ch;
int i = 0;

while(ch != '\n') {
    scanf("%c", &ch);
    str[i] = ch;
    i++;
}
str[i] = '\0';
puts(str);

return 0;
}
```

> Some more Qs

```
# include <stdio.h>
# include <string.h>

// void printString(char arr[]);
// int countLength(char arr[]);
// void salting(char password[]);
// void slice(char str[], int n, int m);

//int countVowels(char str[]);

void checkChar(char str[], char ch);

int main() {
    char str[] = "ApnaCollege";
    char ch = 'x';
    checkChar(str, ch);
}

void checkChar(char str[], char ch) {
    for(int i=0; str[i] != '\0'; i++) {
        if(str[i] == ch) {
            printf("character is present!");
            return;
        }
    }
}
```

```
    }  
}  
printf("character is NOT present:");  
}  
  
// int countVowels(char str[]) {  
//     int count = 0;  
  
//     for(int i=0; str[i] != '\0'; i++) {  
//         if(str[i] == 'a' || str[i] == 'e' || str[i] == 'i' ||  
//            str[i] == 'o' || str[i] == 'u') {  
//             count++;  
//         }  
//     }  
//     return count;  
// }  
  
// void slice(char str[], int n, int m) { // n & m are valid value  
//     char newStr[100];  
//     int j = 0;  
//     for(int i=n; i<=m; i++, j++) {  
//         newStr[j] = str[i];  
//     }  
//     newStr[j] = '\0';  
//     puts(newStr);  
// }  
  
// void salting(char password[]) {  
//     char salt[] = "123";  
//     char newPass[200];  
  
//     strcpy(newPass, password); // newPass = "test"  
//     strcat(newPass, salt); // newPass = "test" + "123";  
//     puts(newPass);
```

```
// }

// int countLength(char arr[]) {
//     int count = 0;
//     for(int i=0; arr[i]!='\0'; i++) {
//         count++;
//     }
//     return count-1;
// }

// void printString(char arr[]) {
//     for(int i=0; arr[i] != '\0' ;i++) {
//         printf("%c", arr[i]);
//     }
//     printf("\n");
// }
```