

Shaping "skills" for "scaling" higher...!!!

## WELCOME, PROGRAMMERS



01.

What is Function?



### WHAT IS FUNCTION?





### **FUNCTION**



A function is a self-contained block of code that performs a specific task.

Functions **provide a way** to modularize code, making it more readable, maintainable, and reusable.



02.

What are Types of Functions?



# WHAT ARE TYPES OF FUNCTIONS?





### TYPES OF FUNCTIONS



There are mainly **two types** of **functions** available in C language:

**Built-in Functions** 

**Examples:** printf(), scanf(), gets(), puts()

User Defined Functions (UDF)

**Examples:** red(), white(), rahul(), bob()



Let's see the **syntax** of creating a **UDF**...

RED & WHITE

Multimedia Education

Shaping "skills" for "scaling" higher...!!!

### Syntax of an UDF



```
return_type function_name(parameters)
   // Function body (code)
    return value; // Return statement (optional)
```















































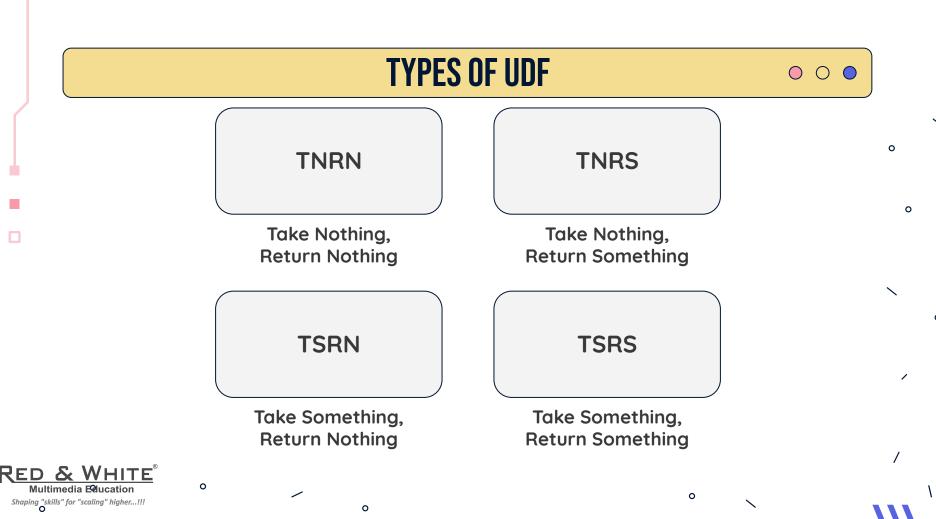












Let's see **each type** in detail...





### **TNRN**

Take Nothing, Return Nothing



### **Example of TNRN**



```
return_type function_name(parameters)
   // Function body (code)
   return value; // Return statement (optional)
```

Example

```
void get_quote()
    printf("Nothing is impossible.");
```



### **Use of TNRN function**



```
#include<stdio.h>
Finds the
         // Function Definition
function
        >void get_quote()-
             printf("Nothing is impossible.");
                                   Replace the output
         void main()
                                   from function body
              get_quote(); // Function Calling
```

Output: Nothing is impossible.





### **TSRN**

Take Something, Return Nothing



### **Example of TSRN**



```
return_type function_name(parameters)
{
    // Function body (code)
    return value; // Return statement (optional)
}
```

Example

void addition(int a, int b)
{
 printf("Addition is %d", a+b);
}



### **Use of TSRN function**



```
#include<stdio.h>
                     Passed in
                    Parameters
void addition(int a, int b)
    printf("Addition is %d", a+b);
                          Replace the output
void main()
                          from function body
     addition(5, 3); // Function Calling
              Arguments
               Passing
```

// Output: Addition is 8





Let's start now...





0