Date / C++ functions

Saathi

Topic's Covered -

- C++ functions
- C++ Function Types
- C++ Function overloading
- C++ Default Argument
- C++ Storage Class
- C++ Recursion
- C++ Return Reference

C++ Functions

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

Suppose we need to create a program to create a circle & color it. We can create two functions to solve this problem.

- A function a draw the circle.
- A function to color the circle.

Dividing a complex problem into smaller chunks makes our problem easy to understand & reusable.

Page No.

There are two types of function.

1. Standerd Library function :
predefine in C++

2. User-define function:
created by users.

C++ User-define function

C++ allows the programmer to define their own function.

A user-define function group Code to perform a specific task & that group of code is given a name (identifier).

When the function is invoked from any part of the program, it all executes the codes, define in the body of the function.

C++ function Decleration

The syntax of function decleration is

```
returntype functionName(parameter1, parameter2...);
{
    // function body
}
```

Here, Example of function declaration.

```
Void greet (){
    Cout<<"Hello Tejas";
}
```

- The name of function is greet()
- The return type of the function is void.
- The empty parameter means it doesn't any parameters.
- The function body is written inside {}.

Note:

Will learn about returnType & parameters later.

Calling a function

In the above program, we have declare a function named greet(). To use the greet() function, we need to call it.

Call the above greet function.

```
int main(){
    // Calling a function
    greet
}
```

We can pass a value parameter while calling the function.

```
int num(){
    int n = 7;

    //Calling the function
    // n is passed to the function as argument
    printNum(n);
    return 0;
}
```

Quick Quiz:

Demonestrate the function with parameter.

Return statement

This means a function is not returning any value.

It's also possible to return a value from a function. for this, we need to specify the returnType of the function during fenction decleration.

Then, the return statement can be used to return vale.

For Example,

```
int add (int a, int b){
    return (a+b);
}
```

Here, we have data type int inside of void. This means that the function return an int value.

The code return (a+b); returns the sum of two parameters as the function value.

The return statement denotes that the function has ended. Any code after return inside the function is not executed.

Function Prototype :

In C++ code, the function declaration should be ~~after~~ before the function call. However, if we want to define a function after the function call, we need to use the function prototype. for example.

```
Void add (int, int);
```

```
returnType functionName (dataType1, dataType2);
```

Date __/__/____ Saathi

C++ Library function

Library function are the build-in-function in c++ programming.

programmers can use library function by invoking the functions directly; they don't need to write function themselves.

Some common library function in c++ function are sqrt(), abs(), isdigit(), etc.

In order to use use library function, we usally need to include the header file in which these library function are defined.

for instance, in order to use mathematical function such as sqrt() & abs(), we need to include the header file cmath.

6.

C++ Function Types

C++ User-defined function Types.

In this tutorial, you will learn about different approaches you can take to solve a single problem using function.

for better understanding of arguments & return in functions, user-defined functions can be categorised as:

- Function with no argument & no return value
- Function with no argument but return value
- Function with argument but no return value.
- function with arguments & return value.

Consider a situation in which you have to check two prime numbers. This problem is solved below by making user-define function in 4 different ways as follows.