7. Shell Scripting – Functions

- Functions are set of instructions to perform specific task and it can be accessed anywhere in the script using function call.
- Functions can be added to any script to make the script modular and flexible. For example, if a feature is to be added or removed from script, if a function is used to add that feature, then it makes the script flexible and makes the process easier.

```
function function_name()
{
    #task to be performed
}
```

- Function should be written with the keyword **function** and every function should have a specific function name.
- Every function can be called anywhere in script using function call format as given below:

> `` `function_name argument1 argument2 . . .` ``
>
> **OR**
>
> **$(function_name argument1 argument2 . . .)**

- Command substitution method should be used to make function call. Only arguments to the function separated with spaces should be passed on function call.
- The result of function call can be even assigned to a variable.

Function formats:

$@	→	**to get all the function arguments**
$#	→	**total number of function arguments**
$i	→	**to access i^{th} argument to function given in the order**

Command line argument formats can be used as function formats inside the function or within { }. When these formats are used outside { } of function, then it will be considered as command line argument formats.