



Functions: Functions are small independent programs that are used in a large program to make the program readable, well understood and easy debuggable. Since 'C' is a function based programming language. The program in it is prepared using functions. 'C' defines useful functions in respective header files as it has a large set of pre-defined header files. Some of the already defined functions have been described below:

[These functions according to their use have been classified into stated categories]

a. Character Data Input/ Output functions

Character Data Input Functions

1. `getch()`
Header File: `conio.h`
Syntax to Use: `<char_var>=getch();`
Example: `char ch=getch();`
Note: Does not write character on screen that has been pressed
2. `getche()`
Header File: `conio.h`
Syntax to Use: `<char_var>=getche();`
Example: `char ch=getche();`
Note: write the character on screen that has been pressed
3. `getchar()`
Header File: `stdio.h`
Syntax to Use:
`<char_var>=getchar();`
Example: `char ch=getchar();`

b. String Data Input/ Output functions

String Data Input Function

`gets()`
Header File: `stdio.h`
Syntax to Use: `gets(<string_var>);`
Example: `gets(name);`

c. Any type of data Input/ Output functions

Any Type of Data Input Function

`scanf()`
Header File: `stdio.h`
Syntax to Use:
`scanf("<Format_strings>",&<var>,...);`
Example: `scanf("%i",&num);`
`scanf("%d",&num);`
`scanf("%c%i%d",&ch,&num,&data);`

Character Data Output Function

1. `putch()`
Header File: `conio.h`
Syntax to Use: `putch(<char_var>);`
Or, `putch('<char_const>');`
Example: `putch(ch);`
`putch('C');`
2. `putchar()`
Header File: `stdio.h`
Syntax to Use:
`putchar(<char_var>);`
Or, `putchar('<char_const>');`
Example: `putchar(ch);`
`putchar('C');`

String Data Output Function

`puts()`
Header File: `stdio.h`
Syntax to Use: `puts(<string_var>);`
Or, `puts("<String_const>");`
Example: `puts(name);`
`puts("Ashutosh Pawan@hdjc");`

Any Type of Data Output Function

`printf()`
Header File: `stdio.h`
Syntax to Use:
`printf("<String_const>");`
or, `printf("<conversion_strings>,<data>,...);`
Example: `printf("Lets C");`
Or,
`printf("%c%d%f",ch,num,data);`