

# C Language Tutorial 2

## Answers

### 1. // or(\*,/\*)

Comments in a program enhance code readability, provide documentation, aid in debugging, and promote effective collaboration among developers.

### 2. main() function

3. The scanf() function is a commonly used input function in the C programming language. It allows you to read input from the user or from a file and store that input in variables of different data types.

4. The C language is case sensitive

### 5. (a) record1 :- valid identifier

(b) 1record :- invalid identifier [It starts with a digit, which is not allowed. Identifier must start with a letter or an underscore.]

(c) file-3 :- invalid identifier [It contains a hyphen, which is not allowed in identifiers. Only letters, digits and underscores are allowed.]

(d) return :- valid identifier

(e)\$tax :- valid identifier

(f)name :- valid identifier

(g)name and address :- invalid identifier [It contains spaces,which are not allowed in identifiers.typically,identifier cannot have spaces and must be a single word or use underscore to separate multiple words.]

(h)name-and-address :- invalid identifier [It contains hyphens,which are not allowed in identifiers.identifiers cannot have hyphens.]

(i)name\_and\_address :- valid identifier

(j)123-45-6789 :- invalid identifier [It starts with a digit,which is not allowed.identifiers must start with a letter or an underscore.additionally,it contains hyphens,which are not allowed in identifiers.]

6. (a>false [ The function 'printf' does not automatically begin printing at the beginning of a new line.by default,it continues printing on the same line unless instructed otherwise.]

(b>false [comments in programming languages are not executed by the computer.they are meant for human readers and are ignored by the compiler or interpreter during program execution.]

(c)True

(d>false [in some programming language s,variables can be used before they are explicitly defined.in c variables must be defined before they are used.]

(e)True

(f)True

(g) false [a program that prints three lines of output does not necessarily need three 'printf' statements. it could use a single 'printf' statement with escape sequences or newlines('\n') to print multiple lines of output.]

7.

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

8.(a) scanf("%d", &value);

(b) printf("The product of %d and %d is %d\n", x, y, product);

(c) scanf("%d", &anInteger);

(d) no error in this statement

(e) printf("The sum is %d\n", x + y);

(f) printf("The value you entered is: %d\n", value);

9. a) 2

b) 4

c) x=

d) x=2

- e)  $5 = 5$
- f) Nothing.
- g) Nothing.
- h) Nothing.
- i) It prints a new line character, resulting in a line break.

10. (a) True

(b) True

(c) False [The statement `'printf("a = 5;");'` is not an assignment statement. It is a printf statement that outputs the text "a = 5;" to the console. An assignment statement in C assigns a value to a variable using the assignment operator "=", such as `'a = 5;'` which assigns the value 5 to the variable 'a'.]

(d) False [A valid arithmetic expression containing no parentheses is evaluated based on operator precedence and associativity rules, not strictly from left to right. In C, certain operators have higher precedence than others, and they are evaluated first. For example, in the expression `'a + b * c'`, the multiplication (`'*'`) has higher precedence than addition (`'+'`), so it will be evaluated first.]

(e) True.