1. **What do you understand by the term “Case Sensitive Language”?**  
   A **case-sensitive language** distinguishes between uppercase and lowercase letters. This means that variables, function names, and other identifiers that differ only in letter case are treated as different entities. For example, in PHP, $var, $Var, and $VAR are three distinct variables because PHP is case-sensitive when it comes to variable names.
2. **How many code declaration blocks can be inserted in a PHP document?**  
   You can insert **as many PHP code declaration blocks** as needed within an HTML document. Each block can be opened and closed with the PHP tags <?php ... ?>. These blocks can be placed anywhere in the HTML, enabling dynamic PHP code to be executed at various points in the document.
3. **Why does the PHP Group recommend that you use standard PHP script delimiters to write PHP code declaration blocks?**  
   The PHP Group recommends using **standard PHP script delimiters** (<?php ... ?>) because they are supported in all configurations of PHP and ensure better **compatibility**. Other delimiters, like <? ... ?> (short tags) or <% ... %> (ASP-style tags), may be disabled on certain servers, which could lead to portability and compatibility issues.
4. **Identify the two types of comments available in PHP and indicate when each would be used.**
   * **Single-line comments**:  
     Use // or # for commenting on a single line. These comments are typically used for brief explanations or inline notes about specific lines of code.
   * // This is a single-line comment

# This is another single-line comment

* + **Multi-line comments**:  
    Use /\* ... \*/ for comments that span multiple lines. These are generally used for longer explanations or for commenting out large sections of code during debugging.

Explain

/\*

This is a multi-line comment

that spans multiple lines.

\*/

1. **What is the difference between the echo and print statement in PHP?**
   * **echo**:
     + Can output **one or more strings**.
     + Slightly faster because it doesn't return a value.
     + Usage: echo "Hello World";
   * **print**:
     + Outputs **only one string** and always returns a value of 1.
     + Slightly slower than echo due to the return value.
     + Usage: print "Hello World"; Overall, echo is preferred when you need to output more than one value, while print can be used when you need a return value.
2. **Write names list of PHP program structure.** The structure of a PHP program generally includes the following components:
   * **PHP tags**: <?php ... ?>
   * **Variables**: To store data.
   * **Constants**: To define values that cannot change.
   * **Operators**: For calculations, assignments, comparisons, etc.
   * **Control structures**: if, else, switch, for, while, etc., for decision-making and loops.
   * **Functions**: Predefined and user-defined for reusable code.
   * **Classes and Objects**: For Object-Oriented Programming (OOP).
   * **Comments**: Single-line and multi-line comments for code documentation.
   * **Output statements**: Like echo or print to display information.