QUESTIONS TASK:

1) **Definition of Type:**
   "Type" in programming refers to how data is categorized according to its characteristics and the actions that can be done on it. The definition is divided into two sections:
   - **Data Type:** Indicates the type of data that can be stored in a variable, including characters, integers, and floating-point numbers.
   - **Type System:** Enforces rules on how different types can be combined and manipulated in a program.

2) **Difference between Strongly and Weakly Typed Languages:**
   - **Strongly Typed Language:**
     - Variables are bound to a specific data type.
     - Type checking is enforced at compile-time, and implicit type conversions are limited.
     - Examples include C++, Java, and Python.
   - **Weakly Typed Language:**
     - Variables are not strictly bound to a specific data type.
     - Type checking may be deferred until runtime, and implicit type conversions are more permissive.
     - Examples include JavaScript, PHP, and Perl.

3) **Reason for Fewer Bugs in Strongly Typed Languages:**
   - **Type Safety:** Strongly typed languages catch type-related errors at compile-time, reducing the chances of runtime errors.
   - **Explicitness:** Code is easier to read and programmers can better understand the intended data types when type declarations are explicit.
   - **Prevents Unintended Conversions:** Strict type enforcement prevents unintended type conversions, reducing unexpected behaviour.
   - **Early Detection:** By identifying type issues early in the development process, runtime failures are decreased and debugging is made simpler.