1. **Advantages of Functions:**
   * Reusability, modularity, abstraction, and organization.
2. **When Code in a Function Runs:**
   * When the function is called.
3. **Statement Creating a Function:**
   * The **def** statement.
4. **Difference Between a Function and a Function Call:**
   * A function is a defined block of code; a function call is the execution of that code.
5. **Global and Local Scopes:**
   * One global scope, and each function call creates a new local scope.
6. **Local Scope Variables After Function Call:**
   * Local variables are destroyed when the function call returns.
7. **Concept of a Return Value:**
   * A value a function can send back. Yes, it's possible to have it in an expression.
8. **No Return Statement in a Function:**
   * The return value is **None**.
9. **Making a Function Variable Refer to a Global Variable:**
   * Use the **global** keyword. Example: **global\_var = 10; def my\_func(): global global\_var**.
10. **Data Type of None:**
    * The data type is **NoneType**.
11. **Import Sentence:**
    * It doesn't do anything; it's a placeholder.
12. **Calling bacon() Feature in Spam Module:**
    * After importing spam: **spam.bacon()**.
13. **Saving Program from Crashing on Error:**
    * Use error handling techniques like try-except blocks.
14. **Purpose of Try and Except Clauses:**
    * **try** is used to enclose code that might raise an exception; **except** catches and handles the exception.