**Stored Procedures in T-SQL**

**Introduction to Stored Procedures**

Stored procedures in T-SQL are a powerful feature of SQL Server. They allow you to encapsulate SQL code, which can be executed repeatedly. Stored procedures are beneficial for several reasons:

* Performance: They are compiled and stored in the database, leading to faster execution times.
* Security: They provide an additional layer of security by restricting direct access to the data.
* Maintainability: Centralizing business logic in stored procedures makes changes easier and more consistent.

**What can you write inside Stored Procedure:**  
In T-SQL, which is the SQL language variant used by Microsoft SQL Server, stored procedures can contain a wide range of SQL statements, control structures, and special features. Here's a detailed list of what you can write inside stored procedures in T-SQL:

* SQL Queries and DML Statements: This includes SELECT, INSERT, UPDATE, DELETE, and MERGE statements for data querying and manipulation.
* Variable Declarations and Assignments: You can declare local variables using the DECLARE statement and set values with the SET or SELECT statements.
* Control Flow Statements:
  + IF...ELSE: For conditional logic.
  + WHILE: For looping.
  + BEGIN...END: To define blocks of code.
  + WAITFOR: To delay execution.
  + GOTO: For jumping to a labeled point in the procedure (though generally discouraged due to readability concerns).
* Error Handling and Transactions:
  + TRY...CATCH: For catching and handling exceptions.
  + TRANSACTION Management: Using BEGIN TRANSACTION, COMMIT, and ROLLBACK to handle transactions.
* Dynamic SQL Execution: Using EXEC or sp\_executesql to execute dynamically built SQL strings.
* Calling Other Stored Procedures and Functions: You can call other stored procedures or user-defined functions within a stored procedure.
* Temporary Tables and Table Variables: You can create and use temporary tables and table variables for intermediate data storage and manipulation.
* Cursor Management: Although generally less efficient than set-based operations, cursors for row-by-row processing are supported in T-SQL.
* System Stored Procedures and Functions Calls: T-SQL allows calling system stored procedures and functions for various tasks.
* Output Parameters: Stored procedures can have output parameters to return data back to the caller.
* RAISERROR or THROW: For generating custom error messages.
* Use of Table-Valued Parameters: Allows passing tables as parameters to stored procedures.
* Common Table Expressions (CTEs): These can be defined within stored procedures for recursive queries or organizing complex queries.
* Use of DDL Statements: Such as CREATE, ALTER, or DROP, typically for temporary objects or within dynamic SQL.
* XML Handling: T-SQL supports XML data manipulation and querying.
* Text and Image Manipulation: Though older and less recommended, T-SQL supports manipulation of text and image data types.

It's important to use best practices while writing stored procedures in T-SQL, such as avoiding unnecessary cursors, ensuring proper error handling, and preventing SQL injection when using dynamic SQL. The capabilities and syntax may evolve with different versions of SQL Server, so always refer to the specific version's documentation for the most accurate information.