Explanation of Keywords in C++

* **Asm:** It is used to declare a block of code that has to be passed to the assembler.
* **auto:** This keyword is a storage class specifier that is used for defining objects in a particular block.
* **break:** This statement terminates any switch statement or any loop.
* **case:** This keyword is used specifically within a switch statement to specify a match for the expression of the statement.
* **catch:** It specifies which actions have to be taken when an exception occurs.
* **char:** This is one of the fundamental data types in C++ language that defines character objects.
* **class:** It is used to declare a user-defined data type that encapsulates any data members and operations or member functions of a particular class.
* **const:** This keyword helps to define objects whose value will not alter throughout the lifetime of execution of that particular program.
* **continue:**It transfers control to the starting point of a loop.
* **default:** This keyword handles expression values in a switch statement that could not be handled by case.
* **delete:**It is a memory deallocation operator.
* **do:**indicate the start of a do-while statement in which the sub-statement is executed repeatedly until the value of the expression is logical-false.
* **double:**Fundamental data type used to define a floating-point number.
* **else:**Used specifically in an if-else statement.
* **enum:**To declare a user-defined enumeration data type.
* **extern:** An identifier specified as an extern has an external linkage to the block.
* **float:** Fundamental data type used to define a floating-point number.
* **for:** Indicates the start of a statement to achieve repetitive control.
* **friend:** A class or operation whose implementation can access the private data members of a class.
* **long:** A data type modifier that defines a 32-bit int or an extended double.
* **new:** Memory allocation operator.
* **operator:** Overloads a C++ operator with a new declaration.
* **private:**Declares class members who are not visible outside the class.
* **protected:**Declares class members who are private except to derived classes
* **public:** Declares class members who are visible outside the class.
* **register:**A storage class specifier that is an auto specifier, but which also indicates to the compiler that an object will be frequently used and should therefore be kept in a register.
* **goto:**This keyword helps to transfer the power of the control to a specified label.
* **if:**It indicates the starting point of an if statement to achieve selective control.
* **inline:** A function specifier that indicates to the compiler that inline substitution of the function body is to be preferred to the usual function call implementation.
* **int: f**undamental data type used to define integer objects.
* **return:** Returns an object to a function’s caller.
* **short:** A data type modifier that defines a 16-bit int number.
* **signed:**A data type modifier that indicates an object’s sign is to be stored in the high-order bit.
* **sizeof:** Returns the size of an object in bytes.
* **static:** The lifetime of an object-defined static exists throughout the lifetime of program execution.
* **struct:**To declare new types that encapsulate both data and member functions.
* **switch:**This keyword is used in the switch statement.
* **template:** parameterized or generic type.
* **this:** A class pointer points to an object or instance of the class.
* **throw:** Generate an exception.
* **try:** Indicates the start of a block of exception handlers.
* **typedef:** Synonym for another integral or user-defined type.
* **union:**Similar to a structure, struct, in that it can hold different types of data, but a union can hold only one of its members at a given time.
* **unsigned:** A data type modifier that indicates the high-order bit is to be used for an object.
* **virtual:** A function specifier that declares a member function of a class that will be redefined by a derived class.
* **void:** This keyword identifies the absence of a type or function parameter list.
* **volatile:** This keyword defines an object that may vary in value in a way that is undetectable to the compiler.
* **while:**This keyword helps to start a while statement and end a do...whileloop.