Nasir Firdaus Opeyemi
17/sci01/051

Firstly, what is Lexeme?
A lexeme: is a sequence of characters in the source program that matches the pattern for a token and is identified by the lexical analyzer as an instance of that token.Hence constitute an instance of a **token**); for **example** if matches the pattern for if , and foo123bar matches the pattern for id. Examples of Lexemes are "float", "abs_zero_Kelvin", "=", "-", "273", ";" .

What is a token?
A token: is a pair consisting of a token name and an

optional attribute value. The token name is an abstract symbol representing a kind of lexical unit, e.g., a particular keyword, or sequence of input characters denoting an identifier. The token names are the input symbols that the parser processes. Examples are;

- Punctuation tokens (IF, void, return, . . .)
- Alphabetic tokens (keywords)

Below shows some of the major differences between Lexemes and Tokens;

Lexemes	Tokens
Lexemes are the words derived from	Tokens are lexemes mapped into

the character input stream.	a token-name and an attribute-value.
A lexeme is the actual character sequence forming a token.	The token is the general class that a lexeme belongs to.
A sequence of characters in the source program that is matched by the pattern for a token.	Sequence of characters that have a collective meaning.
Examples of Lexemes are “float”, “abs_zero_Kelvin”, “=”,	Example of tokens: • Type token (id,

“-”, “273”, “;” .	number, real, . . .) • Punctuation tokens (IF, void, return, . . .) • Alphabetic tokens (keywords)