Documentation:

CodeParser

Attributes:

- `filename`: The name of the code file to parse.
- `token_file`: The name of the file with predefined tokens (reserved words, operators, and separators).
- `constant_table`: An instance of `SymbolTableConstants` to store constants.
- `identifiers_table`: An instance of `SymbolTableIdentifiers` to store identifiers.
- `pif`: A list representing the Position Indicator File (PIF), containing token-type and position pairs.
- `constants_count`: Tracks the count of unique constants.
- `tokens`: A dictionary of predefined tokens, including `reserved_words`, `operators`, and `separators`.

Methods:

- `__init__(filename, token_file)`: Initializes the `CodeParser` with specified filenames, and loads tokens.
- `parse_file()`: Parses the file line by line, classifying tokens into constants, reserved words, operators, separators, or identifiers. If an invalid token is found, a lexical error is displayed.
- `_load_tokens()`: Loads reserved words, operators, and separators from `token_file` into `tokens`.
- `_process_constant(token)`: Checks if a token is a constant and, if new, adds it to `constant_table` and `pif`.
- `_process_reserved_operator_separator(token)`: Checks if a token is a reserved word, operator, or separator, and adds it to `pif`.
- `_process_identifier(token)`: Checks if a token is an identifier and, if new, adds it to `identifiers_table` and `pif`.
- `_write_output()`: Writes symbol tables (identifiers and constants) to `ST.out` and PIF entries to `PIF.out`.