Report on Python Calculator Code

1. Grammar of the Language
The grammar of the Python language used in the provided calculator code can be described as follows:

- Tokens: The basic elements of the language such as keywords, identifiers, operators, and literals.
- Syntax: The rules defining the structure of valid expressions and statements in the language.
- Expressions: Rules for forming valid arithmetic expressions using numbers, variables, and operators like + and *.
- Statements: Rules for defining classes, functions, and control flow structures like loops and conditionals.
- Input/Output: Mechanisms for taking user input and printing output to the console.

2. Type of Parser
The Python language uses a **lexical analyzer** (lexer) and a **parser** to analyze and interpret the code. The lexer tokenizes the input code into meaningful tokens, while the parser analyzes the tokens according to the grammar rules to build a parse tree, determining the structure and validity of the code.

3. Method of Translation and Integration
The provided code does not involve translation from one language to another, but it does involve the translation of infix notation expressions into prefix and postfix notations, which can be considered a form of transformation. This translation is achieved using the Shunting Yard algorithm, which converts infix expressions into postfix expressions. The postfix expression is then evaluated to produce the result.

Integration of the parser and translation methods is inherent in the structure of the code. The `Calculator` class encapsulates methods for parsing user input, converting infix expressions to postfix and prefix notations, and evaluating expressions. These methods are seamlessly integrated into the `run()` method, which orchestrates the functionality of the calculator by presenting options to the user, parsing user input, and performing the necessary operations based on the input.

In summary, the provided Python code demonstrates the use of lexical analysis, parsing, and transformation techniques to implement a simple calculator program capable of evaluating arithmetic expressions and converting them to different notations.