Compiler_Sessional_Report

June 5, 2024

1 Write a program that takes character stream as input and find the keywords, identifier , constants and comments.

[2]:
```
import re  # Import the regular expression module for pattern matching
import keyword  # Import the keyword module to work with Python keywords

# Function to take input from the user
def take_input_stream():
    lines = []
    print("Enter input (type '$' anywhere in a line to finish):")
    while True:
        line = input()  # Read a line from the user
        if "$" in line:  # Check for termination character
            line = line[:line.index("$")].strip()  # Remove anything after '$'␣
 ↪and strip whitespace
            lines.append(line)
            break
        lines.append(line.strip())  # Add the stripped line to the list
    return lines

# Function to count occurrences of Python keywords in the input lines
def count_keyword_occurrences(input_lines):
    keyword_counts = {}
    python_keywords = keyword.kwlist  # Get the list of Python keywords
    for line in input_lines:
        for word in line.split():  # Split each line into words
            if word in python_keywords:  # Check if the word is a Python keyword
                keyword_counts[word] = keyword_counts.get(word, 0) + 1  #␣
 ↪Increment the count for the keyword
    return keyword_counts

# Function to find and count identifiers, constants, and comments in the input␣
 ↪lines
def find_identifiers_constants_and_comments(input_lines):
    identifiers = {}
    constants = {}
    comments = []
```

```
    identifier_pattern = re.compile(r'^[a-zA-Z_][a-zA-Z0-9_]*$')  # Pattern for␣
↪valid Python identifiers
    constant_pattern = re.compile(r'^\d+(\.\d+)?$')  # Pattern for numeric␣
↪constants

    for line in input_lines:
        # Check for comments in the line
        comment_start = line.find('#')
        if comment_start != -1:
            comments.append(line[comment_start:].strip())  # Extract the␣
↪comment part of the line
            line = line[:comment_start].strip()  # Remove the comment part from␣
↪the line for further processing

        for word in re.findall(r'\b\w+\b', line):  # Find all words in the line
            if keyword.iskeyword(word):  # Skip if the word is a Python keyword
                continue
            elif re.match(constant_pattern, word):  # Check if the word is a␣
↪constant
                constants[word] = constants.get(word, 0) + 1  # Increment the␣
↪count for the constant
            elif re.match(identifier_pattern, word):  # Check if the word is an␣
↪identifier
                identifiers[word] = identifiers.get(word, 0) + 1  # Increment␣
↪the count for the identifier

    return identifiers, constants, comments

# Main function to drive the program
def main():
    input_lines = take_input_stream()  # Get the input lines from the user
    keyword_counts = count_keyword_occurrences(input_lines)  # Count keyword␣
↪occurrences
    # Find identifiers, constants, and comments
    identifiers, constants, comments =␣
↪find_identifiers_constants_and_comments(input_lines)

    # Print the received input
    print("\nReceived input:")
    for line in input_lines:
        print(line)

    # Print the occurrences of Python keywords
    print("\nOccurrences of Python keywords:")
    for keyword, count in keyword_counts.items():
        print(f"{keyword}: {count}")
```

```
    # Print the occurrences of identifiers
    print("\nOccurrences of Identifiers:")
    for identifier, count in identifiers.items():
        print(f"{identifier}: {count}")

    # Print the occurrences of constants
    print("\nOccurrences of Constants:")
    for constant, count in constants.items():
        print(f"{constant}: {count}")

    # Print the occurrences of comments
    print("\nOccurrences of Comments:")
    for comment in comments:
        print(comment)

# Entry point of the program
if __name__ == "__main__":
    main()
```

```
Enter input (type '$' anywhere in a line to finish):
#Compiler Sessional
Today is thursday.
CSE421 Compiler Sessional class will start at 2 p.m.
if it rains we goto the campus with umbrella and enter into the class
#End of program$

Received input:
#Compiler Sessional
Today is thursday.
CSE421 Compiler Sessional class will start at 2 p.m.
if it rains we goto the campus with umbrella and enter into the class
#End of program

Occurrences of Python keywords:
is: 1
class: 2
if: 1
with: 1
and: 1

Occurrences of Identifiers:
Today: 1
thursday: 1
CSE421: 1
Compiler: 1
Sessional: 1
```

```
will: 1
start: 1
at: 1
p: 1
m: 1
it: 1
rains: 1
we: 1
goto: 1
the: 2
campus: 1
umbrella: 1
enter: 1
into: 1

Occurrences of Constants:
2: 1

Occurrences of Comments:
#Compiler Sessional
#End of program
```