Unit Test:

1. main

Path [1,2,5,6,7,8]

Input: no arguments

Output: no output.

Path [1,3,5,6,7,8]

- Input: Run with java Printtokens valid_file.txt (file contains tokens).
- Output: Prints tokens from valid_file.txt.

Path [1,4,5,6,7,8]

- Input: Run with java Printtokens arg1 arg2.
- Output: Error! Please give the token stream.

---

2. open_token_stream

Path [1,2,4]

- Input: fname = null.
- Output: Returns BufferedReader for System.in.

Path [1,3,4]

- Input: fname = "valid_file.txt".
- Output: Returns BufferedReader for the file.

---

3. open_character_stream

Path [1,2,4]

- Input: fname = null.
- Output: BufferedReader for stdin.

Path [1,3,4]

- Input: fname = "valid_file.txt".

- Output: BufferedReader for the file.

---

4. get_token

Path [1,2]

- Input: Empty input.
- Output: null.

Path [1,3,4,5,6]

- Input: "123 ".
- Output: "123".

Path [1,3,4,5,7,8]

- Input: ";".
- Output: ";".

Path [1,3,4,5,7,9,10,11,12,13,14]

- Input: "\"hello\"".
- Output: "hello".

Path [1,3,4,5,7,9,10,11,12,13,15,16,18,19,20]

- Input: ";comment".
- Output: ";comment".

Path [1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,22]

- Input: "x".
- Output: "x".

Path [1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,25]

- Input: "12a".
- Output: "12a".

Path [1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,26]

- Input: "\"hello (unterminated string).

- Output: "hello.

Path [1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,27,28]

- Input: "#a".
- Output: "#a".

Path [1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,27,29]

- Input: "$".
- Output: "$".

---

5. get_char

Path [1,2]

- Input: BufferedReader with EOF.
- Output: -1.

---

6. unget_char

Path [1,2]

- Input: Any character.
- Output: 0 (no error).

---

7. is_spec_symbol

Path [1,2]

- Input: '('.
- Output: true.

Path [1,3,4]

- Input: ')'.
- Output: true.

Path [1,3,5,6]

- Input: '['.
- Output: true.

Path [1,3,5,7,8]

- Input: ']'.
- Output: true.

Path [1,3,5,7,9,10]

- Input: '/'.
- Output: true.

Path [1,3,5,7,9,11,12]

- Input: "` .
- Output: true.

Path [1,3,5,7,9,11,13,14]

- Input: ',.
- Output: true.

Path [1,3,5,7,9,11,13,15]

- Input: 'x'.
- Output: false.

---

8. is_token_end

Path [1,2]

- Input: res = -1.
- Output: true.

Path [1,3,4,5]

- Input: str_com_id = 1, ch = '"'.
- Output: true.

Path [1,3,4,6]

- Input: str_com_id = 2, ch = '\n'.
- Output: true.

Path [1,3,7,11,12]

- Input: is_spec_symbol(ch) = true.
- Output: true.

Path [1,3,7,11,13,15]

- Input: ch = ' '.
- Output: true.

---

9. print_token

Path [1,3,4]

- Input: "if".
- Output: keyword,"if".

Path [1,3,5,6]

- Input: "(".
- Output: lparen.

Path [1,3,5,7,8]

- Input: "varName".
- Output: identifier,"varName".

Path [1,3,5,7,9,10]

- Input: "12345".
- Output: numeric,12345.

Path [1,3,5,7,9,11,12]

- Input: ""hello"".

- Output: string,"hello".

Path [1,3,5,7,9,11,13,14]

- Input: "'x'".
- Output: character,"x".

Path [1,3,5,7,9,11,13,15,16]

- Input: ";this is a comment".
- Output: comment,";this is a comment".

Path [1,3,5,7,9,11,13,15,17]

- Input: "$".
- Output: error,"$".

---

10. token_type

Path [1,2]

- Input: "if".
- Output: keyword.

Path [1,3,4]

- Input: ";".
- Output: spec_symbol.

Path [1,3,5,6]

- Input: "x".
- Output: identifier.

Path [1,3,5,7,8]

- Input: "123".
- Output: num_constant.

Path [1,3,5,7,9,10]

- Input: "\"hello\"".
- Output: str_constant.

Path [1,3,5,7,9,11,12]

- Input: "#a".
- Output: char_constant.

Path [1,3,5,7,9,11,13,14]

- Input: ";comment".
- Output: comment.

Path [1,3,5,7,9,11,13,14,15]

- Input: "$".
- Output: error.

---

11. print_spec_symbol

Path [1,2]

- Input: "(".
- Output: lparen..

Path [1,3,4]

- Input: ")".
- Output: rparen..

Path [1,3,5,6]

- Input: "[".
- Output: lsquare..

Path [1,3,5,7,8]

- Input: "]".
- Output: rsquare..

Path [1,3,5,7,9,10]

- Input: """.
- Output: quote..

Path [1,3,5,7,9,11,12]

- Input: "`".
- Output: bquote..

Path [1,3,5,7,9,11,13,14]

- Input: ",".
- Output: comma..

Path [1,3,5,7,9,11,13,15,16]

- Input: "x".
- Output: No output (invalid spec symbol).

---

12. is_keyword

Path [1,2]

- Input: "if".
- Output: true.

Path [1,3]

- Input: "foo".
- Output: false.

---

13. is_identifier

Path [1,7]

- Input: "x1".
- Output: false.

Path [1,2,6]

- Input: "1x".
- Output: false.

Path [1,2,3,5]

- Input: "x".
- Output: false.

Path [1,2,3,4,2]

- Input: "x$".
- Output: false.

---

14. is_num_constant

Path [1,7]

- Input: "123".
- Output: true.

Path [1,2,6]

- Input: "a123".
- Output: false.

Path [1,2,3,5]

- Input: "12a".
- Output: false.

Path [1,2,3,4,2]

- Input: "12".
- Output: true.

---

15. is_str_constant

Path [1,7]

- Input: "\"hello\"".

- Output: true.

Path [1,2,6]

- Input: "hello".
- Output: false.

Path [1,2,3,5]

- Input: "\"hello (unterminated).
- Output: false.

Path [1,2,3,4,2]

- Input: "\"\"".
- Output: true.

---

16. is_char_constant

Path [1,2]

- Input: "#a".
- Output: true.

Path [1,3]

- Input: "a".
- Output: false.

---

17. is_comment

Path [1,2]

- Input: ";comment".
- Output: true.

Path [1,3]

- Input: "comment".
- **Output**: false.

End-To-End Test:

1. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,2] -> print_token[1-16]

- *Scenario*: No filename, read from stdin (open_character_stream[1,2,4]), no tokens (EOF).

2. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,6] -> print_token[1-16]

- *Scenario*: Token ends at whitespace (e.g., 123).

3. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,8 -> is_spec_symbol[1,2]] -> print_token[1-16]

- *Scenario*: Token is a special symbol (e.g., +).

4. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,14 -> is_token_end[1,2]] -> print_token[1-16]

- *Scenario*: String token (e.g., "hello").

5. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,18,19,20 -> is_token_end[1,3,7,11,13,15]] -> print_token[1-16]

- *Scenario*: Comment token (e.g ;comment).

6. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,22 -> unget_char[1,2]] -> print_token[1-16]

- *Scenario*: Identifier with unget_char failure (e.g., x).

7. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,25 -> is_num_constant[1,7]] -> print_token[1-16]

- *Scenario*: Invalid numeric constant (e.g., 12a).

8. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,26 -> is_str_constant[1,7]] -> print_token[1-16]

- *Scenario*: Unterminated string (e.g., "hello).

9. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,27,28 -> is_char_constant[1,2]] -> print_token[1-16]

- *Scenario*: Valid character constant (e.g., #a).

10. Main[1,2,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,17,19,21,23,27,29] -> print_token[1-16 -> token_type[1,3,5,7,9,11,13,14,15]]

- *Scenario*: Invalid token (e.g., $).

11. Main[1,3,5,6 -> open_token_stream[1,3 -> open_character_stream[1,3,4],4] -> get_token[1,2] -> print_token[1-16]

- *Scenario*: Valid file input, no tokens (EOF).

12. Main[1,3,5,6 -> open_token_stream[1,3 -> open_character_stream[1,3,4],4] -> get_token[1,3,4,5,6] -> print_token[1-16]

- *Scenario*: Token ends at whitespace in file (e.g., 456).

13. Main[1,3,5,6 -> open_token_stream[1,3 -> open_character_stream[1,3,4],4] -> get_token[1,3,4,5,7,8 -> is_spec_symbol[1,3,5,7,9,10]] -> print_token[1-16 -> token_type[1,3,4]]

- *Scenario*: Special symbol in file (e.g., ;).

14. Main[1,3,5,6 -> open_token_stream[1,3 -> open_character_stream[1,3,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,14 -> is_token_end[1,3,4,6]] -> print_token[1-16 -> token_type[1,3,5,7,9,10]]

- *Scenario*: String token from file (e.g., "world").

15. Main[1,3,5,6 -> open_token_stream[1,3 -> open_character_stream[1,3,4],4] -> get_token[1,3,4,5,7,9,10,11,12,13,15,16,18,19,20 -> is_token_end[1,3,7,11,12]] -> print_token[1-16 -> token_type[1,3,5,7,9,11,13,14]]

- *Scenario*: Comment token from file (e.g., ;file_comment).

16. Main[1,4,5,6 -> open_token_stream[1,2 -> open_character_stream[1,2,4],4] -> get_token[1,3,4,5,6] -> print_token[1-16]
- *Scenario*: Invalid args (length >1) → error message, but token parsed.