

/*

Abhi Avasarala

Dec 13

Punctuation.java

Period 5

Pseudocode - This program searches through an entire section of text and outputs the words that have punctuation marks surrounding them. This program uses charAt, lastIndexOf, and substring as string methods to search for the punctuation, check if its present next to a string and to find the index of the spaces as well.

- Main: calls searchIt
- searchIt - calls printWords
- getPunctuationWords - Store text in variable, check for punctuation by calling checkForPunctuation method, store checkForPunctuation method return value in variable
- checkForPunctuation - checks the index value of the spaces and punctuation. Returns the value if there is no space.
- Print - prints out the words

*/

```
public class Punctuation {
    public static void main(String[] args) { //main method
        Punctuation punctuation = new Punctuation(); //instance of class
        punctuation.searchIt(); //calls searchIt
    }

    public void searchIt() {
        String print = getPunctuationWords();
        printWords(print);
    }

    public String getPunctuationWords() {
        String text = new String("Blood, Sweat, and Tears by Winston Churchill "
            + "May 13, 1940 " +
            "Mr. Speaker:" +
            "On Friday evening last I received His Majesty's commission to form a new " +
            "Administration. It was the evident wish and will of Parliament and the nation that this
            should be conceived on the broadest possible basis and that it should include all parties, both
            those who supported the late Government and also the parties of the Opposition." +
            "I have completed the most important part of this task. "+
            "A War Cabinet has been formed of five Members, representing, with the Liberal
            Opposition, the unity of the nation. "+
```

"The three party Leaders have agreed to serve, either in the War Cabinet or in high executive office. The three Fighting Services have been filled. "+

"It was necessary that this should be done in one single day, on account of the extreme urgency and rigour of events. "+

"A number of other key positions were filled yesterday, and I am submitting a further list to His Majesty tonight. "+

"I hope to complete the appointment of the principal Ministers during tomorrow. "+

"The appointment of the other Ministers usually takes a little longer, but I trust that, when Parliament meets again, this part of my task will be completed, and that the Administration will be complete in all respects." +

"Sir, I considered it in the public interest to suggest that the House should be summoned to meet today. "+

"Mr. Speaker agreed and took the necessary steps, in accordance with the powers conferred upon him by the Resolution of the House. "+

"At the end of the proceedings today, the Adjournment of the House will be proposed until Tuesday, the 21st May, with, of course, provision for earlier meeting, if need be. "+

"The business to be considered during that week will be notified to Members at the earliest opportunity. "+

"I now invite the House, by the Resolution which stands in my name, to record its approval of the steps taken and to declare its confidence in the new Government.");

```
String x = "";    //New empty string
for (int i = 0; i < text.length(); i++) { //loop through characters in text
    x += checkForPunctuation(text, i);
}
return x;
}

public String checkForPunctuation(String text, int i) {
    int punctuationIndex = -1; //Find index num of punctuation
    int spaceIndex = -1; //Find index num of space
    if (text.charAt(i) == '!' || text.charAt(i) == '.' || text.charAt(i) == '?' || text.charAt(i) == "\"
        || text.charAt(i) == \"\\\" || text.charAt(i) == ',' || text.charAt(i) == ';' || text.charAt(i) == ':') {
//check all the punctuation marks
        punctuationIndex = i; //find index of punctuation
        spaceIndex = text.lastIndexOf(' ', i); //find index of space
        if (spaceIndex == -1)
            spaceIndex = 0;
        return text.substring(spaceIndex, punctuationIndex + 1); //Return substring
    }
    return " ";
}

public void printWords(String text) {
    System.out.println(text); //print group of words
```

}
}