Präsenzaufgaben 2

14.10.2021

Die Lösung der Aufgaben wird am Ende der Übung von Ihnen vorgestellt.

Kommentare löschen

Programmieren Sie eine Klasse

```
public class JavaCode
```

die eine Java-Source-Code-Datei (.java) einlesen und ausgeben kann. Die Source-Code-Datei wird intern in einer `ArrayList<String>` gespeichert. Fertigen Sie außerdem für jede Teilaufgabe ein Testbeispiel an.

Aufgabe 1: Schreiben Sie die folgenden Methoden:

```
public void ladeCode(File f)
//laedt einen Java-Quelltext aus der angegebenen Datei

public void schreibeCode(File f)
//Schreibt den Code in die angegebene Datei
```

Die beiden Methoden sollen dabei die folgenden Hilfsmethoden aufrufen (die auch in den nächsten Aufgaben noch gebraucht werden):

```
private void ladeCode(Scanner sc)
//laedt einen Java-Quelltext mit Hilfe des angegebenen Scanners

private void schreibeCode(PrintWriter pw)
//Schreibt den Code mit Hilfe des angegebenen PrintWriters
```

Eine Testdatei `KommentarTest.java` finden Sie auf der Veranstaltungsseite.

Aufgabe 2: Holen Sie sich Ihre Java-Dateien direkt aus dem Web. Benutzen Sie dazu folgenden Code:

```
public void ladeCode(URL url) {
    URLConnection con = url.openConnection();
    Object content = con.getContent();
    Scanner sc = new Scanner((InputStream)content);
    //Setzen Sie die Methode fort.
```

Nehmen Sie zum Test die URL der Testdatei auf der Veranstaltungsseite.

Aufgabe 3: Schreiben Sie zwei Methoden

```
public void ladeCode()
public void schreibeCode()
```

die ein Dateiauswahlfenster öffnen, um den vom Benutzer gewünschten Dateinamen zu ermitteln. Benutzen Sie dazu den folgenden Code. Die ausgewählte Datei steht in der Variablen `file`:

```
File file = null;
//Dateiauswahlfenster auf derzeitigen Projektordner
JFileChooser fc = new JFileChooser(new File(System.getProperty("user.dir")));
int state = fc.showOpenDialog(null);
if (state == JFileChooser.APPROVE_OPTION) {
    file = fc.getSelectedFile();
} else {
    throw new FileNotFoundException("Auswahl abgebrochen");
}
```

Aufgabe 4: Fügen Sie der Klasse die folgende Methode hinzu:

```
public void entferneKommentare()
//Entfernt alle Kommentare im Code
```

Diese Methode entfernt aus dem Code in der `ArrayList` alle Kommentare. Beschränken Sie sich zur Vereinfachung auf die folgenden drei Fälle

1. Ein Zeilenkommentar beginnt mitten in einer Zeile. → Der Zeilenkommentar wird abgeschnitten. Beispiel:

```
int a; //Sinnlose Variable
```

2. Ein Zeilenkommentar beginnt in Spalte 1. → Die ganze Zeile wird gelöscht. Beispiel:

```
// Weiterer sinnloser Text
```

3. Ein Blockkommentar geht über mehrere komplette Zeilen. → Alle Zeilen werden gelöscht. Beispiel:

```
/*
Dies ist eine sinnlose Klasse mit Kommentar
*/
```

Berücksichtigen Sie nicht den Fall, dass ein Blockkommentar mitten in einer Zeile beginnt oder endet. Entfernen Sie in der Klasse `KommentarTest` alle Kommentare.

Zusatzaufgabe 5:

Berücksichtigen Sie auch den Fall, dass ein Blockkommentar mitten in einer Zeile anfängt oder aufhört. Das umfasst auch folgende Sonderfälle:

```
/*
* 1. Blockkommentar
*/ int i=5; //Zeilenkommentar
```

```
/*
* 1. Blockkommentar
*/ int i=5; /*
* 2. Blockkommentar
*/
```