

Návod na používanie nového logovania

Vypracovali:

- Bc. Lukáš Rešutík

Logovanie do súboru

Trieda *JimLocalFileCreator* umožňuje vytvorenie logovania do vlastného súboru. Pokiaľ je použitý konštruktor bez argumentov, vytvorí sa defaultne súbor **JavaLocalLogs.txt**. Metóda *writeLog(String text)* zapíše text do jedného riadku logovacieho súboru.

```
public void createLogFile() {  
    String string1= "Prvy riadok";  
    String string2= "Druhy riadok";  
    JimLocalFileCreator file = new JimLocalFileCreator();  
    file.writeLog(string1);  
    file.writeLog(string2);  
}
```

Obr. 1: Jednoduché písanie do súboru

Logovanie do súboru s vlastným menom

Pokiaľ chceme logovať do súboru s vlastným názvom trieda *JimLocalFileCreator* ponúka konštruktor s parametrom názvu súboru. Tento konštruktor vytvorí súbor so zadaným menom. Následne logovanie umožňuje metóda *writeLog(String názovSúboru, String text)*.

```
public void createLogFile() {  
    String string1= "Prvy riadok";  
    String string2= "Druhy riadok";  
    JimLocalFileCreator file = new JimLocalFileCreator("MojLogSubor.txt");  
    file.writeLog("MojLogSubor.txt",string1);  
    file.writeLog("MojLogSubor.txt",string2);  
}
```

Obr. 2: Zapisovanie do súboru s vlastným menom

Logovanie do súboru s vlastným menom aj dátumom vytvorenia záznamu

Trieda *JimLocalFileCreator* umožňuje zapisovanie do logovacie súboru s aktuálnym časom (hhmmss). Táto funkcionality je prístupná metódou *writeLog(String názovSúboru, String text, boolean date)*. Treba poslať ako argument *date true* aby logovalo s dátumom.

```

public void createLogFile() {
    String string1= "Prvy riadok";
    String string2= "Druhy riadok";
    JimLocalFileCreator file = new JimLocalFileCreator("MojLogSubor.txt");
    file.writeLog("MojLogSubor.txt",string1,true);
    file.writeLog("MojLogSubor.txt",string2,true);
}

```

Obr. 3: Logovanie do súboru s vlastným menom aj dátumom vytvorenia záznamu

Logovanie do csv súboru

Trieda *JimLocalCsvFileCreator* poskytuje možnosť vytvorenia vlastného csv súboru. Konštruktor triedy s parametrom mena súboru vytvorí súbor do ktorého môžeme zapisovať metódou `writeCsvLo(String menoSúboru, String...stlpceCsv)`. Konštruktor triedy taktiež umožňuje nastaviť separátor a ohraničnia dát v csv súbore.

```

public void createCsvFile() {
    Integer cislo1 = 10, cislo2 =15;
    String string1 = "druhy stlpec", string2 ="stvrty stlpec";
    JimLocalCsvFileCreator csv = new JimLocalCsvFileCreator("TestovacieCsv5");
    csv.writeCsvLog("TestovacieCsv5.csv", cislo1.toString(),string1,cislo2.toString(),string2);
    csv.writeCsvLog("TestovacieCsv5", new String[] {"10", "druhe", "5", "32"});
}

```

Obr. 4: Logovanie do csv súboru

Logovanie do vlastného csv súboru aj s dátumom

Pokiaľ chceme pridať do csv s dátumom zápisu trieda *JimLocalCsvFileCreator* vlastní metódu `writeCsvLog(String menoSúboru, boolean Date, String...data)`. Použitím `true` ako argumentu boolean date sa csv súbor vytvorí spolu s dátumom zapísania.

```

public void createCsvFile() {
    Integer cislo1 = 10, cislo2 =15;
    String string1 = "druhy stlpec", string2 ="stvrty stlpec";
    JimLocalCsvFileCreator csv = new JimLocalCsvFileCreator("TestovacieCsv5");
    csv.writeCsvLog("TestovacieCsv5.csv",true, cislo1.toString(),string1,cislo2.toString(),string2);
    csv.writeCsvLog("TestovacieCsv5",true, new String[] {"10", "druhe", "5", "32"});
}

```

Obr. 5: Logovanie do vlastného csv súboru aj s dátumom