Web Server Logs

Quiz, 10 questions

1 point

1. Suppose a web log is modified to now have a sixth piece of information, a priority, that can be represented as a String.

Which one of the following is the least likely change to the LogEntry class to accommodate this new part of a web log?

- A new **getPriority** method is created to return the priority.
- The **toString** method is modified to include a String parameter.
- A new String parameter is added to the constructor.
- The **toString** method is modified to include the new priority as part of the return String.
- A new private variable named **priority** is created.
- A new String field is initialized in the constructor.

1 point

2. Consider the following code for the **readFile** method of the **LogAnalyzer** class.

```
public void readFile(String filename) {
    FileResource fr = new FileResource(filename);
    for (String line : fr.lines()) {
        LogEntry le = WebLogParser.parseEntry(line);
    }
}
```

In the **Tester** class, **readFile** is called with a correct filename, and then **printAll** is called, but nothing is printed.

Which one of the following is likely the best reason why?

- In **readFile**, the wrong parameter is sent to **parseEntry**.
- In **readFile**, a **System.out.println** statement is missing that should be right after the for loop.
- In **readFile**, a **System.out.println** statement is missing from the body of the for loop.
- In **readFile**, the log entries were not stored in **records**.
- In **readFile**, a **System.out.println** statement is missing that should be right before the for loop.

1 point

3. Consider the following code for the method **printAllHigherThanNum** with one integer parameter **num**. This method should print all the logs that have a status code higher than **num**.

Which one of the following would be the best choice for suitable code for this method?

```
if (le.getStatusCode() > num) {
    for (LogEntry le : records) {
        if (le.getStatusCode() > num) {
            System.out.println(le);
        }
    }
}
```

```
for (LogEntry le : records) {
    if (le.getStatusCode() > num) {
        System.out.println(le);
    }
    else {
        System.out.println();
    }
}
```

```
for (LogEntry le : records) {
    if (le.getStatusCode() > num) {
        System.out.println(le);
    }
}
```

```
if (le.getStatusCode() > num) {
    for (LogEntry le : records) {
        System.out.println(le);
    }
}
```

```
if (le.getStatusCode() > num) {
    for (LogEntry le : records) {
        System.out.println(le);
    }
}
else {
    System.out.println();
}
```

1 point

4. Run the method **countUniqueIPs** on the file **weblog2_log**.

How many unique IP addresses are in the file?

45

1 point

5. Run the method **uniqueIPVisitsOnDay("Sep 24")** on the file **weblog2_log**.

What size is the ArrayList that is returned?

14

1 point

6. Run the method **countUniqueIPsInRange(400,499)** on the file **weblog2_log**.

What number is returned?

23

1 point

7. Run the method **mostNumberVisitsByIP** after a HashMap has been created from the method **countVisitsPerIP** on the file **weblog2_log**.

What number is returned?

63

1 point

8. Run the method **iPsMostVisits** after a HashMap has been created from the method **countVisitsPerIP** on the file **weblog2_log**.

What single IP address is returned in the ArrayList?

- 103.57.41.178
- 188.162.84.63
- 200.69.213.251
- 210.4.104.99
- 212.128.74.248

1 point

9. Run the method **dayWithMostIPVisits** with a HashMap has been created from the method **iPsForDays** on the file **weblog2_log**.

What day is returned?

- Sep 24
- Sep 26
- Sep 28
- Sep 30

1 point

10. Run the method **iPsWithMostVisitsOnDay** with two parameters—one, a HashMap that has been created from the method **iPsForDays** on the file **weblog2_log** and two, the day "Sep 30".

Two IP addresses are returned in the ArrayList—which are they?

- [ ] 66.67.61.44
- [ ] 103.57.41.178
- [ ] 106.220.155.36
- [ ] 188.162.84.63
- [ ] 210.4.104.99
- [x] 212.128.74.248

I, **Ning Zheng**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Learn more about Coursera's Honor Code

Submit Quiz