Research the Internet for features added to Java 9, 10, 11, 12, 13, 14, 15, 16, or 17. Select one from 3 different Java versions. Write a summary on three features for one of the versions you have selected and provide an example you may have found or one you have written. Cite your sources.

**Assignment Requirements and Grading:**

Throughout the years, Java has made many improvements and additions to its features. Today, Java’s latest version is 23, but earlier versions are still available for download. For this post, I explored the differences and similarities between Java 9, 10, 11, 12, 13, 14, 15, 16, and 17. In Java 10, a new update was incorporated to make code easier to read (Aneraye, 2022). Many updates to Java focus on reducing the boilerplate code, and Java 10 was no exception. In Java 15, text blocks were fully integrated within Java (Aneraye, 2022). Now, with three double quotes, a text block can be created, improving readability and increasing format options.

Methods had long existed in Java before, but in Java 11, more methods were added to the String class, including “isBlank, lines, strip, stripLeading, stripTrailing, and repeat” (Aneraye, 2022). If you want the exact string to output more than once, then the repeat method will achieve this using the repeat(n) method. If a string has unwanted extra space, the strip() function will remove it. Within Java 11, there were also new features to the file methods, which helped to make writing and reading files easier (Aneraye, 2022). The isBlank() method results in either true or false, depending on whether the conditions in the string or strings are met. Vijay Aneraye includes an example of the isBlank():

Example  
class Example {   
String str1 = "";  
System.out.println(str1.isBlank());  
}  
output: True

If characters, variables, or names were typed in, the output would change to false since the String would no longer return blank.

**References**

Vijay Aneraye. (2022, February 5). *Important Features — Java 9, Java 10, Java 11, Java 12, Java 13, Java 14, Java 15, Java 16, Java 17*. Medium. https://medium.com/@vijayaneraye/important-features-java-9-java-10-java-11-java-12-java13-java-14-java-15-java-16-java-17-650420ee7337

* An initial post of approximately 250 words is due by **Thursday, 11:59 p.m., CST**.
* Submit your post by clicking on the assignment link above, then Create Thread. You must create a thread in order to view your peers' posts. Tip: Create your post in a Word document and then copy and paste your work into the thread.
* A minimum of three (3) responses, to the original threads of other students, of 100-200 words each are due by **Sunday, 11:59 p.m. CST**.
* To view the rubric grading criteria, click the following link: [Discussion Board Grading Rubric](https://content.bellevue.edu/cst/csd/rubricdbv3.pdf).

Anton, you did a really nice job on your post for this module! I was intrigued to see how far Java has come through all its updates. I cannot begin to imagine how much more difficult Java would be to learn if we did not have all the latest improvements. I like how you utilized a reliable source, like Oracle, to document the major changes. That is a fantastic resource to use when needing accurate information on Java now and throughout the years.

Was there a particular feature that stuck out to you more than others throughout the different versions?

Miles, I enjoyed reading your discussion post elaborating on Java 13, 16, & 17. It must have been even more interesting treading all the version updates from 8 to 21! It is crazy to think that there is already a version 23 available for use. It seems like Java 16 migrating Git from Mercurial was a huge upgrade and aids how we use that in our classes today! I wonder if any of the more recent updates further expanded on GitHub properties.

Were there any additions or removals mentioned that you do not understand or think are for the worst?

Colton, you did a nice job walking through the new Java 11, 15, and 17 features. I also chose to focus on Java 11 and 15. I am grateful that they included text blocks as a new feature because I recently used them in a program and would have been confused trying to read long lines of code, so not having to scroll was a nice addition. Out of all the features you mentioned, which do you like the most? It is difficult for me to choose because so many are helpful. I think having Single-file source-code programs is a gamechanger.