**AP Computer Science I**

Performance Task

# **Create — Consumer Review**

## **Overview**

The Consumer Review Lab provides students with the opportunity to work on a larger

assignment involving multiple classes early in the course, before they necessarily

know how to write their own classes. By focusing on calling methods and using the

control structures of selection and iteration, students will have practice that directly

relates to the “Methods and Control Structures” free-response question type that

they will see it on the end-of-year exam. Because students often struggle with methods

in the String class, this lab specifically focuses on calling these methods and

building strings in an engaging and interesting way.

We all use reviews on a regular basis, to help us determine which movie to see, which

video game to play, or even which pair of headphones to buy. But how easy is it to fake

a review? The truth is that it‘s incredibly simple, and fake reviews can have significant

impacts on the products that are purchased or consumed, including movies that

people watch and restaurants where they choose to eat. If consumers knew that the

review they were reading and using to base a decision on came from a computer

program and not a human, they might be less likely to believe it.

What motivation do people have for creating these fake reviews or comments, and

What should be done to prevent it, if anything? If technology can be used to create

and post positive reviews for the company‘s own products or negative reviews of the

competition, what are the ethical implications of this?

Creating fake reviews has ethical implications, and it is important to emphasize that

although students are going through the process to see one way in which they can

be generated, this is in no way advocating this type of behavior. This is similar to

“white hat” hackers needing to understand the same tools and processes of criminal

hackers in order to do their job. Discuss with students the ethical concerns and how

Understanding this process is important to preventing unwanted behaviors online.

## **Assessment**

You will be provided with 12 hours of class time to complete and submit the following:

* A video of your program running
* Written responses about your program and development process
* Program Code

Your teacher will share submission guidelines that include suggestions for creating video and PDF files.

## **General Requirements**

You are required to:

* Iteratively design, implement, and test your program.
* Independently create at least one significant part of your program.
* Create a video that displays the running of your program and demonstrates its functionality.
* Write responses to questions about your program.
* Include your entire program code.

## **Program Requirements**

Your program must demonstrate:

* Students are able to create objects of type String, especially using string literals, but have difficulty writing program code to access information about the string using the available methods.
* Students are able to write single branching program code but have difficulty writing programs that require more than two options or pathways through the program.
* Students can write program code to call methods on a String object but often use incorrect parameters when calling the methods.
* Students can write program code to call methods on a String object using the correct parameters but have difficulty seeing how to put calls to multiple methods together in a solution

## **Submission Requirements**

### 1. **Video**

Submit one video in .mp4, .wmv, .avi, or .mov format that demonstrates the running of at least one significant feature of your program. Your video must not exceed 1 minute in length and must not exceed 30MB in size.

### 2**. Written Responses**

Submit one PDF document in which you respond directly to each prompt. Clearly label your responses 2a – 2e in order. Your response to all prompts combined must not exceed 750 words, exclusive of the Program Code.

## **Program Purpose and Development**

1. Provide a written response or audio narration in your video that:

Identifies the programming language.

* Identifies the purpose of your program.
* Explains what the video illustrates.

(Approximately 150 words)

1. Why did you choose to implement this program? Describe the development process used in the completion of the project.

(Approximately 200 words)

1. Provide the method header for a method that you implemented that takes at least one parameter. Explain why you chose the given parameter(s), including type, and why you made the method static or non-static. Provide the code segment(s) where two distinct methods in the String class are called. Describe what each method call is doing, and what is being returned (if anything) by the method calls

(Approximately 200 words)

1. Copy and paste one code segment that uses nested conditional statements or compound Boolean expressions. What is one other way that this code could be written to achieve the same result? Provide an equivalent code segment to the one included above. Copy and paste one code segment that uses iteration. Describe how the loop you used works and provide an equivalent code segment to the one included above that uses a different type of loop.

(Approximately 200 words)

1. Capture and paste your entire program code into the PDF.

* Include comments or citations for program code that has been written by someone else.

## Tasks

### **Activity 1 - Explore**

**Description**

This activity introduces students to the sentiment value of words. A word can have

a positive or negative connotation based on the context with which it is often used.

Before students start working with the Review class, it is helpful for them to read a

a few reviews.

Have students read two to three online reviews of their choice. Some options include music reviews, game reviews, movie reviews, or restaurant reviews, although if a student has a specific interest not listed here, consider allowing them to pursue it. If students are unsure of where to find sample reviews, spend time as a class brainstorming sites that contain different types of reviews that students could use.

* Review and Sentiment Values

Time To Complete: 1 Hours

### **Activity 2 - Research**

**Description**

In this lesson, students will be introduced to and practice using the Java main skeleton that includes the class and main method arguments. They will use the System.out.print and System.out.println commands to print string literals in the editor. Students will learn about Boolean expressions and relational operators. Relational Operators are constructs in programming that allow for the comparison of the values of two expressions.

* Unit 1
* Unit 2

Time To Complete: 18 Hours

### **Activity 3 - Ideate**

**Description**

Have students work with a partner to discuss the algorithm for determining the

total sentiment of a review (a string with multiple words, each of which will have a

sentiment value).

When calling the two-parameter substring method, the first parameter is the starting index of the returned substring and will be included in the returned string object, while the second parameter is the first index not included in the returned string. This information is also included in the Java Quick Reference, and it is useful for students to get accustomed to using it during the year.

If students have trouble determining the algorithm for totalSentiment or starValue, walk them through the provided pseudocode in the activity solutions either in pairs or as an entire class.

* Unit 3
* Unit 4
* Sentiment Value and Star Rating

Time To Complete: 12-14 Hours

### **Activity 4 - Evaluate**

**Description**

Students will evaluate each other's star review program and provide feedback to one another. Once done students will then play out the next part of the project which is auto generated a review.

Time To Complete: 1-2 Hours

### **Activity 5 - Construct a Prototype**

**Description**

In this activity students will write and test a method that will auto generate a review

from the review used in Activity 2. The method will create a new fake review by

replacing adjectives in the given review with randomly selected adjectives.

As a class, begin to generate lists of positive and negative adjectives that appear in

a single review. Students can look back on the reviews they read for the first activity, or can find new reviews to look at to help generate adjectives. Keep track of the list generated by the class, and then have students complete the list alone or with a partner. Note that because sentiment is subjective, it is not necessarily the case that words chosen as negative adjectives will have a negative sentiment value.

The same is true for words that are chosen as positive adjectives.

To complete this activity, students will need a review to work with. Students can use the same review that they used in the previous activity, such as SimpleReview.txt. In addition, they will be calling the same method used in Activity 2 to read in a text file and return the contents of the file as a single long string.

Students can technically annotate adjectives with whatever type of symbol or markup they want. The “\*” is used as a matter of keeping things simple for both teachers and students, but there is no difference between this and any other symbol or string. It’s important not to choose anything that might come naturally at the beginning of a word, since we want a clear way to differentiate an adjective from other types of words. This is why choosing something like “a”, or any other letter, would not be a good idea.

* Auto generate Review

Time To Complete: 2-3 Hours

### **Activity 6 - Improve the Design**

**Description**

In this activity, students will modify the method they completed in the previous

activity to create a review that is either more positive or more negative than the

original. Before students answer questions related to auto generating reviews, it might be helpful to discuss the ethical implications of auto generated reviews again. One possible question for discussion is what responsibility or liability a programmer would have if the code they wrote were used to autogenerate reviews?

* Positive or negative review

Time To Complete: 3-5 Hours

### **Activity 7 - Share Solutions**

**Description**

As a class, spend a few minutes reviewing the requirements of the open-ended activity.

The goal of the activity is to allow students to demonstrate their knowledge of

methods and control structures in a way that is interesting and engaging to them.

While it’s possible to create additional constraints or requirements, it’s best to

provide students with as much freedom as possible. The use of the Review class

is intentionally not included in the list of requirements; however, it’s possible that

students will continue to work with the given classes from this lab.

* Open Ended Activity - Share

Time To Complete:1-3 Hours

### **Activity 8 - Reflect**

**Description**

Students should complete the reflection and hand in all files.

Time To Complete: 1 Hour