

# Practice Exam

## Background

You are working as a software development consultant. One of your clients is the popular restaurant Salads to Go (STG). Salads to Go (STG) is an upscale, fast-casual restaurant that specializes in “build your own salads”. STG has requested that you develop a Java-based application to help STG customers order salads. STG customers choose a type of salad greens, such as kale, romaine, or iceberg lettuce. Then they specify a protein for the salad. Finally, the customers choose from a range of toppings, such as cheese, croutons, tomatoes, onions, bacon, and olives.

## Instructions

1. Use the Eclipse to create a new Java application that meets the requirements listed on the next page.
2. Test your application and make sure it functions correctly.



# Practice Exam

## Application Requirements

1. Your application will have two classes – an OrderingSystem class with a main method and a Salad class that will serve as a blueprint to create a new salad for the customer.
2. Create your Salad class. Add attributes to hold the type of greens, the type of protein, and the toppings the customer specifies. Add an attribute called isVegan to track if the salad is vegan or not. Add a constructor and the accessor methods (getters and setters.)
3. Create an Ordering system class with a main method. Add the code for the remaining requirements to the main method.
4. Declare a variable to hold the STG greeting: “Welcome to Salads to Go!”
5. Declare a variable to hold the STG slogan: “Build Your Own Salads, Your Way. Time to build a salad.”
6. Declare a Boolean variable called “isVegan”. (We will use this variable to track whether the user’s salad order is vegan or not.)
7. Declare three ArrayLists to hold the types of greens, types of proteins, and types of toppings. Populate the lists with the values below.
  - a. Greens types: kale, romaine lettuce, iceberg lettuce
  - b. Proteins: tuna, chicken, or black beans
  - c. Toppings: cheese, croutons, tomatoes, onions, bacon, and olives
8. Add a method to the application that displays the greens types, proteins, and toppings to the customer. Within the method, use a loop to iterate over the items in the list and display them to the customer.
9. Display the greeting and the slogan to the customer.
10. Call the method to display the greens types to the customer.
11. Then prompt the customer to choose a type of salad greens by asking “What type of salad greens do you want?”
12. Create a new salad object and with the type of greens the customer has chosen.
13. Call the method to display the proteins to the customer.
14. Prompt the customer to choose a type of protein by asking “What type of protein do you want?”
15. Set the type of protein that the customer chose on the salad object.
16. Use a decision statement to determine if the protein the customer chose is vegan. Tuna and chicken are not vegan. The black beans are vegan.
17. If the protein is not vegan, set the isVegan property of the salad to false. If the protein is vegan, set the isVegan property of the salad to true.

# Practice Exam

18. Create a new empty ArrayList called "toppings". This list will hold the toppings for the salad that are chosen by the customer.
19. Use a loop to keep prompting the customer to select another topping until the customer indicates they do not want any more toppings.
  - a. Inside the loop body, prompt the customer to choose a topping by asking "What type of topping do you want?" Then call the method to display the topping types to the customer.
  - b. Inside the loop body, add the topping to the toppings list.
  - c. Inside the loop body, use a decision statement to determine if the topping the customer chose is vegan. Cheese and bacon are not vegan. Croutons, tomatoes, olives, and onions are vegan. If the topping is not vegan, set the isVegan property of the salad to false. If the topping is vegan, set the isVegan property of the salad to true.
20. Add the list of toppings to the salad object.
21. Display a message to the customer that says, "You ordered a [protein type] salad on [greens type]. Use string concatenation to fill in the greens type and protein type that the customer chose.
22. If the user chose at least one topping, display a message that says "Your salad is topped with: ". Then loop through the toppings on the salad and display them.
23. If the user ordered a vegan salad, display a message saying, "Thank you for trying our vegan salad." Otherwise, display a message thanking the customer for ordering a salad from Salads to Go.
24. Open a file called SaladOrder.txt and write the information about the salad to the file, including the greens type, protein type, and toppings.
25. Add exception handling to catch any file I/O exceptions.

## Optional Challenge

You are showing off the new application to Filsan Hassan, the IT Manager at Salads to Go (STG). Ms. Hassan asks why the customer isn't able to order salad dressing with their salad. You realize you forgot to add salad dressing as an option! Refactor the Salad class so that it contains an attribute for dressing. Then refactor your main method so the customer is prompted to choose a type of dressing and set the dressing value on your Salad object. You will also need to display the dressing to the user and write it to the file with the rest of the order.