1. What is the purpose of the Integer, Double, and Character Wrapper classes?
2. What happens when you declare ArrayList without the type parameter?
3. What is the purpose of the constructor?
4. Why are public fields frowned upon?
5. Are constructors static?
6. When does the compiler automatically generate the no-arguments constructor?
7. How do you resize an array?
8. Write a code fragment which would iterate through a two-dimensional array "backwards" (right-to left, row-by row, starting at the bottom).
9. What method must be implemented if a class were to implements the Comparable interface?
10. When would you want to use an array over an ArrayList?
11. What is the difference between an unchecked exception and a checked exception?
12. When will a PrintWriter throw a FileNotFoundException
13. What is the difference between an accessor and a mutator?
14. When is casting between classes necessary?
15. Where can the "this" keyword be used?
16. What is the "canonical" way to write a for loop that should run 10 times?
17. What does instanceof do?
18. What is DeMorgan's Law?
19. Name two methods that are automatically inherited from the Object superclass. What do these methods do by default?
20. Where can final fields be initialized?
21. Take the string "There are 5 bears in the cave." What call to substring() would yield the string "bears"?
22. Can you have a catch statement with no try? (http://www.youtube.com/watch?v=BQ4yd2W50No)
23. What is the difference between a static method and an instance method?
24. What does "." mean in the context of a path? How about ".."?
25. I define a new exception class which inherits from FileNotFoundException. Will it be checked or unchecked?
26. What is a shallow copy?
27. Describe the concept of input streams.
28. Who has access to private members?
29. Suppose I have public static void main() within an instantiable class. Will it have access to the class's instance methods?
30. Can you construct an interface object?
31. What is the significance of a Random object's seed value?
32. What is "short-circuit evaluation" or "lazy evaluation"?
33. What is the difference between an absolute path and a relative path?
34. Will a finally statement execute when a thrown exception isn't handled by any of the catch blocks?
35. What is the purpose of String[] args in the header for public static void main(String[] args)?
36. What does System.exit(1) do? Why is using it considered bad practice?

37. Describe overloading.
38. What is a shared reference?
39. How do you import all of the classes from a given package?
40. What is the purpose of break; statements inside of switch statements?
41. Will a finally statement execute when return is called inside a try block?
42. Describe encapsulation
43. What do escape sequences do?
44. What is linear search?
45. What's the difference between a reference variable and a primitive variable?
46. What is the syntax of a do-while loop? When would you want to use one?
47. Describe auto-boxing.
48. When does a for loop counter go out of scope?
49. What default values are given to instance fields if they are not initialized? Is it different for local variables?
50. Suppose I give you an ArrayList of Integers. Write an code fragment that removes every odd number. (Not every odd-indexed number)
51. When do you need to add a throws declaration onto the end of a method header?
52. What are the differences between next() and nextLine()
53. When will an InputMismatchException get thrown?
54. What is garbage collection?
55. Why do we use import statements? Do we absolutely need them?
56. What happens when I try to assign the value from a post-decrement operation?
57. Can interfaces have instance fields?
58. Can constructors call other constructors of the same object?
59. If main were to pass an int to some method foo, and foo modifies that int, would that value change from main's perspective?
60. Can instance methods refer to static fields?
61. What happens when you scan a string?