1. Какие есть типы данных в Java?
2. Чем отличаются объектные типы от примитивных типов данных?
3. В чем разница передачи параметров по ссылке и по значению?
4. Что такое OutOfMemoryError?
5. **Что такое стектрейс? Как его получить?**
6. Назовите все методы класса Object.
7. Что такое конструкторы? Какие типы знаете?
8. ***Объекты каких стандартных классов immutable в Java?***
9. ***Дайте краткую характеристику immutable object. Зачем они нужны? Как их создать? Какие преимущества immutable object перед обычными объектами?***
10. Что такое ООП? Назовите принципы с примерами.
11. Расскажите о принципе наследования в ООП. Зачем он нужен?
12. Дайте определение принципа полиморфизма в ООП. Как работает полиморфизм?
13. Дайте определение принципа абстракции в ООП.
14. Какие элементы языка отвечают за инкапсуляцию?
15. ***Какие элементы языка отвечают за наследование? Какие особенности использования ключевого слова super?***
16. Какие элементы языка отвечают за полиморфизм?
17. Что такое перегрузки (overloading) метода?
18. Что такое переопределение (override) метода?
19. Что такое класс, объект, интерфейс?
20. Какие элементы могут содержать класс?
21. ***Что такое сигнатура метода?***
22. ***Можно ли в конструкторе применять return?***
23. Можно ли в конструкторе выкинуть исключение (exception)?
24. Когда применяется ключевое слово this?
25. Что такое модификаторы доступа в Java? Назовите их. Для чего используются?
26. Назовите основную особенность статических переменных и методов. Какие основные ограничения действуют на статические переменные и методы?
27. ***Что означает ключевое слово static? Может ли статический метод быть переопределенным или перегруженным?***
28. Можно ли использовать статические методы внутри обычных? Наоборот? Почему?
29. Что означает ключевое слово final?
30. ***Что такое abstract? Абстрактный класс? Абстрактный метод?***
31. ***Что такое interface? Может ли быть final interface?***
32. ***В чем разница между абстрактным классом и интерфейсом в Java?***
33. Какие модификаторы доступа могут быть у класса?
34. ***Что такое класс Assert? Зачем и как его использовать?***
35. Расскажите про String в Java.
36. Какие есть способы создания объекта String? Где он создается?
37. Как сравнить две строки в Java и/или отсортировать их?
38. ***Что такое строковый пул и зачем он нужен?***
39. Какая разница между String, StringBuffer и StringBuilder?
40. Дайте краткую характеристику Enum в Java.
41. ***Может ли Enum реализовывать (implement) интерфейс?***
42. ***Может ли Enum расширить (extends) класс?***
43. ***Можно ли создать Enum без экземпляров объектов?***
44. ***Можем ли мы переопределить метод toString() для Enum?***
45. ***Что будет, если не переопределять метод toString() для Enum?***
46. Можем ли мы указать конструктор внутри Enum?
47. В чем разница между == и equals()?
48. Что делает метод ordinal() в Enum?
49. ***Можно ли использовать Enum с TreeSet или TreeMap в Java?***
50. ***Как связаны методы ordinal() и compareTo() в Enum?***
51. Можно ли использовать Enum в switch case?
52. Как получить все имеющиеся значения в экземпляре Enum?
53. ***Что такое функциональный интерфейс?***
54. ***Какие функциональные интерфейсы входят в стандартную библиотеку Java?***
55. Что такое лямбда-выражение?
56. Что такое ссылка на метод (::)? Чем это отличается от лямбда-выражения?
57. Что такое Stream API в Java?
58. Как создать Stream из коллекции в Java?
59. ***Что такое промежуточные операции в Stream API? Какие примеры промежуточных операций в Stream API вы знаете?***
60. ***Что такое терминальные операции в Stream API? Какие примеры терминальных операций в Stream API вы знаете?***
61. Что такое параллельные стримы?
62. ***Какие методы можно использовать для группировки элементов Stream?***
63. ***Какой параметр принимает в себя метод filter?***
64. ***Какая иерархия коллекций в Java Collection Framework?***
65. ***Какое внутреннее строение ArrayList?***
66. ***Какое внутреннее строение LinkedList?***
67. ***Какое внутреннее строение HashMap?***
68. Чем отличаются ArrayList и LinkedList?
69. Чем отличаются ArrayList и HashSet?
70. ***Зачем в Java такое многообразие имплементации динамического массива?***
71. ***Зачем в Java такое многообразие имплементации key-value storage?***
72. Как отсортировать коллекцию элементов?
73. В чем преимущество массива перед ArrayList?
74. Дайте краткую характеристику class object в Java.
75. Для чего используют Equals and HashCode в Java?
76. Расскажите о контракте между Equals and HashCode в Java.
77. ***Какие условия выдвигаются касательно переопределения соглашения при переопределении Equals?***
78. Что будет, если не переопределить Equals and HashCode?
79. Какие значения мы получим, если у нас не перераспределены Equals and HashCode?
80. ***Почему симметричность выполняется, только если x.equals(y) возвращает true?***
81. Что такое коллизия в HashCode? Как с ней бороться?
82. ***Что будет, если элемент, участвующий в контракте с HashCode, меняет свое значение?***
83. ***В чем разница применения if (obj instanceof Student) и if (getClass() == obj.getClass())?***
84. Дайте краткую характеристику метода clone().
85. Дайте определение понятию exception (исключительная ситуация).
86. Какие знаете особенности использования оператора try ... catch?
87. ***В чем разница между error и exception?***
88. ***Какая разница между checked и unchecked, exception, throw, throws?***
89. ***Какая есть иерархия исключений?***
90. Что такое checked и unchecked exception?
91. Нужно ли проверять checked exception?
92. О чем говорит ключевое слово throws и как его использовать?
93. ***Какие возможные способы обработки исключений вы знаете?***
94. Какие есть правила для проверки исключений при наследовании?
95. Какой оператор позволяет принудительно выбросить исключение? Напишите пример.
96. ***Может ли метод main выбросить throws-исключение? Если да, то куда передаст?***
97. ***Как добиться StackOverflowError?***
98. В каких случаях возникает OutOfMemoryError?
99. ***Зачем нужен Garbage Collector?***
100. Что такое Heap? ***Что такое Stack?***
101. Что такое пул строк (string pool) и как он работает в Java?
102. ***Что такое процесс и поток?***
103. Что такое многопоточность и зачем она используется в Java?
104. Как создать и запустить новый поток в Java?
105. ***Какие способы синхронизации доступа к общим ресурсам в Java вы знаете?***
106. Какие проблемы могут возникнуть при работе с многопоточностью в Java?
107. ***Что такое состояние гонки (race condition) и как ее можно предотвратить в Java?***
108. ***Что такое дэдлок (deadlock) и как его можно избежать в Java?***
109. ***Какие классы и интерфейсы в Java используются для работы с многопоточностью?***
110. ***Какие методы и классы в Java используются для синхронизации потоков?***
111. ***Что такое пул потоков (thread pool) и как он используется в Java?***
112. ***Что такое ForkJoinPool?***
113. ***Что такое агрегатные операторы?***
114. Что такое нормальные формы? Какие нормальные формы вы знаете?
115. Что такое индексы? Какие индексы вы знаете? ***Какая структура данных лежит в их основе?***
116. ***В чем отличие WHERE от HAVING?***
117. Что такое JOIN? Какие типы JOINs вы знаете?
118. Какие основные типы данных в SQL вы знаете?
119. Что такое первичный ключ?
120. ***Что такое Spring Framework и какие преимущества он предоставляет?***
121. ***Какие модули входят в Spring Framework?***
122. ***Что такое инверсия контроля (IoC) и внедрение зависимостей (DI)? Как они реализуются в Java?***
123. ***Что такое спринг бин? Как создать спринг бин?***
124. ***Опишите жизненный цикл бина.***
125. ***Что такое скоупы бина? Какие скоупы вы знаете?***
126. ***Зачем нужны аннотации @Qualifier и @Primary?***
127. ***Какие проблемы могут возникнуть при использовании Spring и как их можно решить?***
128. ***Что такое Spring Boot и какие преимущества он предоставляет?***
129. ***Какая разница между Spring и Spring Boot?***
130. ***Какая разница между аннотациями @Service, @Component и @Repository?***
131. ***Какие аннотации используются в Spring MVC и для чего они предназначены?***
132. Зачем нужна аннотация @Value?
133. ***Что такое Spring Boot Starter и как он используется?***
134. ***Какие средства предоставляет Spring Boot для управления конфигурацией приложения?***
135. Какие инструменты предоставляет Spring Boot для разработки и тестирования приложений?
136. ***Что такое репозиторий в Spring Data и как он используется?***
137. ***Как подключиться к базе данных с помощью Spring Data?***
138. Какие требования предъявляются энтити-классу?
139. Зачем нужна аннотация @Query?
140. ***Что такое фетч-стратегия? Какие они бывают?***
141. ***Что такое проблема N+1?***