**Дженерики**

1. Что такое дженерики?
2. Для чего нужны дженерики? Параметр vs Аргумент. (в дженериках)?
3. Типизированные методы (Generic Methods)
4. Типизированные классы (Generic Types)
5. Что такое сырые типы (raw type)?
6. Можно ли выбрасывать исключение generic-типа?
7. Дженерики в исключениях – что можно, а что нельзя?
8. Что такое вайлдкарды?
9. Расскажите про принцип PECS

**Коллекции**

1. Что такое «коллекция»?
2. Расскажите про иерархию коллекций
3. Почему Map — это не Collection, в то время как List и Set являются Collection?
4. В чем разница между классами java.util.Collection и java.util.Collections?
5. Какая разница между итераторами с fail-fast и fail-safe поведением? (с примерами) Каким будет результат выполнения данного кода?
6. Чем различаются Enumeration и Iterator?
7. Как избежать ConcurrentModificationException во время перебора коллекции?
8. Как между собой связаны Iterable, Iterator и «for-each»?
9. Можно ли, обходя ArrayList, удалить элемент? Какое вылетит исключение?
10. Как удалить элемент из ArrayList при итерации?
11. Как поведёт себя коллекция, если вызвать iterator.remove()?
12. Чем Set отличается от List?
13. Расскажите про интерфейс Set.
14. Расскажите про реализации интерфейса Set
15. Зачем нужны и чем отличаются интерфейсы Comparable и Comparator?
16. В чем отличия TreeSet и HashSet?
17. Чем LinkedHashSet отличается от HashSet?
18. Что будет, если добавлять элементы в TreeSet по возрастанию?
19. Что будет в этом случае?
20. Как устроен HashSet, сложность основных операций.
21. Как устроен LinkedHashSet, сложность основных операций.
22. Как устроен TreeSet, сложность основных операций.
23. Расскажите про интерфейс List
24. Как устроен ArrayList, сложность основных операций.
25. Как устроен LinkedList, сложность основных операций.
26. Почему LinkedList реализует и List, и Deque?
27. Чем отличаются ArrayList и LinkedList?
28. Что такое Queue?
29. Что такое Deque? Чем отличается от Queue?
30. Приведите пример реализации Deque.
31. Какая коллекция реализует FIFO?
32. Какая коллекция реализует LIFO?
33. Оцените количество памяти на хранение одного примитива типа byte в LinkedList?
34. Оцените количество памяти на хранение одного примитива типа byte в ArrayList?
35. Как устроена HashMap? (Расскажите про принцип корзин)
36. Что такое бинарное дерево?
37. Когда и как происходит перестроение мапы в дерево?
38. Каковы особенности красно-чёрного дерева?
39. Сложность основных операций HashMap?
40. Что такое LinkedHashMap? LinkedHashMap - что в нем от LinkedList, а что от HashMap?
41. Как устроена TreeMap, сложность основных операций?
42. Что такое WeakHashMap?
43. В чем разница между HashMap и WeakHashMap? Для чего используется WeakHashMap?
44. Как работает HashMap при попытке сохранить в него два элемента по ключам с одинаковым hashCode(), но для которых equals() == false?
45. Что будет, если мы кладем в HashMap ключ, у которого equals и hashCode определены некорректно?
46. Возможна ли ситуация, когда HashMap выродится в список даже с ключами имеющими разные hashCode()?
47. Почему нельзя использовать byte[] в качестве ключа в HashMap?
48. Будет ли работать HashMap, если все добавляемые ключи будут иметь одинаковый hashCode()?
49. Какое худшее время работы метода get(key) для ключа, которого НЕТ в HashMap?
50. Какое худшее время работы метода get(key) для ключа, который ЕСТЬ в HashMap?

**Функциональные интерфейсы**

1. Что такое функциональный интерфейс?
2. Для чего нужна аннотация @FunctionalInterface?
3. Какие встроенные функциональные интерфейсы вы знаете?
4. Все способы реализации функционального интерфейса?
5. Что такое ссылка на метод?
6. Что такое лямбда-выражение? Чем его можно заменить? Как можно и как нельзя?

**Stream API**

1. Что такое Stream API? Для чего нужны стримы?
2. Почему Stream называют ленивым?
3. Какие существуют способы создания стрима?
4. Как из коллекции создать стрим?
5. Какие промежуточные методы в стримах вы знаете?
6. Методы конвейерные
7. Методы терминальные
8. Методы числовых стримов
9. Расскажите про класс Collectors и его методы.
10. Расскажите о параллельной обработке в Java 8.
11. Что такое IntStream и DoubleStream?

**Java 8 и более старшие версии**

1. Какие нововведения появились в java 8?
2. Нововведения Java 17 (посмотреть видео по ссылке)
3. Какие новые классы для работы с датами появились в java 8?
4. Расскажите про класс Optional
5. Что такое Nashorn?
6. Что такое jjs?
7. Какой класс появился в Java 8 для кодирования/декодирования данных?
8. Как создать Base64 кодировщик и декодировщик?
9. Какие дополнительные методы для работы с ассоциативными массивами (maps) появились в Java 8?
10. Что такое LocalDateTime?
11. Что такое ZonedDateTime?