



Upgrade.td_Разработка на Java для школьников

- 1) Чем отличается от обычных массивов `arraylist`?
- 2) Существует ли в `ArrayList` метод `remove()`?
- 3) Существует ли метод `add()` в `LinkedList`?
- 4) Что такое `Comparable`?
- 5) Какой интерфейс стоит использовать для нестандартного сравнения в JAVA?
- 6) Что должно вернуть выражение `x.equals(null)` в соответствии с контрактом по написанию этого метода?
- 7) Какие классы коллекций позволяют изменять размер и предоставляют индексированный доступ к элементам, но чьи методы не синхронизированы?
- 8) Какое утверждение верно для класса `java.util.ArrayList`?
- 9) Если метод `equals` возвращает значение `true` для двух объектов, какие значения должен вернуть метод `hashCode()`?
- 10) Алгоритм, требует реализации `java.util.List`, которая обеспечивает эффективную реализацию `add(0, object)`, но НЕ нуждается в поддержке быстрого произвольного доступа. Что поддерживает эти требования?
- 11) Корректное ли это лямбда выражение? `(a1, int a2) -> a1 + a2`
- 12) Каков результат выполнения кода? `IntUnaryOperator u1 = i -> i / 6;`
`IntUnaryOperator u2 = i -> i + 12;`
`System.out.println(u1.compose(u2).applyAsInt(12));`
- 13) Корректное ли это лямбда выражение? `(int a) -> --a; return a;`
- 14) Каков возвращаемый тип если тело лямбда выражения состоит из множества выражений и ключевое слово `return` пропущено?
- 15) Сколько выражений может содержать в себе тело лямбда выражения?
- 16) Может ли класс из `Collection Framework` быть отсортированным и неупорядоченным?





- 17 Какой функциональный интерфейс лучше использовать вместо Function?
- 18 При автоматическом преобразовании примитивных типов выполняется преобразование с расширением или сужением?
- 19 Как аргументы разделяются в лямбда выражении?
- 20 Что такое функциональный интерфейс?

