Оглавление

[1. Что такое функциональные интерфейсы? 2](#_Toc40458547)

[2. Перечислите функциональные интерфейсы из пакета java.util.function. 2](#_Toc40458548)

[3. Что такое функции высшего порядка? 2](#_Toc40458549)

[4. Какие функциональные интерфейсы из пакета java.util.function поддерживают функции высшего порядка? 3](#_Toc40458550)

[5. Что такое ссылки на методы? 3](#_Toc40458551)

[6. Что такое ссылки на конструкции? 3](#_Toc40458552)

[7. Расскажите о зоне видимости переменных в lamdba - выражениях? 3](#_Toc40458553)

[8. Как быть в ситуации, если внутри lamdba - выражении операторы могут выкинуть исключение? 4](#_Toc40458554)

[9. Что такое Stream API? 4](#_Toc40458555)

[10. Расскажите, какой шаблон проектирования используется внутри Stream API? 4](#_Toc40458556)

[11. Что делает метод filter? 4](#_Toc40458557)

[12. Что делает метод map. 4](#_Toc40458558)

[13. Что делает метод flatMap? 4](#_Toc40458559)

[14. Что делает метод collect? 4](#_Toc40458560)

[15. Что делает метод reduce? 4](#_Toc40458561)

[16. Что делает метод findFirst? 5](#_Toc40458562)

[17. Возможно ли прервать выполнение потока по аналогии с break? 5](#_Toc40458563)

[18. Возможно ли пропустить элемент потока по аналогии с continue? 5](#_Toc40458564)

[19. Что такое Optional? 5](#_Toc40458565)

[20. Перечислите методы Optional? 5](#_Toc40458566)

[21. Расскажите про фабричные методы List.of, Set.of, Map.of? 5](#_Toc40458567)

[22. Для чего используется ключевое слово var? 6](#_Toc40458568)

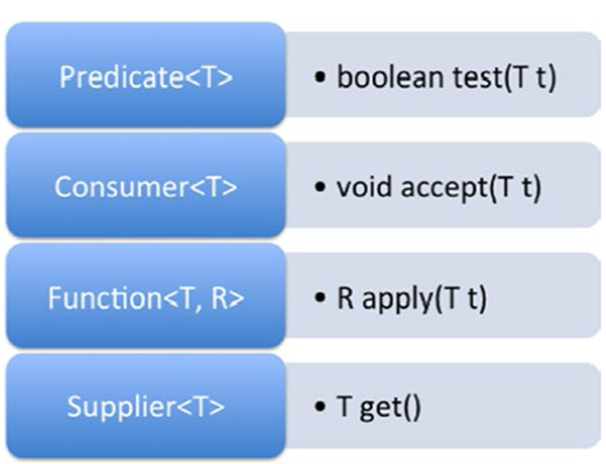
[23. В каких случаях можно использовать var? 6](#_Toc40458569)

# Что такое функциональные интерфейсы?

Функциональный интерфейс в Java – это интерфейс, который содержит только 1 абстрактный метод. Основное назначение – использование в лямбда выражениях и method reference.

Может содержать также сколько угодно default или static методов. Функциональные интерфейсы могут содержать дополнительно абстрактные методы, определенные в классе Object

# Перечислите функциональные интерфейсы из пакета java.util.function.



# Что такое функции высшего порядка?

Единственное, что нужно, чтобы сделать функцию высшего порядка, — это взять один из функциональных интерфейсов Java, или интерфейса SAM-типа в качестве аргумента и / или возвращать одно из них.

# Какие функциональные интерфейсы из пакета java.util.function поддерживают функции высшего порядка?

Да в принципе все базовые.

# Что такое ссылки на методы?

Сокращенный вызов методов через ключевое слово ::. Можно вызывать либо статические через обращение к классу, либо нестатические через объект.

# Что такое ссылки на конструкции?

Это типа, когда мы в ФИ засовываем лямбду и потом работаем с ФИ где ходим?

????

# Расскажите о зоне видимости переменных в lamdba - выражениях?

Лямбда выражения являются альтернативой анонимным классам. Но они не одинаковы.

Общее:

Локальные переменные могут быть использованы только если они final или effective final.

Разрешается доступ к переменным класса и статическим переменным класса.

Они не должны выбрасывать больше исключений чем определено в throws метода функционального интерфейса.

Различия:

Для анонимных классов ключевое слово this ссылается на сам класс. Для лямбда выражений на внешний класс.

Анонимные классы компилируются во внутренние классы. Но лямбда выражения преобразуются в статические private методы класса, в котором они используют invokedynamic инструкцию. Лямбда более эффективны, так как не надо загружать еще один класс.

# Как быть в ситуации, если внутри lamdba - выражении операторы могут выкинуть исключение?

Обернуть выражение внутри lambda в try-catch.

# Что такое Stream API?

Stream API это новый способ работать со структурами данных в функциональном стиле. Чаще всего с помощью stream в Java 8 работают с коллекциями, но на самом деле этот механизм может использоваться для самых различных данных.

# Расскажите, какой шаблон проектирования используется внутри Stream API?

Ну по ходу Builder.

# Что делает метод filter?

Аналог if, он есть предикат и убирает значаения, которые не проходят.

# Что делает метод map.

Преобразует один объект в другой. Ест в себя function.

# Что делает метод flatMap?

Похож на map, но возвращает несколько значений.

Например, если мы разобьем в flatMap объект на несколько, то на выходе получится объединенный стрим из всех этих значений.

А в Flat мы на выходе получим поток потоков.

# Что делает метод collect?

Собирает данные из потока в коллекции.

# Что делает метод reduce?

Метод reduce принимает лямбда-выражение известное как аккумулятор (Accumulator), которое служит для сворачивания данных в одну "кучу".

# Что делает метод findFirst?

Берет первый элемент из потока

# Возможно ли прервать выполнение потока по аналогии с break?

Терминальной операцией.

# Возможно ли пропустить элемент потока по аналогии с continue?

Можно через skip или limit. Ну или через filter.

# Что такое Optional?

Контейнер, который может содержать значение, либо его не содержать.

# Перечислите методы Optional?

Get(), isPresent. orElse

# Расскажите про фабричные методы List.of, Set.of, Map.of?

Создают неизменяемые коллекции, удобны для тестирования.

**Ограничения**

Вышеупомянутые коллекции нельзя изменить, при попытке получим UnsupportedOperationException;

В метод of() нельзя передать null, во всех коллекциях получим NullPointerException;

Метод of() не создает привычные ArrayList, HashSet или HashMap. В Java 9 были созданы специальные ImmutableCollections которые и возвращаются;

ImmutableCollections являются сериализуемыми, если содержимое тоже является сериализуемым;

# Для чего используется ключевое слово var?

Чтобы явно не указывать тип переменной.

# В каких случаях можно использовать var?

Когда мы явно получаем тип переменной из её инициализации.