

**1. Какие основные операции есть в лямбда-вычислениях? Опишите своими словами**

**2.**

Абстракция: создание функций. В лямбда-вычислениях это выражение вида  $\lambda x. x + 1$ , где  $\lambda x$  обозначает аргумент функции, а  $x + 1$  — тело функции.

Аппликация: применение функции к аргументу. Это выглядит как  $(\lambda x. x + 1) 2$ , и в результате выдаёт 3.

Редукция: процесс вычисления результата аппликации.

**3. Что такое функциональный интерфейс?**

Функциональный интерфейс в Java — это интерфейс, который имеет ровно один абстрактный метод. Он может иметь также дефолтные или статические методы.

**4. Что дает аннотация @FunctionalInterface? Обязательно ли ставить его над функциональным интерфейсом?**

Эта аннотация указывает, что интерфейс является функциональным и имеет только один абстрактный метод. Она необязательна, но добавляет ясность и позволяет компилятору провести проверку.

**5. Что такое анонимный класс?**

Анонимный класс — это класс без имени, обычно используемый для реализации интерфейсов на лету.

Пример:

```
new Thread(new Runnable() {  
    public void run() {  
        // код  
    }  
}).start();
```

**6. Как выглядит лямбда в джава и чем она является?**

В Java лямбда-выражения представляют собой краткий способ реализации функциональных интерфейсов.

Пример:

$(a, b) \rightarrow a + b$

Это лямбда-выражение реализует функциональный интерфейс с методом, принимающим два аргумента и возвращающим их сумму.

**7. Какие функциональные интерфейсы существуют в джава?**

Runnable: не принимает аргументов и не возвращает результат.

Supplier<T>: не принимает аргументов, но возвращает результат типа T.

Consumer<T>: принимает аргумент типа T и не возвращает результат.

Function<T, R>: принимает аргумент типа T и возвращает результат типа R.

Predicate<T>: принимает аргумент типа T и возвращает boolean.