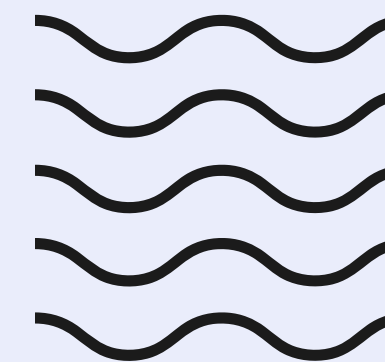


Lambdas



ПОДГОТОВИЛ: АЛИШЕР ХАМИДОВ



Основные моменты:

План лекции

1. Что такое лямбда?
2. Функциональный интерфейс
3. Блочные и однострочные лямбды





Лямбда-выражения являются в некотором роде сокращенной формой внутренних анонимных классов, которые ранее применялись в Java.





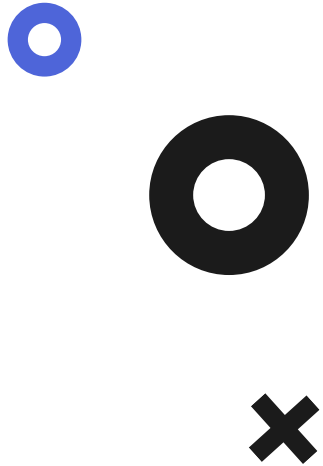
Лямбда представляет набор инструкций, которые можно выделить в отдельную переменную и затем многократно вызвать в различных местах программы.






Лямбда-выражение не выполняется само по себе,
а **образует реализацию метода**, определенного в
функциональном интерфейсе.

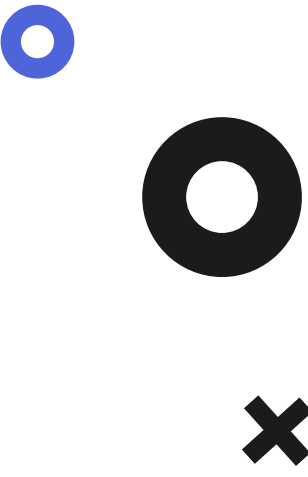
Функциональный интерфейс должен содержать
только один единственный метод без реализации.



```
// функциональный интерфейс  
interface Operationable{  
    int calculate(int x, int y);  
}
```



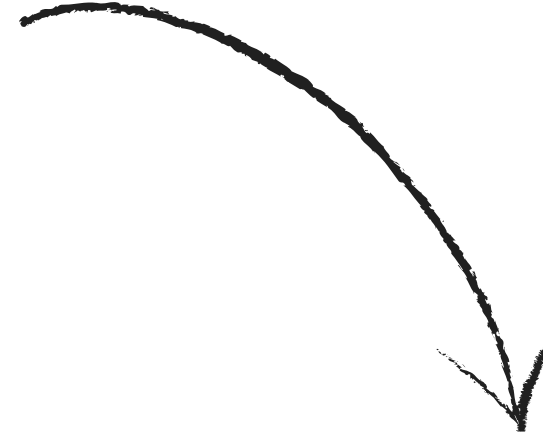
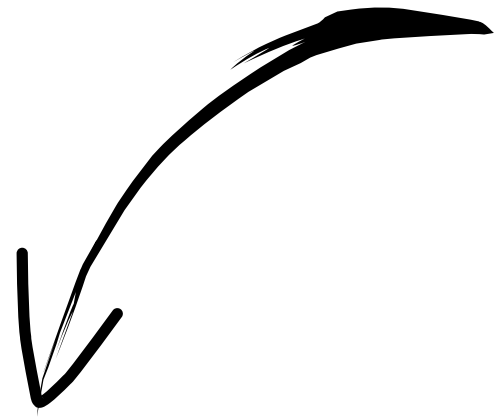
```
// рекомендуется отмечать аннотацией  
@FunctionalInterface  
interface Operationable{  
    int calculate(int x, int y);  
}
```



```
public static void main(String[] args) {  
    Operationable operation;  
    operation = (x,y)->x+y;  
  
    int result = operation.calculate(5, 10);  
    System.out.println(result); // 15  
}
```


- Определение ссылки на функциональный интерфейс:
`Operationable operation;`
- Создание лямбда-выражения:
`operation = (x,y)->x+y;`
- Использование лямбда-выражения в виде вызова метода интерфейса:
`int result = operation.calculate(5, 10);`

Лямбды




Блочные

```
Operation op = (a+b)->{  
    return m+n;  
};
```




Однострочные

```
Operation op = (a+b)->m+n;
```

- 
- Блочные выражения обрамляются фигурными скобками.
 - В блочных лямбда-выражениях можно использовать внутренние вложенные блоки, циклы, конструкции if, switch, создавать переменные и т.д.
 - Если блочное лямбда-выражение должно возвращать значение, то явным образом применяется оператор return



Лямбда выражение, которое ничего не возвращает
называется **терминальным**



Функциональный интерфейс может быть обобщенным(с дженериками), однако в лямбда-выражении использование обобщений не допускается.

В этом случае нам надо типизировать объект интерфейса определенным типом, который потом будет применяться в лямбда-выражении.