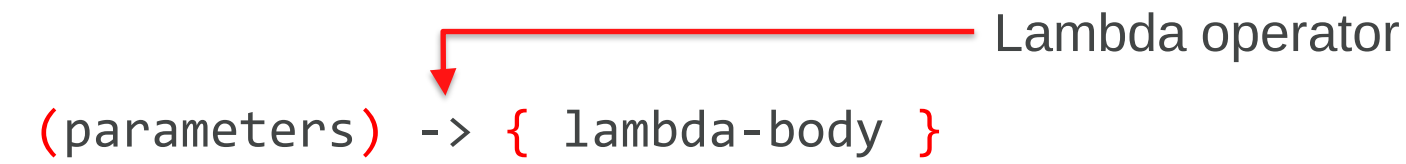
ORACLE

Lesson 1-2:
Lambda Expression Syntax

Lambda Expressions Are Anonymous Functions

Which Are Like Methods But Without A Class



- Body of the Lambda may throw exceptions
- Single line Lambdas
  - Do not need braces
  - Do not need an explicit return statement
- Lambdas with a single parameter do not need brackets
- Lambdas with no parameters must have empty brackets

Lambda Expression Syntax

Examples

- `() -> System.out.println("Hello Lambda")`
- `x -> x + 10`
- `(int x, int y) -> { return x + y; }`
- `(String x, String y) -> x.length() – y.length()`
-
```
(String x) -> {
    listA.add(x);
    listB.remove(x);
    return listB.size();
}
```

Lambda Expressions

Type Inference

- Example method definition
  - `static <T> T process(List<T> l, Comparator<T> c)`
- Use the method
  - `List<String> list = getList();`
  - `process(list, (String x, String y) -> x.length() – y.length());`
- Compiler is now smarter
  - `String r = process(list, (x, y) -> x.length() – y.length())`
- More typing with less typing

Section 2

Summary

- Syntax for Lambda expressions is simple
  - Brackets and braces are optional for certain situations
- Type inference means types often do not need to be explicitly stated
  - Java remains strongly, statically typed


ORACLE®