









































































































Lambda expressions

- a lambda expression is always surrounded by curly braces
- parameter declarations in the full syntactic form go inside curly braces and have optional type annotations
- the body goes after an `->` sign
- if the inferred return type of the lambda is not `Unit`, the last (or possibly single) expression inside the lambda body is treated as the return value



`{s: string -> s.length}`

```
{  a: Int,  b: Int ->  
    println("$a + $b")  
    a + b  
}
```

Lambda expressions

```
{ s: String -> s.length }
```

```
{ a: Int, b: Int ->  
  println("$a + $b")  
  a + b  
}
```

- a lambda expression is always surrounded by curly braces
- parameter declarations in the full syntactic form go inside curly braces and have optional type annotations
- the body goes after an `->` sign
- if the inferred return type of the lambda is not `Unit`, the last (or possibly single) expression inside the lambda body is treated as the return value

Lambda expressions

are expressions