Wealthsimple

Functional Programming With Ruby

Pei Li

pei@wealthsimple.com

wealthsimple.com
@wealthsimple

Why Functional Programming?

- Concise.

  Achieve more with fewer lines of code.

- Predictable.

  Easily understand and predict what the code does.

- Testable.

  Greater confidence in the correctness of your code.

Functional Programming…

- Avoids shared states.

- Avoids mutable data.

- Avoids side effects.

The Fundamentals

- map

  Applies an operation to every element in a list.

- select

  Filters a list based on the evaluation of an expression.

- reduce

  Applies an operation that aggregates every element in a list.

Map

```
list.map { |element| operation(element) }
```

Select

```
list.select { |element| expression(element) }
```

Reduce

```
list.reduce { |aggregation, element| operation(aggregation, element) }
```

Putting It All Together

Live Demo

Wealthsimple

Questions?

Pei Li

pei@wealthsimple.com

wealthsimple.com
@wealthsimple