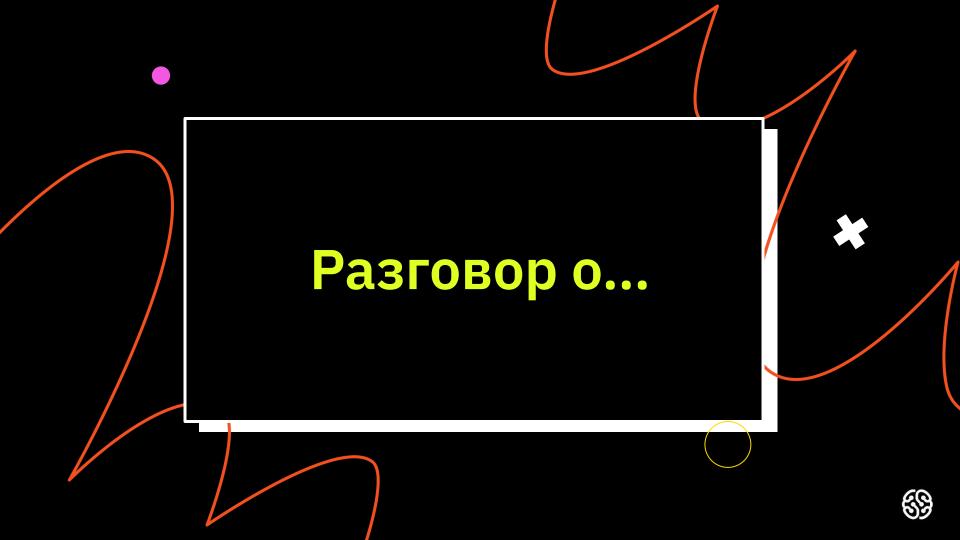




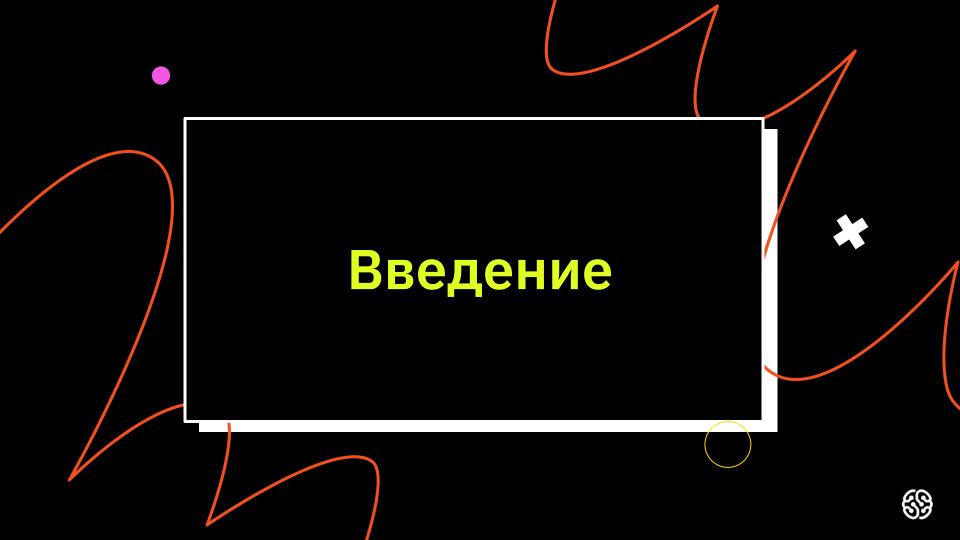
... и новое поведение



Разговор о...

- **1.** Введение
- 2. Iterator<E>
- 3. Iterable<E>
- 4. Comparator<E>
- 5. Comparable < E >
- **6.** Comparable, equals, ==
- 7. foreach для своих типов
- 8. Примеры
- 9. Итоги

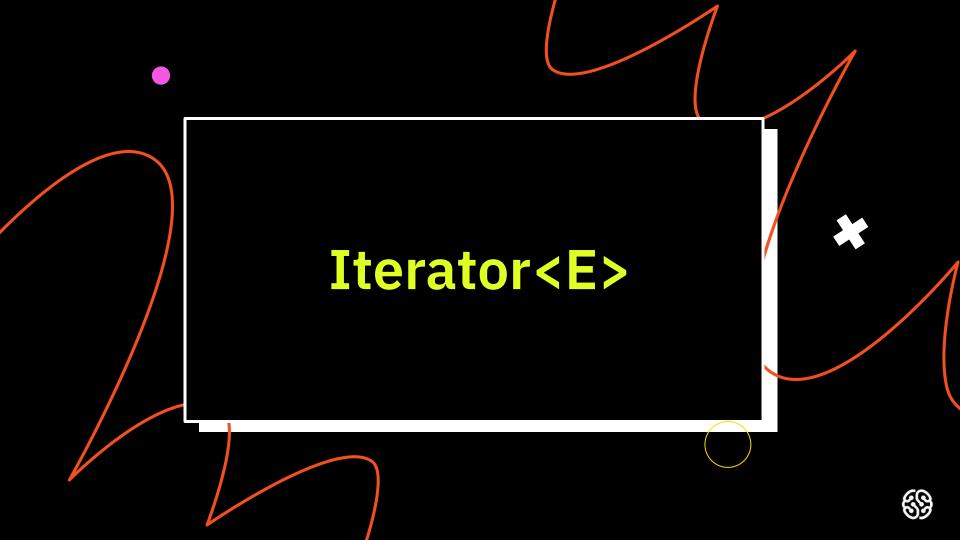




Введение

Некоторые стандартные интерфейсы и придание нового поведения своим типам.



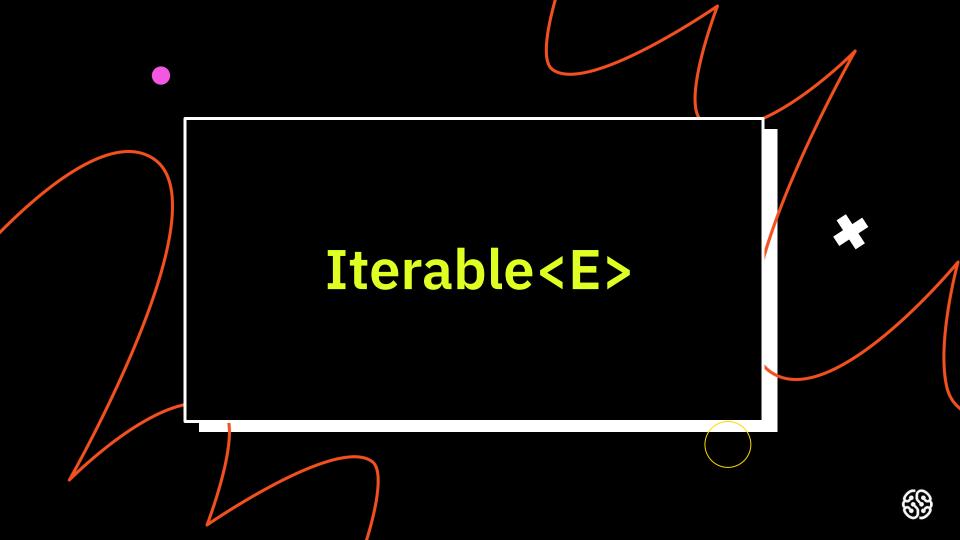


Iterator<E>

Итератор над «коллекцией». Iterator занимает место Enumeration'в Java Collections Framework.

hasNext()
next()

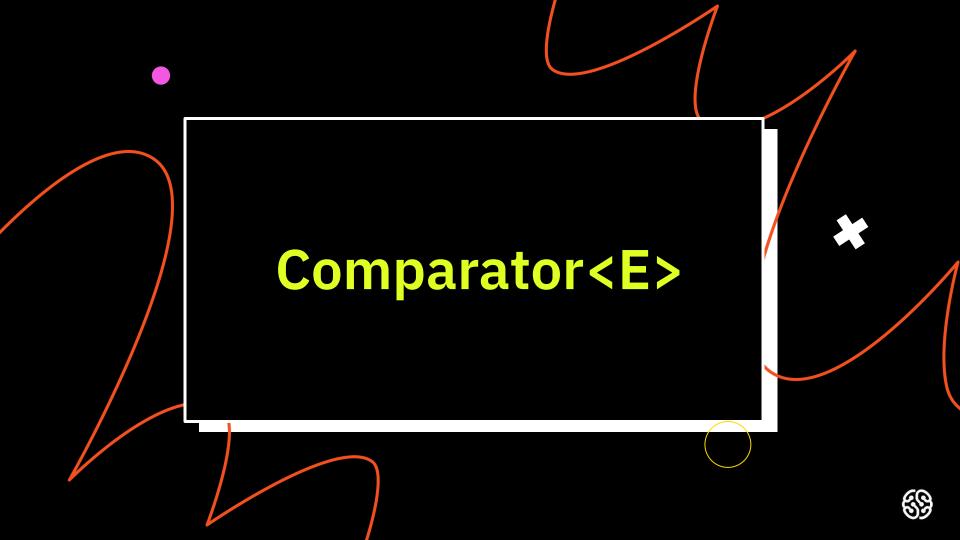




Iterable < E >

Позволяет собственному типу быть типом оператора «for-each loop»





Comparator<E>

Предназначен для упорядочивания собственных типов.

obj.compare $(x, y) \le 1$

obj.compare $(x, y) \le 0$

obj.compare $(x, y) \le -1$



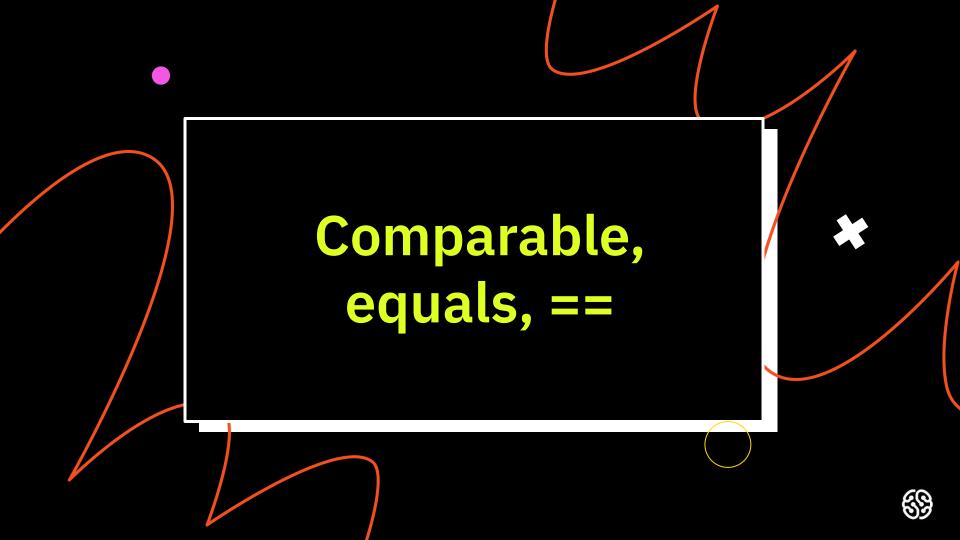


Comparable < E >

Списки (и массивы) собственных типов позволяют автоматически сортироваться при помощи

Collections.sort Arrays.sort





Comparable, equals, ==

Демонстрация





Итоги

Интерфейсов много

Cloneable

Serializable

*able и др.

«Заставить» работать стандартный функционал. Работа со своими свои типы «как нужно». Определить «свою» логику работы.



Спасибо // за внимание /

