Advanced Topics

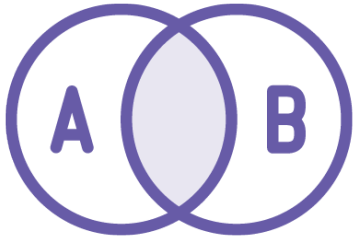
Richard Warburton

@richardwarburto www.monotonic.co.uk

Contents

Functional Interfaces

Type Inference



Intersection Types

Varargs and Generics

Functional Interfaces: Wildcards Use Case

```
Comparator<T> → int compare(T o1, T o2);
```

```
Comparator<Foo>
```

```
Comparator<? super Foo>
```

The Comparator Interface

```
Function<T, R> → R apply(T arg)

Function<Foo, Bar>

Function<? super Foo, ? extends Bar>
```

The Function Interface

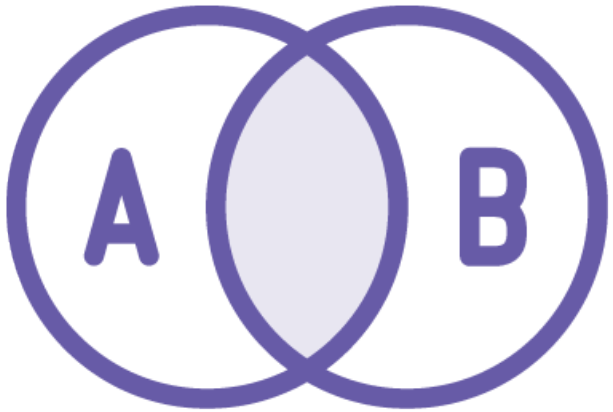
Type Inference and Generics

Intersection Types

Intersection

$A \cap B$

Elements are in both A and B



Intersection Type

<T extends A & B>

T is a subtype of A and B

```
<T extends Object & Comparable<? super T>>

T max(Collection<? extends T> coll)
```

A Very Confusing Example of Intersection Types

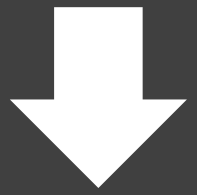
```
Object max(Collection coll)
```

Pre-generics Signature

`max` has to preserve binary compatibility with this signature.

You can only find the `max` if the objects are `Comparable`

```
<T extends Comparable<? super T>>
T max(Collection<? extends T> coll)
```

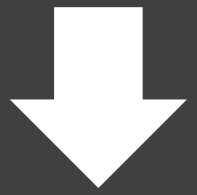


```
Comparable max(Collection coll)
```

Badly Erased Signature

Javac compiles this in an awkward way

```
<T extends Object & Comparable<? super T>>
T max(Collection<? extends T> coll)
```



```
Object max(Collection coll)
```

Intersection Types Enable Compatibility

Varargs and Generics

Conclusions and Recap

Generics and Collections

Generics let us add type parameters to classes

The most common users of this feature are Java Collections

Subtypes and Wildcards

In Java, we often implement interfaces or extend classes

- We can do this with Generic Classes

Wildcards make type parameters more flexible

- super → data in
- extends → data out

Reflection and Rawtypes

You can't reflect all generic information

- Because it gets erased at compile time

But we can talk to legacy code

- Rawtypes let's us represent a type without a parameter

Lambdas and Intersection Types

Lambdas have their types inferred

- Generics can result in confusing errors.

Intersection Types

- <T extends A & B> → advanced but powerful


THE NATIONAL BANK LIMITED
NIGHT SAFE