Advanced Topics

Richard Warburton

@richardwarburto | www.insightfullogic.com

Functional Interfaces | Type Inference | Intersection Types

Functional Interfaces

A great use case for wildcards

The Comparator Interface

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

Comparator<Foo>

Comparator<? super Foo>
```

The Function Interface

```
Function<T, R> → R apply(T arg)

Function<Foo, Bar>

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

Type Inference

How does type inference interplay with Generics?

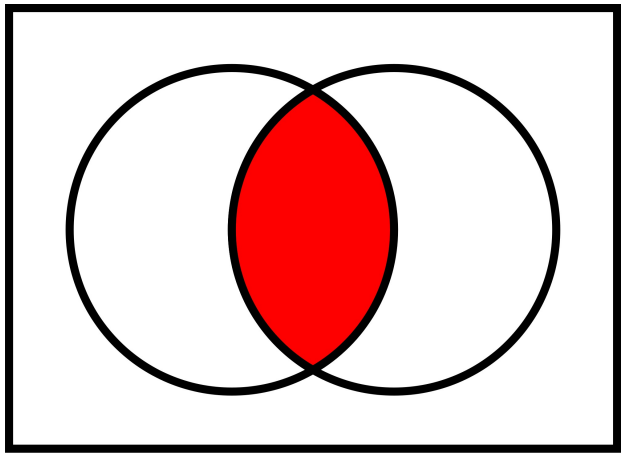
Intersection Types

A very advanced Generics feature

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

Conclusions & 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 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