

# Priority Queues

A Java `PriorityQueue<T>` works for any type `T` that has a *natural ordering*.

Any type `T` that implements the `Comparable<T>` interface has a natural ordering.

Java's `PriorityQueue` prioritises values that are lower in the natural ordering.

Kotlin's `PriorityQueue` class can optionally be constructed with a `Comparator` object, i.e. an object that implements the `Comparator` interface. If provided, this ordering will be used for prioritisation.

If the type `T` already has a natural ordering and a custom ordering is provided via a comparator, the custom ordering will be used.

If `null` is passed as the optional comparator argument, then `T`'s natural ordering (if it exists) will be used, just as if no comparator argument had been provided to the `PriorityQueue` constructor.