Guava Release 20.0: Release Notes

- 20.0 was released on October 28, 2016.
- 20.0-rc1 was released on October 06, 2016.

(See [[ReleaseHistory]].)

API documentation:

- guava
- guava-testlib

Known issues

- If you see errors like "cannot access com.google.errorprone.annotations.CanIgnoreReturnValue," you can work around them by adding a local dependency on `error_prone_annotations`. (This problem is fixed in Guava 22 (but not 21), which makes that dependency present for users.)
- For GWT, see another known issue under GWT notes.

Using Guava in your project

	Guava	Guava (GWT)
Maven Identifier	com.google.guava:guava:20.0	com.google.guava:guava-gwt:20.0
Jar	guava-20.0.jar	guava-gwt-20.0.jar
Javadoc	guava-20.0-javadoc.jar	guava-gwt-20.0-javadoc.jar
Sources	guava-20.0-sources.jar	guava-gwt-20.0-sources.jar

See [[UseGuavaInYourBuild]] for help integrating Guava into your build environment.

GWT notes

- **Known issue:** If you see `No source code is available for type java.lang.InterruptedException`, add `<inherits name="java.lang.Lang"/>` to your `.gwt.xml`.
- Guava 20.0 *requires* GWT 2.8.0.
- Additionally, Guava 20.0 no longer works with the deprecated GWT "classic" Dev Mode. It continues to work with Super Dev Mode.

Issues resolved

34+ issues are resolved in this release.

API Changes

Full JDiff Report of changes since release 19.0.

Significant API additions and changes

New package! [[common.graph|GraphsExplained]]

`common.graph` is a library for modeling graph-structured data, that is, entities and the relationships between them. Its purpose is to provide a common and extensible language for working with such data.

common.base

- `CharMatcher` constants have been deprecated in favor of the static factory methods that were added in 19.0. The constants will be removed after a 2-year deprecation cycle.
- `Preconditions` : new overloads of `checkNotNull` and `checkState` added to avoid varargs array allocation and primitive boxing for the most common argument combinations.
- `Predicates` : `assignableFrom(Class<?>)` deprecated and the correctly-named equivalent `subtypeOf(Class<?>)` added.
- `Throwables`
    - `throwIfInstanceOf` and `throwIfUnchecked` added.
    - `propagate` , `propagateIfInstanceOf` and `propagateIfPossible` deprecated.

common.collect

- `ConcurrentHashMultiset` : `create(MapMaker)` deprecated and `create(ConcurrentMap)` added.
- `FluentIterable` : a number of new static factory methods, such as `concat(Iterable)` and `of()` added.
- `Iterators` : deprecated method `emptyIterator()` removed.
- `MapConstraints` : Most methods removed; the class was scheduled to be removed in this release, but full removal is pushed back.
- `Maps` : `subMap(NavigableMap, Range)` added.
- `Ordering` : `binarySearch` deprecated.
- `RangeSet` : `intersects(Range)` added.
- `Sets` : `subSet(NavigableSet, Range)` added.
- `TreeTraverser` : factory method `using(Function)` added to adapt a node -> children `Function` to a `TreeTraverser` .

common.hash

- `Hashing` : a number of new hash functions added, including FarmHash Fingerprint64 and a number of HMAC algorithms.

common.io

- `InputSupplier` and `OutputSupplier` : removed.
- `BaseEncoding` : `canDecode(CharSequence)` added.
- `ByteStreams` : `exhaust(InputStream)` added.
- `CharSource` : `asByteSource(Charset)` added.
- `CharStreams` : `exhaust(Readable)` added.

common.math

Many additions, most related to statistics:

- `Quantiles`
- `Stats` and `StatsAccumulator` , `PairedStats` and `PairedStatsAccumulator`
- `LinearTransformation`
- `DoubleMath` : `mean` methods deprecated in favor of using `Stats` .

New methods added to `IntMath` , `LongMath` and `DoubleMath` such as:

- `ceilingPowerOfTwo`
- `isPrime`

common.net

- `HostAndPort` : `getHostText()` deprecated in favor of new `getHost()` method.
- `HttpHeaders` and `MediaType` : a number of new header/media type constants added.

common.reflect

- `ClassPath.ResourceInfo` : `asByteSource()` and `asCharSource(Charset)` methods added.
- `TypeToken` : `isAssignableFrom` methods removed (replaced by `isSubtypeOf` in 19.0).

common.util.concurrent

- `FutureFallback` : removed.
  - `Futures.withFallback` methods removed.
- `AsyncCallable` : added.
  - `Callables.asAsyncCallable(Callable, ListeningExecutorService)` added.
- `Futures.FutureCombiner` : added.
  - `Futures.whenAllComplete` and `whenAllSucceed` , returning `FutureCombiner` , added.
- `AbstractFuture` : `afterDone()` callback added.
- `AtomicLongMap` : `removeIfZero(K)` added.
- `Futures` :
  - `get` methods taking an exception `Class` removed; previously replaced with `getChecked` .
  - `transform` methods taking `AsyncFunction` removed; previously replaced with `transformAsync` .
  - `getDone(Future)` added.