|  |  |
| --- | --- |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [acceptEither](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#acceptEither-java.util.concurrent.CompletionStage-java.util.function.Consumer-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed with the corresponding result as argument to the supplied action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [acceptEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#acceptEitherAsync-java.util.concurrent.CompletionStage-java.util.function.Consumer-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed using this stage's default asynchronous execution facility, with the corresponding result as argument to the supplied action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [acceptEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#acceptEitherAsync-java.util.concurrent.CompletionStage-java.util.function.Consumer-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed using the supplied executor, with the corresponding result as argument to the supplied function. |
| static [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [allOf](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#allOf-java.util.concurrent.CompletableFuture...-)([CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<?>... cfs)  Returns a new CompletableFuture that is completed when all of the given CompletableFutures complete. |
| static [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Object](https://docs.oracle.com/javase/8/docs/api/java/lang/Object.html)> | [anyOf](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#anyOf-java.util.concurrent.CompletableFuture...-)([CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<?>... cfs)  Returns a new CompletableFuture that is completed when any of the given CompletableFutures complete, with the same result. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [applyToEither](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#applyToEither-java.util.concurrent.CompletionStage-java.util.function.Function-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),U> fn)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed with the corresponding result as argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [applyToEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#applyToEitherAsync-java.util.concurrent.CompletionStage-java.util.function.Function-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),U> fn)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed using this stage's default asynchronous execution facility, with the corresponding result as argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [applyToEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#applyToEitherAsync-java.util.concurrent.CompletionStage-java.util.function.Function-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> other, [Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),U> fn, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when either this or the other given stage complete normally, is executed using the supplied executor, with the corresponding result as argument to the supplied function. |
| boolean | [cancel](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#cancel-boolean-)(boolean mayInterruptIfRunning)  If not already completed, completes this CompletableFuture with a **[CancellationException](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CancellationException.html" \o "class in java.util.concurrent)**. |
| boolean | [complete](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#complete-T-)([T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) value)  If not already completed, sets the value returned by [**get()**](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get--) and related methods to the given value. |
| static <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [completedFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#completedFuture-U-)(U value)  Returns a new CompletableFuture that is already completed with the given value. |
| boolean | [completeExceptionally](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#completeExceptionally-java.lang.Throwable-)([Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html) ex)  If not already completed, causes invocations of [**get()**](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get--) and related methods to throw the given exception. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> | [exceptionally](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#exceptionally-java.util.function.Function-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<[Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html),? extends [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> fn)  Returns a new CompletableFuture that is completed when this CompletableFuture completes, with the result of the given function of the exception triggering this CompletableFuture's completion when it completes exceptionally; otherwise, if this CompletableFuture completes normally, then the returned CompletableFuture also completes normally with the same value. |
| [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) | [get](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get--)()  Waits if necessary for this future to complete, and then returns its result. |
| [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) | [get](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get-long-java.util.concurrent.TimeUnit-)(long timeout, [TimeUnit](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/TimeUnit.html" \o "enum in java.util.concurrent) unit)  Waits if necessary for at most the given time for this future to complete, and then returns its result, if available. |
| [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) | [getNow](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#getNow-T-)([T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) valueIfAbsent)  Returns the result value (or throws any encountered exception) if completed, else returns the given valueIfAbsent. |
| int | [getNumberOfDependents](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#getNumberOfDependents--)()  Returns the estimated number of CompletableFutures whose completions are awaiting completion of this CompletableFuture. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [handle](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#handle-java.util.function.BiFunction-)([BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "type parameter in CompletableFuture),[Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html),? extends U> fn)  Returns a new CompletionStage that, when this stage completes either normally or exceptionally, is executed with this stage's result and exception as arguments to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [handleAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#handleAsync-java.util.function.BiFunction-)([BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "type parameter in CompletableFuture),[Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html),? extends U> fn)  Returns a new CompletionStage that, when this stage completes either normally or exceptionally, is executed using this stage's default asynchronous execution facility, with this stage's result and exception as arguments to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [handleAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#handleAsync-java.util.function.BiFunction-java.util.concurrent.Executor-)([BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "type parameter in CompletableFuture),[Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html),? extends U> fn, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this stage completes either normally or exceptionally, is executed using the supplied executor, with this stage's result and exception as arguments to the supplied function. |
| boolean | [isCancelled](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#isCancelled--)()  Returns true if this CompletableFuture was cancelled before it completed normally. |
| boolean | [isCompletedExceptionally](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#isCompletedExceptionally--)()  Returns true if this CompletableFuture completed exceptionally, in any way. |
| boolean | [isDone](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#isDone--)()  Returns true if completed in any fashion: normally, exceptionally, or via cancellation. |
| [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) | [join](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#join--)()  Returns the result value when complete, or throws an (unchecked) exception if completed exceptionally. |
| void | [obtrudeException](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#obtrudeException-java.lang.Throwable-)([Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html) ex)  Forcibly causes subsequent invocations of method [**get()**](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get--) and related methods to throw the given exception, whether or not already completed. |
| void | [obtrudeValue](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#obtrudeValue-T-)([T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html) value)  Forcibly sets or resets the value subsequently returned by method [**get()**](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#get--) and related methods, whether or not already completed. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterBoth](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterBoth-java.util.concurrent.CompletionStage-java.lang.Runnable-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when this and the other given stage both complete normally, executes the given action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterBothAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterBothAsync-java.util.concurrent.CompletionStage-java.lang.Runnable-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when this and the other given stage complete normally, executes the given action using this stage's default asynchronous execution facility. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterBothAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterBothAsync-java.util.concurrent.CompletionStage-java.lang.Runnable-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this and the other given stage complete normally, executes the given action using the supplied executor. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterEither](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterEither-java.util.concurrent.CompletionStage-java.lang.Runnable-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when either this or the other given stage complete normally, executes the given action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterEitherAsync-java.util.concurrent.CompletionStage-java.lang.Runnable-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when either this or the other given stage complete normally, executes the given action using this stage's default asynchronous execution facility. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAfterEitherAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAfterEitherAsync-java.util.concurrent.CompletionStage-java.lang.Runnable-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html)<?> other, [Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when either this or the other given stage complete normally, executes the given action using the supplied executor. |
| static [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAsync-java.lang.Runnable-)([Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) runnable)  Returns a new CompletableFuture that is asynchronously completed by a task running in the **[ForkJoinPool.commonPool()](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/ForkJoinPool.html" \l "commonPool--)** after it runs the given action. |
| static [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [runAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#runAsync-java.lang.Runnable-java.util.concurrent.Executor-)([Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) runnable, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletableFuture that is asynchronously completed by a task running in the given executor after it runs the given action. |
| static <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [supplyAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#supplyAsync-java.util.function.Supplier-)([Supplier](https://docs.oracle.com/javase/8/docs/api/java/util/function/Supplier.html)<U> supplier)  Returns a new CompletableFuture that is asynchronously completed by a task running in the **[ForkJoinPool.commonPool()](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/ForkJoinPool.html" \l "commonPool--)** with the value obtained by calling the given Supplier. |
| static <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [supplyAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#supplyAsync-java.util.function.Supplier-java.util.concurrent.Executor-)([Supplier](https://docs.oracle.com/javase/8/docs/api/java/util/function/Supplier.html)<U> supplier, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletableFuture that is asynchronously completed by a task running in the given executor with the value obtained by calling the given Supplier. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAccept](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAccept-java.util.function.Consumer-)([Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action)  Returns a new CompletionStage that, when this stage completes normally, is executed with this stage's result as the argument to the supplied action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAcceptAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAcceptAsync-java.util.function.Consumer-)([Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action)  Returns a new CompletionStage that, when this stage completes normally, is executed using this stage's default asynchronous execution facility, with this stage's result as the argument to the supplied action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAcceptAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAcceptAsync-java.util.function.Consumer-java.util.concurrent.Executor-)([Consumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this stage completes normally, is executed using the supplied Executor, with this stage's result as the argument to the supplied action. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAcceptBoth](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAcceptBoth-java.util.concurrent.CompletionStage-java.util.function.BiConsumer-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U> action)  Returns a new CompletionStage that, when this and the other given stage both complete normally, is executed with the two results as arguments to the supplied action. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAcceptBothAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAcceptBothAsync-java.util.concurrent.CompletionStage-java.util.function.BiConsumer-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U> action)  Returns a new CompletionStage that, when this and the other given stage complete normally, is executed using this stage's default asynchronous execution facility, with the two results as arguments to the supplied action. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenAcceptBothAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenAcceptBothAsync-java.util.concurrent.CompletionStage-java.util.function.BiConsumer-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U> action,[Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this and the other given stage complete normally, is executed using the supplied executor, with the two results as arguments to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenApply](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenApply-java.util.function.Function-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends U> fn)  Returns a new CompletionStage that, when this stage completes normally, is executed with this stage's result as the argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenApplyAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenApplyAsync-java.util.function.Function-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends U> fn)  Returns a new CompletionStage that, when this stage completes normally, is executed using this stage's default asynchronous execution facility, with this stage's result as the argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenApplyAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenApplyAsync-java.util.function.Function-java.util.concurrent.Executor-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends U> fn, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this stage completes normally, is executed using the supplied Executor, with this stage's result as the argument to the supplied function. |
| <U,V> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<V> | [thenCombine](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenCombine-java.util.concurrent.CompletionStage-java.util.function.BiFunction-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U,? extends V> fn)  Returns a new CompletionStage that, when this and the other given stage both complete normally, is executed with the two results as arguments to the supplied function. |
| <U,V> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<V> | [thenCombineAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenCombineAsync-java.util.concurrent.CompletionStage-java.util.function.BiFunction-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U,? extends V> fn)  Returns a new CompletionStage that, when this and the other given stage complete normally, is executed using this stage's default asynchronous execution facility, with the two results as arguments to the supplied function. |
| <U,V> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<V> | [thenCombineAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenCombineAsync-java.util.concurrent.CompletionStage-java.util.function.BiFunction-java.util.concurrent.Executor-)([CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<? extends U> other, [BiFunction](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiFunction.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super U,? extends V> fn,[Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this and the other given stage complete normally, is executed using the supplied executor, with the two results as arguments to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenCompose](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenCompose-java.util.function.Function-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends [CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<U>> fn)  Returns a new CompletionStage that, when this stage completes normally, is executed with this stage as the argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenComposeAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenComposeAsync-java.util.function.Function-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends [CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<U>> fn)  Returns a new CompletionStage that, when this stage completes normally, is executed using this stage's default asynchronous execution facility, with this stage as the argument to the supplied function. |
| <U> [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html" \o "class in java.util.concurrent)<U> | [thenComposeAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenComposeAsync-java.util.function.Function-java.util.concurrent.Executor-)([Function](https://docs.oracle.com/javase/8/docs/api/java/util/function/Function.html)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? extends [CompletionStage](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletionStage.html" \o "interface in java.util.concurrent)<U>> fn, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this stage completes normally, is executed using the supplied Executor, with this stage's result as the argument to the supplied function. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenRun](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenRun-java.lang.Runnable-)([Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when this stage completes normally, executes the given action. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenRunAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenRunAsync-java.lang.Runnable-)([Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action)  Returns a new CompletionStage that, when this stage completes normally, executes the given action using this stage's default asynchronous execution facility. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[Void](https://docs.oracle.com/javase/8/docs/api/java/lang/Void.html)> | [thenRunAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#thenRunAsync-java.lang.Runnable-java.util.concurrent.Executor-)([Runnable](https://docs.oracle.com/javase/8/docs/api/java/lang/Runnable.html) action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage that, when this stage completes normally, executes the given action using the supplied Executor. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> | [toCompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#toCompletableFuture--)()  Returns this CompletableFuture. |
| [String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) | [toString](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#toString--)()  Returns a string identifying this CompletableFuture, as well as its completion state. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> | [whenComplete](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#whenComplete-java.util.function.BiConsumer-)([BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super [Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html)> action)  Returns a new CompletionStage with the same result or exception as this stage, that executes the given action when this stage completes. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> | [whenCompleteAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#whenCompleteAsync-java.util.function.BiConsumer-)([BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super [Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html)> action)  Returns a new CompletionStage with the same result or exception as this stage, that executes the given action using this stage's default asynchronous execution facility when this stage completes. |
| [CompletableFuture](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)<[T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html)> | [whenCompleteAsync](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html#whenCompleteAsync-java.util.function.BiConsumer-java.util.concurrent.Executor-)([BiConsumer](https://docs.oracle.com/javase/8/docs/api/java/util/function/BiConsumer.html" \o "interface in java.util.function)<? super [T](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/CompletableFuture.html),? super [Throwable](https://docs.oracle.com/javase/8/docs/api/java/lang/Throwable.html)> action, [Executor](https://docs.oracle.com/javase/8/docs/api/java/util/concurrent/Executor.html) executor)  Returns a new CompletionStage with the same result or exception as this stage, that executes the given action using the supplied Executor when this stage completes. |