Here’s a summary of the main **Java 8 Concurrency API** enhancements:

**1. CompletableFuture**

* Asynchronous computation, chaining, combining, and exception handling.
* Located in java.util.concurrent.

**Common Methods:**

* CompletableFuture.supplyAsync(Supplier<T>)
* thenApply(Function<T,R>)
* thenAccept(Consumer<T>)
* thenCombine(CompletableFuture<U>, BiFunction<T,U,R>)
* exceptionally(Function<Throwable,T>)
* get(), join()
* allOf(), anyOf()

**Example:**

CompletableFuture.supplyAsync(() -> "Hello")

    .thenApply(s -> s + " World")

    .thenAccept(System.out::println);

**2. Parallel Streams**

* Process collections in parallel using .parallelStream().

**Example:**

list.parallelStream().forEach(System.out::println);

**3. Other Enhancements**

* New methods in ConcurrentHashMap (e.g., forEach, search, reduce).
* Improved fork/join framework.

**Note:**  
Java 8 makes asynchronous and parallel programming easier and more expressive with these APIs.