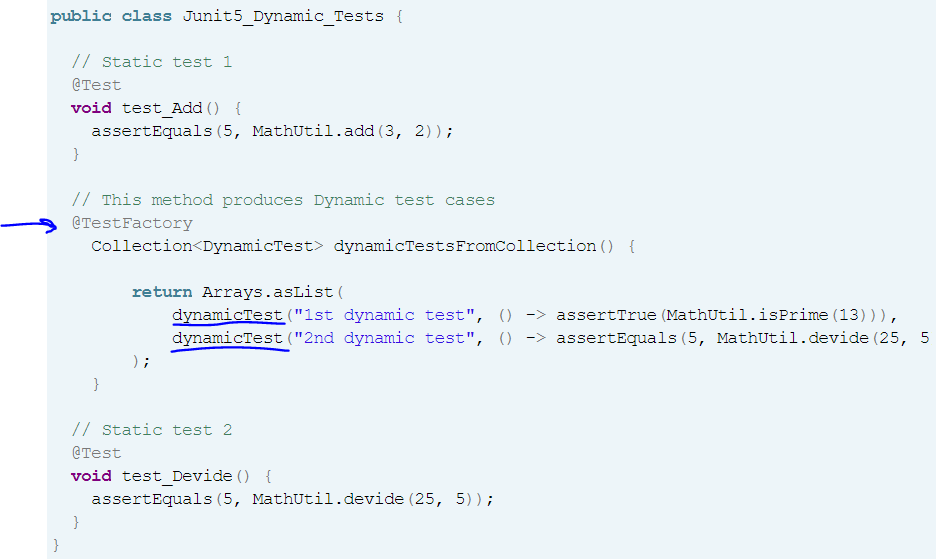
A dynamic test is a test that generated at runtime by factory method using [@TestFactory](https://javabydeveloper.com/junit-5-dynamic-tests-testfactory-with-examples/#@TestFactory) annotation. The method marked **@TestFactory** is not a test case, rather it’s a factory for test cases. @TestFactory denotes that the method is test factory for dynamic tests. @TestFactory method differs with @Test method in such a way that @TestFactory method is not itself a test case but rather a factory for test cases. So we can say that dynamic test is the product of a factory method.  
  
A **DynamicTest** stores the test's name as a String and its code as an Executable - that's like a Runnable that can throw a Throwable. It is created with a static factory method:

Syntax: public static DynamicTest dynamicTest(**String name, Executable test**);  


**Rules to create Junit 5 Dynamic Tests**

1. @TestFactory methods must not be private or static.
2. @TestFactory methods must return Stream, Collection, Iterable, Iterator, or array of DynamicNode instances.



