

#	Time	File	Event Type	Raw Event	Zorro's Inference
1	(1) 07:20:53	TestIntegerToRoman.java	ADD METHOD	TestIntegerToRoman(String)	<p>This portion of development appears to be TDD conformant because:</p> <p>Tests were written before production code.</p> <p>This episode looks like a typical test-first episode because:</p> <p>Some tests were added (5). Then a compilation error occurred (6). Then production code was added (11). However, tests ran without failure.</p>
	(2) 07:20:54	TestIntegerToRoman.java	ADD CLASS	TestIntegerToRoman.java	
	(3) 07:20:54	TestIntegerToRoman.java	BUFFTRANS	FROM TestStack.java	
	(4) 07:21:05	TestIntegerToRoman.java	ADD METHOD	void testZeroReturnsEmpty()	
	(5) 07:24:44	TestIntegerToRoman.java	TEST EDIT	212sec MI=+2(2), SI=+3(1), TI=+1(1), AI=+1(0), FI=+307(307)	
	(6) 07:24:44	TestIntegerToRoman.java	COMPILE	Roman cannot be resolved to a type	
	(7) 07:25:08	Roman.java	ADD CLASS	Roman.java	
	(8) 07:25:09	Roman.java	BUFFTRANS	FROM TestIntegerToRoman.java	
	(9) 07:25:23	Roman.java	ADD METHOD	Roman(int)	
	(10) 07:25:38	Roman.java	ADD FIELD	int intValue	
	(11) 07:26:19	Roman.java	PRODUCTION EDIT	36sec MI=+1(1), SI=+1(1), FI=+158(158)	
	(12) 07:26:19	Roman.java	COMPILE	integerValue cannot be resolved	
	(13) 07:26:42	Roman.java	PRODUCTION EDIT	0sec MI=0(1), SI=0(1), FI=+16(174)	
	(14) 07:26:48	Roman.java	ADD METHOD	String toString()	
	(15) 07:27:09	Roman.java	PRODUCTION EDIT	0sec MI=+1(2), SI=0(1), FI=+25(199)	
	(16) 07:27:09	Roman.java	COMPILE	This method must return a result of type String	
	(17) 07:27:16	Roman.java	PRODUCTION EDIT	4sec MI=0(2), SI=+1(2), FI=+10(209)	
	(18) 07:27:35	TestIntegerToRoman.java	UNIT TEST	TEST FAILED	
	(19) 07:27:39	TestIntegerToRoman.java	BUFFTRANS	FROM Roman.java	
	(20) 07:28:05	TestIntegerToRoman.java	COMPILE	The method toString() is undefined for the type Roman	
	(21) 07:28:08	TestIntegerToRoman.java	UNIT TEST	TEST OK	
2	(1) 07:28:12	TestIntegerToRoman.java	TEST EDIT	0sec MI=0(2), SI=0(1), TI=0(1), AI=0(0), FI=+1(308)	<p>This portion of development appears to be TDD conformant because:</p> <p>Tests were written before production code.</p> <p>This episode looks like a typical test-first episode because:</p> <p>Some tests were added (2). Then a compilation error occurred (4). Then production code was edited (9). Then tests were run with failures (6). Then production code was again edited (9).</p>
	(2) 07:28:22	TestIntegerToRoman.java	ADD METHOD	void testOneReturnsI()	
	(3) 07:28:46	TestIntegerToRoman.java	TEST EDIT	4sec MI=+1(3), SI=0(1), TI=+1(2), AI=0(0), FI=+111(419)	
	(4) 07:28:46	TestIntegerToRoman.java	COMPILE	Syntax error, insert ";" to complete Statement	
	(5) 07:28:49	TestIntegerToRoman.java	TEST EDIT	0sec MI=0(3), SI=+2(2), TI=0(2), AI=+1(0), FI=+1(420)	
	(6) 07:28:56	TestIntegerToRoman.java	UNIT TEST	TEST FAILED	
	(7) 07:29:05	TestIntegerToRoman.java	BUFFTRANS	FROM TestStack.java	
	(8) 07:29:13	Roman.java	BUFFTRANS	FROM TestIntegerToRoman.java	
	(9) 07:31:15	Roman.java	PRODUCTION EDIT	62sec MI=0(2), SI=+1(3), FI=+44(253)	
	(10) 07:31:19	TestIntegerToRoman.java	UNIT TEST	TEST OK	
3	(1) 07:32:04	Roman.java	PRODUCTION EDIT	21sec MI=0(2), SI=0(3), FI=0(253)	<p>This portion of development appears to be TDD conformant because:</p> <p>Only tests were run, following a TDD-conformant episode.</p>
	(2) 07:32:06	TestIntegerToRoman.java	UNIT TEST	TEST OK	