JUNIT MOCK TEST

This section presents you various set of Mock Tests related to **JUnit Framework**. You can download these sample mock tests at your local machine and solve offline at your convenience. Every mock test is supplied with a mock test key to let you verify the final score and grade yourself.

JUNIT MOCK TEST III

Q 1 - Which of the following method of TestResult class gets the number of detected errors?

A - int getErrors

B - int errorCount

C - int countErrors

D - int getErrorCount

Q 2 - Which of the following method of TestResult class returns an Enumeration for the errors?

A - Enumeration<Object> errors

B - Enumeration<TestFailure> errors

C - Enumeration<TestFailure> getErrors

D - Enumeration<Object> getErrors

Q 3 - Which of the following method of TestResult class gets the number of detected failures?

A - int error

B - int errorCount

C - int failureCount

D - int failure

Q 4 - Which of the following method of TestResult class runs a TestCase?

A - void run*TestCasetest*

B - void execute*TestCasetest*

C - void runTest*TestCasetest*

D - void executeTest*TestCasetest*

Q 5 - Which of the following method of TestResult class gets the number of run tests?

A - int testCount

B - int runCount

C - int countExecutions

D - int countRun

Q 6 - Which of the following method of TestResult class informs the result that a test will be started?

A - void startTest*Testtest*

B - void start*Testtest*

C - void execute*Testtest*

D - void executeTest*Testtest*

Q 7 - Which of the following method of TestResult class marks that the test run should stop?

A - void stopTest

B - void stop

C - void stopTest*Testtest*

D - void breakTest*Testtest*

Q 8 - Which of the following method of TestSuite class adds a test to the suite?

A - void add*Test*

B - void add

C - void addTest*Testtest*

D - void addTestCase*Testtest*

Q 9 - Which of the following method of TestSuite class adds the tests from the given class to the suite?

A - void addTest*Class* < ?*extendsTestCase* > *testClass*

B - void addTestSuite*Class* < ?*extendsTestCase* > *testClass*

C - void addTest*TestCasetestClass*

D - void addTestCase*TestCasetestClass*

Q 10 - Which of the following method of TestSuite class counts the number of test cases that will be run by this test?

A - int countTestExecutions

B - int countTest

C - int count

D - int countTestCases

Q 11 - Which of the following method of TestSuite class returns the name of the suite?

A - String getNameOfTestSuite

B - String getTestName

C - String getName

D - String getTestSuite

Q 12 - Which of the following method of TestSuite class runs the tests and collects their result in a TestResult?

A - void execute*TestResultresult*

B - void runTest*TestResultresult*

C - void run*TestResultresult*

D - void executeTest*TestResultresult*

Q 13 - Which of the following method of TestSuite class sets the name of the suite?

A - void setSuiteName*Stringname*

B - void setName*Stringname*

C - void setTestSuiteName*Stringname*

D - void setTestName*Stringname*

Q 14 - Which of the following method of TestSuite class returns the test at the given index?

A - Test testAt*intindex*

B - Test testAtIndex*intindex*

C - Test getTestAt*intindex*

D - Test test*intindex*

Q 15 - Which of the following method of TestSuite class returns the number of tests in this suite?

A - int testCount

B - int testCaseCount

C - int getTestCount

D - int testSuiteCount

Q 16 - Which of the following method of TestSuite class returns a test which will fail and log a warning message?

A - static Test displayWarning*Stringmessage*

B - static Test showWarning*Stringmessage*

C - static Test getWarning*Stringmessage*

D - static Test warning*Stringmessage*

Q 17 - The @Test annotation tells JUnit that the public void method to which it is attached can be run as a test case.

A - true

B - false

Q 18 - Annotating a public void method with @Before causes that method to be run before each Test method.

A - false

B - true

Q 19 - Annotating a public void method with @After causes that method to be run after each Test method.

A - false

B - true

Q 20 - Annotating a public void method with @BeforeClass causes that method to be run before each Test method.

A - false

B - true

Q 21 - Annotating a public void method with @AfterClass causes that method to be run after each Test method.

A - false

B - true

Q 22 - The @Ignore annotation is used to ignore the test and that test will not be executed.

A - false

B - true

Q 23 - Which of the following class is used to run test cases?

A - JUnitCore

B - TestCase

C - TestSuite

D - TestResult

Q 24 - Which of the following class is used to bundle unit test cases and run them together?

A - JUnitCore

B - TestCase

C - TestSuite

D - TestResult

Q 25 - @RunWith and @Suite annotation are used to run the suite test.

A - false

B - true

ANSWER SHEET

Question Number	Answer Key
1	B
2	C
3	C
4	A
5	B
6	A
7	B
8	C
9	B
10	D
11	C

12	C
13	B
14	A
15	A
16	D
17	A
18	B
19	B
20	A
21	A
22	B
23	A
24	C
25	B

Processing math: 100%