Contents | Filter |

Class Jtest2

java.lang.Object
    Jtesting.Jtest2

```
public class Jtest2
extends Object
```

JUnit test class for testing database-related operations such as adding, modifying, and deleting questions and answers.

Constructor Summary

Constructors

Constructor	Description
Jtest2()	Default constructor for Jtest2.

Method Summary

All Methods   Instance Methods

Concrete Methods

Modifier and Type	Method	Description
void	**setUp**()	Sets up the database connection and initializes objects before each test.
void	**tearDown**()	Closes the database connection after each test.
void	**testAddQuestion**()	Test to see you user can add and retrieve a question
void	**testDeleteAnswer**(	Test to see if the user can delete answers

void	**testDeleteQuestio**	Test to delete questions
void	**testModifyAnswer(**	Test modify existing answers
void	**testQuestionValid**	Test to see if the question field was left blank

Methods inherited from class java.lang.**Object**

equals , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait

Constructor Details

Jtest2

```
public Jtest2()
```

Default constructor for Jtest2. Initializes the test class for JUnit execution.

Method Details

setUp

```
@BeforeEach
public void setUp()
          throws SQLException
```

Sets up the database connection and initializes objects before each test.

Throws:

   SQLException   - if a database connection error occurs.

tearDown

```
@AfterEach
public void tearDown()
```

Closes the database connection after each test.

testDeleteQuestion

```
@Test
@DisplayName("Delete question from database")
public void testDeleteQuestion()
                              throws SQLException
```

Test to delete questions

Throws:

   SQLException - if a database access error occurs.

testModifyAnswer

```
@Test
@DisplayName("Modify answer")
public void testModifyAnswer()
```

Test modify existing answers

testQuestionValidationEmptyContent

```
@Test
@DisplayName("Empty question should be invalid")
public void testQuestionValidationEmptyContent()
```

Test to see if the question field was left blank

testAddQuestion

```
@Test
@DisplayName("Add and retrieve question")
public void testAddQuestion()
```

Test to see you user can add and retrieve a question

testDeleteAnswer

```
@Test
@DisplayName("Delete answer")
public void testDeleteAnswer()
```

Test to see if the user can delete answers