

ID	24
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	init() Alle vorhandenen Tabellen werden gelöscht und neu erstellt
Vorbedingung	Datenbank ist aktiv
Eingaben	-
Erwartetes Ergebnis	Tabelleninhalt von der Tabelle <i>GENERATIONS</i> kann abgerufen werden
Ergebnis	Tabelleninhalt von der Tabelle <i>GENERATIONS</i> kann abgerufen werden Test erfolgreich

ID	25
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	update(String sqlStatement) Ein gegebenes SQL-Statement wird ausgeführt
Vorbedingung	Datenbank ist aktiv
Eingaben	"INSERT INTO GENERATIONS (id , generation, route , distance) " + "VALUES ("+ 1 + " , "+ 1 + " , "+ "0,1,0" + " , "+ 2 + ") "
Erwartetes Ergebnis	Tabelle <i>Generations</i> hat eine Zeile
Ergebnis	Tabelle <i>Generations</i> hat eine Zeile Test erfolgreich

ID	26
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	select(String sqlStatement): ResultSet Ein gegebenes SQL-Statement wird ausgeführt und das Resultat zurückgegeben
Vorbedingung	Datenbank ist aktiv Tabelle <i>Generations</i> hat eine Zeile mit vordefinierten Daten
Eingaben	"SELECT * FROM GENERATIONS"
Erwartetes Ergebnis	Integer in der Spalte <i>generation</i> ist 1 String in der Spalte <i>route</i> ist 0,1,0 Integer in der Spalte <i>distance</i> ist 2
Ergebnis	Integer in der Spalte <i>generation</i> ist 1 String in der Spalte <i>route</i> ist 0,1,0 Integer in der Spalte <i>distance</i> ist 2 Test erfolgreich

ID	27
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	getTable(String table): ResultSet Eine komplette Tabelle wird abgerufen und zurückgegeben
Vorbedingung	Datenbank ist aktiv
Eingaben	"GENERATIONS"
Erwartetes Ergebnis	Tabelle <i>GENERATIONS</i> hat 4 Spalten
Ergebnis	Tabelle <i>GENERATIONS</i> hat 4 Spalten Test erfolgreich

ID	28
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	dropTable() Alle vorhandenen Tabellen werden gelöscht
Vorbedingung	Datenbank ist aktiv Tabelle <i>GENERATIONS</i> ist vorhanden
Eingaben	-
Erwartetes Ergebnis	Tabelle <i>GENERATIONS</i> ist nicht mehr vorhanden (NullPointerException muss geworfen werden)
Ergebnis	NullPointerException wird geworfen Test erfolgreich

ID	29
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	createTable() Es werden alle Tabellen neu erstellt
Vorbedingung	Datenbank ist aktiv
Eingaben	-
Erwartetes Ergebnis	Tabelle <i>GENERATIONS</i> hat 4 Spalten Tabelle <i>HISTORY</i> hat 4 Spalten Tabelle <i>TEST</i> existiert nicht
Ergebnis	Tabelle <i>GENERATIONS</i> hat 4 Spalten Tabelle <i>HISTORY</i> hat 4 Spalten Tabelle <i>TEST</i> existiert nicht Test erfolgreich

ID	30
Verfahren	Parametrisierter JUnit-Test
Klasse	HSQLDBManager
Methoden	updateTable(String table, int generation, String route, int distance) Auf eine gegebene Tabelle wird ein INSERT mit den gegebenen Daten ausgeführt
Vorbedingung	Datenbank ist aktiv
Eingaben	"GENERATIONS", 1, „0,1,0“, 2
Erwartetes Ergebnis	Tabelle <i>GENERATIONS</i> hat eine Zeile
Ergebnis	Tabelle <i>GENERATIONS</i> hat eine Zeile Test erfolgreich