```java
package DBLayer;

import static org.junit.Assert.*;
import java.util.ArrayList;
import org.junit.Test;
import ModelLayer.Employee;

public class DBEmployeeTest {

	@Test
	public void testGetAllEmployees() {
		DBEmployee emp = new DBEmployee();
		ArrayList<Employee> mylist = emp.getAllEmployees(false);
		if(mylist.size()== 5)
		{
			System.out.println("Working");

		}

		else
		{
			fail("Error in getAllEmployees ");
		}

	}

	@Test
	public void testFindEmployeeByFname() {
		DBEmployee dbEmp = new DBEmployee();
		Employee emp = dbEmp.findEmployeeByFname("Pilatos", false);
		if(emp != null)
		{
			System.out.println("Returned an employee with name " +
			emp.getFname());
			System.out.println("Working");
		}
		else
		{
		fail("could not return an employee with this name.");
		}
	}

	@Test
	public void testFindEmployeeByLname() {
		DBEmployee dbEmp = new DBEmployee();
		Employee emp = dbEmp.findEmployeeByLname("Andeby", false);
		if(emp != null)
		{
			System.out.println("Returned an employee with name " +
			emp.getLname());
			System.out.println("Working");
		}
		else
		{
		fail("could not return an employee with this name.");
		}
	}

	@Test
	public void testFindEmployeeID() {
		DBEmployee dbEmp = new DBEmployee();
		Employee emp = dbEmp.findEmployeeByID(3003, true);
		if(emp != null)
		{
```

```
65                System.out.println("The employee's name is: " + emp.
66                getFname() + " and his telephone number is: " + emp.
67                getPhoneNo());
68          }
69          else
70          {
71          fail("failed to search with an ID.");
72          }
73    }
74
75    @Test
76    public void testInsertEmployee() {
77          DBEmployee dbEmp = new DBEmployee();
78          //TODO !!! redefine the constructor of Employee class.
79          Employee testEmp = new Employee(3006,"Frodo", "Bagins",
80          "Shire Ave 342", 6558, "The Shire", 88339955, "hobbit@lame.
81          com");
82
83          try{
84                int x = dbEmp.insertEmployee(null);
85
86          if(x > 0)
87          {
88                Employee emp = dbEmp.findEmployeeByID(3006, false);
89                System.out.println("A new employee has been created with
90                the following values: " + "Employee ID: " + emp.
91                getEmployeeID() + "First name: " + emp.getFname() + "Last
92                name: " + emp.getLname() + "lives at: " + emp.getAddress()
93                + "zipcode: " + emp.getZipCode() + "city: " + emp.getCity(
94                ) + "Phone number: " + emp.getPhoneNo() + "Email: " + emp.
95                getEmail());
96          }
97          else
98          {
99          fail("Nothing new was created FAILS");
100         }
101   }
102         catch(Exception e)
103         {
104               System.out.println("Nothing new was created");
105         }
106   }
107
108 //      @Test
109 //      public void testUpdateEmployee() {
110 //            fail("Not yet implemented");
111 //      }
112 //
113 //      @Test
114 //      public void testDeleteEmployee() {
115 //            fail("Not yet implemented");
116 //      }
117
118 }
```