

```

package test;

import hash_table.Hash_Table;
import junit.framework.TestCase;
import junit.framework.TestResult;

import org.junit.Test;

import agac.Node;
import agaclar.Vector_Agac;
import test.Herbalist_App;
import veri_tabani.Bitki_veri_tabani;
import veri_tabani.Hastalik_veri_tabani;
import veri_tabani.Kategori_veri_tabani;

public class TestHerbalist extends TestCase {
    private static Hash_Table hash_table;

    @Test
    public void test() {
        run();
        assertEquals("Bulundu",
            hash_table.hastalik_hash_table_de_var_mi("Anemi"));
    }

    Test decoratedSuite;

    void AbstractSetupDecorator(Test testSuiteToDecorate) {
        decoratedSuite = testSuiteToDecorate;
    }

    public void setUp() {
        // subclass responsibility
        veri_tabanlarini_olustur();
        try {
            //veri_tabanlarini_veri_yapilarina_kopyala();
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public static void veri_tabanlarini_olustur() {
        setKategori_veri_tabani(new Kategori_veri_tabani("kategori.txt"));
        setBitki_veri_tabani(new Bitki_veri_tabani("bitki.txt"));
        setHastalik_veri_tabani(new Hastalik_veri_tabani("hastalik.txt"));
    }

    @SuppressWarnings("unused")
    private static void veri_tabanlarini_veri_yapilarina_kopyala()
        throws Exception {
        kategorilere_gore_agac_vektorunu_olustur();
        bitki_veri_tabanini_agac_vektorene_kopyala();
        hastalik_veri_tabanini_hash_table_ye_kopyala();
    }

    private static void kategorilere_gore_agac_vektorunu_olustur() {
        setAgac_vektoru(new Vector_Agac(Kategori_veri_tabani.getKategoriler()
            .size() + 1));
        // Son vektor elemanı Kategorisiz olan bitkiler için
    }
}

```

```

public static void bitki_veri_tabanini_agac_vektorne_kopyala()
    throws Exception {
    int aranan, endex;
    Node newnode;
    for (int i = 0; i < getBitki_veri_tabani().getBitkiler().size(); i++) {
        newnode = new Node(getBitki_veri_tabani().getBitkiler()
            .elementAt(i));
        aranan = newnode.getBitki().getKategori();
        endex =
getKategori_veri_tabani().bitkinin_kategorisini_bul(aranan);
        getAgac_vektoru().getAgaclar().elementAt(endex).insert(newnode);
    }
}

private static void hastalik_veri_tabanini_hash_table_ye_kopyala() {
    hash_table = new Hash_Table(Hastalik_veri_tabani.getHastaliklar());
}

public static void setKategori_veri_tabani(
    Kategori_veri_tabani kategori_veri_tabani) {
    Herbalist_App.setKategori_veri_tabani(kategori_veri_tabani);
}

public static Kategori_veri_tabani getKategori_veri_tabani() {
    return getKategori_veri_tabani();
}

public static void setBitki_veri_tabani(Bitki_veri_tabani bitki_veri_tabani) {
    Herbalist_App.setBitki_veri_tabani(bitki_veri_tabani);
}

public static Bitki_veri_tabani getBitki_veri_tabani() throws Exception {
    return getBitki_veri_tabani();
}

public static void setAgac_vektoru(Vector_Agac agac_vektoru) {
    Herbalist_App.setAgac_vektoru(agac_vektoru);
}

public static Vector_Agac getAgac_vektoru() {
    return getAgac_vektoru();
}

public static void setHash_table(Hash_Table hash_table) {
    Herbalist_App.setHash_table(hash_table);
}

public static Hash_Table getHash_table() {
    return hash_table;
}

public static void setHastalik_veri_tabani(
    Hastalik_veri_tabani hastalik_veri_tabani) {
    Herbalist_App.setHastalik_veri_tabani(hastalik_veri_tabani);
}

public static Hastalik_veri_tabani getHastalik_veri_tabani() {
    return getHastalik_veri_tabani();
}

public void tearDown() throws Exception {
    // subclass responsibility

```

```
}

public TestResult run() {
    try {
        setUp();
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    ((TestCase) decoratedSuite).run();// casting to TestHerbalist
    try {
        tearDown();
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return null;
}
}
```