Giovanni Acireale

Assignment #7

// Abstract class that must be extended not instantiated

public abstract class Plant

{

// Two instance variables that every type of plant would use Age and Height

private int age = 0;

private int height = 0;

public void setHeight(int height) {

this.height = height;

}

public int getAge() {

return age;

}

public int getHeight() {

return height;

}

public void addYearToAge() {

age++;

}

/\*\* Each abstract method represents the actions that a plant must take during the specified season. These actions are specific to the type of plant, therefore cannot be generalized\*/

// Any class extends the Plant class must implement these abstract methods.

abstract public void doSpring();

abstract public void doSummer();

abstract public void doFall();

abstract public void doWinter();

} // End of Plant class

public class MapleTree extends Plant {

private static final int AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON = 2;

private void grow() {

int currentHeight = getHeight();

setHeight(currentHeight + AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON);

}

@Override

public void doSpring() {

grow();

addYearToAge();

System.out.println("Spring: The maple tree is starting to grow leaves and new branches");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doSummer() {

grow();

System.out.println("Summer: The maple tree is continuing to grow");

System.out.println("\tCurrentAge: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doFall() {

System.out.println("Fall: The maple tree has stopped growing and is losing its leaves");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doWinter() {

System.out.println("Winter: The maple tree is dormant");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

} // end of MapleTree class

public class Tulip extends Plant {

private static final int AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON = 1;

private void grow() {

int currentHeight = getHeight();

setHeight(currentHeight + AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON);

}

private void dieDownForWinter() {

setHeight(0);

}

@Override

public void doSpring() {

grow();

addYearToAge();

System.out.println("Spring: The tulip is starting to grow up from the ground");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doSummer() {

System.out.println("Summer: The tulip has stopped to growing and is flowering");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doFall() {

System.out.println("Fall: The tulip begins to wilt");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doWinter() {

dieDownForWinter();

System.out.println("Winter: The tulip is dormant underground");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

}// end of Tulip class

public class Rose extends Plant

{

private static final int AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON = 1;

private void grow()

{

int currentHeight = getHeight();

setHeight(currentHeight + AMOUNT\_TO\_GROW\_IN\_ONE\_GROWING\_SEASON);

}

@Override

public void doSpring()

{

grow();

addYearToAge();

System.out.println("Spring: The rose tree is starting to grow leaves and new branches");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doSummer()

{

grow();

System.out.println("Summer: The rose tree is continuing to grow");

System.out.println("\tCurrentAge: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doFall()

{

System.out.println("Fall: The rose tree has stopped growing and is losing its leaves");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

@Override

public void doWinter()

{

System.out.println("Winter: The rose tree is dormant");

System.out.println("\tCurrent Age: " + getAge() + " " + "Current Height: " + getHeight());

}

}

public class Simulator

{

public static void main(String[] args)

{

System.out.println("Creating a maple tree, tulip, and rose tree...");

MapleTree mapleTree = new MapleTree();

Tulip tulip = new Tulip();

Rose rose = new Rose();

System.out.println("Entering a loop to simulate 3 years");

for (int i = 0; i < 3; i++)

{

mapleTree.doSpring();

tulip.doSpring();

rose.doSpring();

mapleTree.doSummer();

tulip.doSummer();

rose.doSummer();

mapleTree.doFall();

tulip.doFall();

rose.doFall();

mapleTree.doWinter();

tulip.doWinter();

rose.doWinter();

}

}

}

