Is a, Has a, Uses a relationship

1. **Is a Relation:**
   It can be demonstrated in inheritance.
   In this the child class shares common characteristics with the parent class.

   DEMONSTRATION:

```
class Plant {
    void absorb() {
        System.out.println("Plant is absorbing nutrients");
    }
}

class Tree extends Plant {
    void sway() {
        System.out.println("Tree is swaying in the breeze");
    }
}

public class Main {
    public static void main(String[] args) {
        Tree sapling = new Tree();
        sapling.absorb(); // Inherited from Plant
        sapling.sway();
    }
}
```

2. **Has a Relationship**
   When one class contains an instance of another class it is said to have a has a relationship.

   DEMONSTRATION:

```
class Sun {
    void shine() {
        System.out.println("Sun is shining");
    }
}


class Flower {
    private Sun sunlight;

    Flower() {
        this.sunlight = new Sun();
    }

    void bloom() {
        sunlight.shine();
        System.out.println("Flower bloomed");
    }
}


public class Main {
    public static void main(String[] args) {
        Flower myFlower = new Flower();
        myFlower.bloom();
    }
}
```

3. **Uses a Relationship:**
   one class uses the functionality of another class without being a part of it, it is referred to as a uses a relationship.

   DEMONSTRATION:

```
class Calculator {
    // Method to add two numbers
    int add(int a, int b) {
        return a + b;
    }
}

class MathStudent {

    void performAddition(int x, int y, Calculator calculator) {
        int result = calculator.add(x, y);
        System.out.println("The result of addition is: " + result);
    }
}

public class Main {
    public static void main(String[] args) {
         Calculator calculator = new Calculator();
        MathStudent student = new MathStudent();


        student.performAddition(5, 3, calculator);
    }
}
```