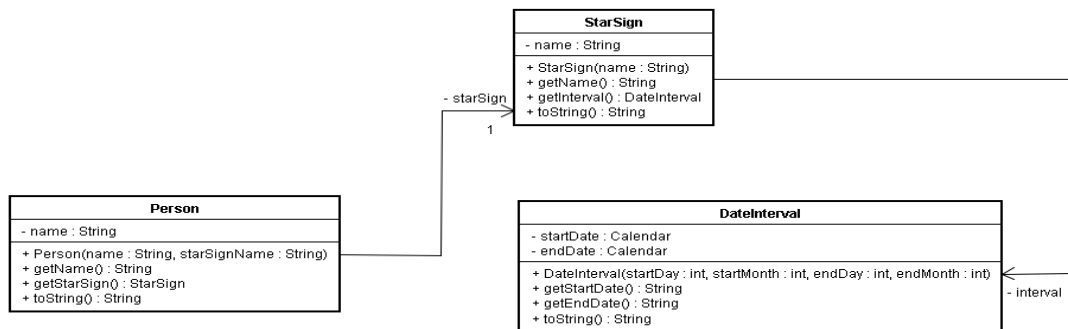
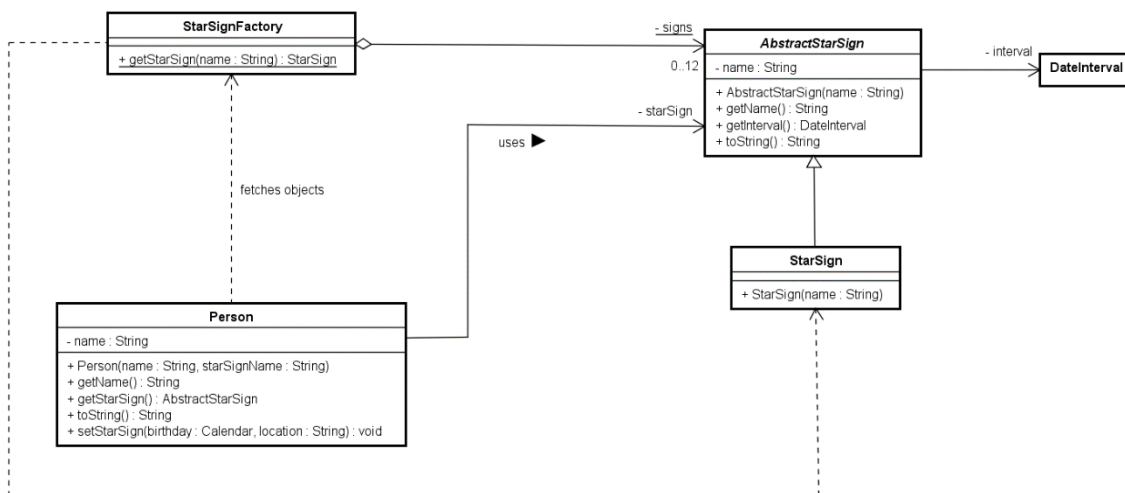


Exercise 12.02 – Flyweight

The StarSign example from the slide is given below as a class diagram (java code can be found in appendix).

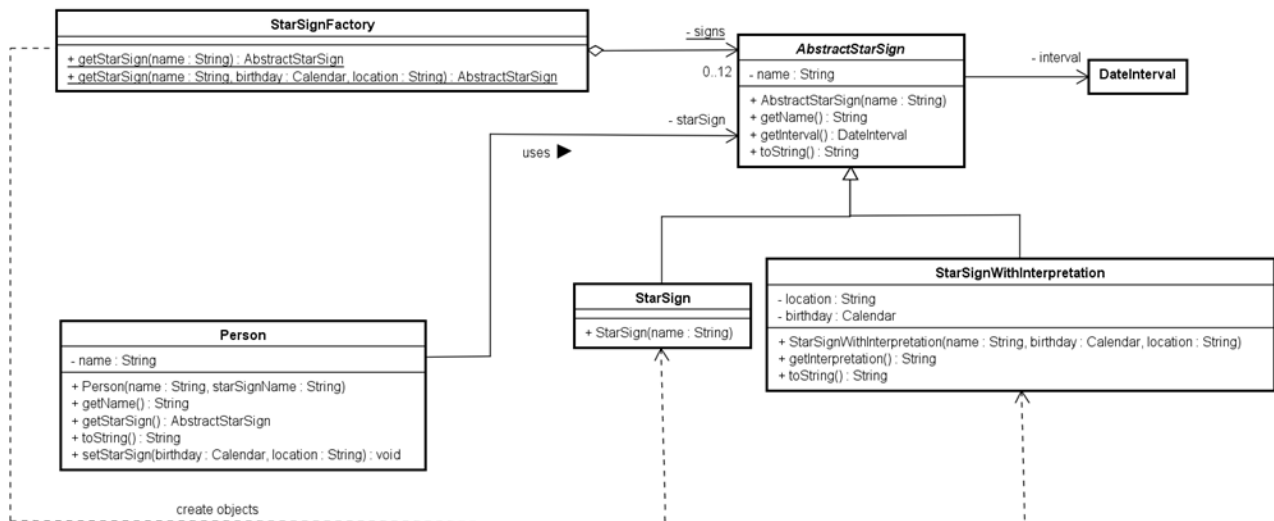


Use this program as a template to implement the Flyweight Design Pattern (without unshared objects) as shown in the diagram below. (The factory uses a HashMap or another collection after your choice)



Exercise 12.03 – Flyweight with shared and unshared objects

Change the previous exercise such that you implement the Flyweight design pattern with shared and unshared objects. The unshared objects are from the subclass `StarSignWithInterpretation`. Try to figure out how to implement this change (an interpretation is unique for a person with same birthday, time and location – in this exercise just return strings after your choice)



Appendix: Classes related to exercise 12.02:

Class DateInterval:

```
import java.util.Calendar;
import java.util.GregorianCalendar;

public class DateInterval
{
    private Calendar startDate;
    private Calendar endDate;

    public DateInterval(int startDay, int startMonth, int endDay, int endMonth)
    {
        this.startDate = new GregorianCalendar(0, startMonth - 1, startDay);
        this.endDate = new GregorianCalendar(0, endMonth - 1, endDay);
    }

    private static String getCalendarDate(Calendar calendarDate)
    {
        return (calendarDate.get(Calendar.DAY_OF_MONTH) + "/"
            + (calendarDate.get(Calendar.MONTH) + 1));
    }

    public String getStartDate()
    {
        return getCalendarDate(startDate);
    }

    public String getEndDate()
    {
        return getCalendarDate(endDate);
    }

    public String toString()
    {
        return getCalendarDate(startDate) + " - " + getCalendarDate(endDate);
    }
}
```

Class StarSign:

```
public class StarSign
{
    private String name;
    private DateInterval interval;

    // the following class variable is not shown in the class diagram:
    public static final String[] LEGAL_NAMES =
    { "Aries", "Taurus", "Gemini", "Cancer", "Leo", "Virgo", "Libra", "Scorpio",
        "Sagittarius", "Capricorn", "Aquarius", "Pisces" };
}
```

```

public StarSign(String name)
{
    this.name = Character.toUpperCase(name.charAt(0))
        + name.substring(1).toLowerCase();
    setDateInterval();
}

public String getName()
{
    return name;
}

public DateInterval getInterval()
{
    return interval;
}

private void setDateInterval()
{
    if (name.equals("Aries"))
        interval = new DateInterval(21, 3, 19, 4);
    else if (name.equals("Taurus"))
        interval = new DateInterval(20, 4, 20, 5);
    else if (name.equals("Gemini"))
        interval = new DateInterval(21, 5, 20, 6);
    else if (name.equals("Cancer"))
        interval = new DateInterval(21, 6, 22, 7);
    else if (name.equals("Leo"))
        interval = new DateInterval(23, 7, 22, 8);
    else if (name.equals("Virgo"))
        interval = new DateInterval(23, 8, 22, 9);
    else if (name.equals("Libra"))
        interval = new DateInterval(23, 9, 22, 10);
    else if (name.equals("Scorpio"))
        interval = new DateInterval(22, 11, 21, 12);
    else if (name.equals("Sagittarius"))
        interval = new DateInterval(20, 4, 20, 5);
    else if (name.equals("Capricorn"))
        interval = new DateInterval(22, 12, 19, 1);
    else if (name.equals("Aquarius"))
        interval = new DateInterval(20, 1, 18, 2);
    else if (name.equals("Pisces"))
        interval = new DateInterval(19, 2, 20, 3);
    else
        interval = new DateInterval(0, 0, 0, 0);
}

public String toString()
{
    return getName() + " (" + interval + ")";
}
}

```

Class Person:

```
public class Person
{
    private String name;
    private StarSign starSign;

    public Person(String name, StarSign starSign)
    {
        this.name = name;
        this.starSign = starSign;
    }
    public String getName()
    {
        return name;
    }
    public StarSign getStarSign()
    {
        return starSign;
    }
    public String toString()
    {
        return name + ", " + starSign;
    }
}
```

Class TestOfStarSign (test class):

```
public class TestOfStarSign
{
    public static void main(String[] args)
    {
        String[] legalStarSignNames = StarSign.LEGAL_NAMES;
        System.out.println("Legal star sign names:");
        for (int i=0; i<legalStarSignNames.length; i++)
            System.out.println(legalStarSignNames[i]);
        System.out.println();

        Person[] list = new Person[10];
        list[0] = new Person("Allan", new StarSign("Leo"));
        list[1] = new Person("Bob", new StarSign("Scorpio"));
        list[2] = new Person("Carl", new StarSign("Leo"));
        list[3] = new Person("Dennis", new StarSign("Sagittarius"));
        list[4] = new Person("Eric", new StarSign("Leo"));
        list[5] = new Person("Franc", new StarSign("Leo"));
        list[6] = new Person("Graham", new StarSign("Sagittarius"));
        list[7] = new Person("Hans", new StarSign("Leo"));
        list[8] = new Person("Igor", new StarSign("Virgo"));
        list[9] = new Person("Jens", new StarSign(legalStarSignNames[8]));

        System.out.println("Person objects:");
        for (int i=0; i< list.length; i++)
        {
            System.out.printf("%-40s %d\n", list[i], list[i].getStarSign().hashCode());
        }
    }
}
```

```
}  
}  
  
/* OUTPUT: (note hash codes could be different)  
Legal star sign names:  
Aries  
Taurus  
Gemini  
Cancer  
Leo  
Virgo  
Libra  
Scorpio  
Sagittarius  
Capricorn  
Aquarius  
Pisces  
  
Person objects:  
Allan, Leo (23/7 - 22/8)           1641745  
Bob, Scorpio (22/11 - 21/12)      9023134  
Carl, Leo (23/7 - 22/8)           19336051  
Dennis, Sagittarius (30/11 - 30/11) 6336176  
Eric, Leo (23/7 - 22/8)           23899971  
Franc, Leo (23/7 - 22/8)           6718604  
Graham, Sagittarius (20/4 - 20/5)   8918002  
Hans, Leo (23/7 - 22/8)            30771886  
Igor, Virgo (23/8 - 22/9)          8637543  
Jens, Sagittarius (20/4 - 20/5)     14718739  
*/
```