Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №8

З предмету: «Інженерія програмного забезпечення»

номер залікової книжки 2230

Виконав: студент групи ІО-22

Щербина М.В.

Київ 2013р.

1. Повторити шаблони поведінки для проектування ПЗ. Знати загальну характеристику шаблонів поведінки та призначення кожного з них.

2. Детально вивчити шаблони поведінки для проектування ПЗ Prototype, Singleton та Factory Method

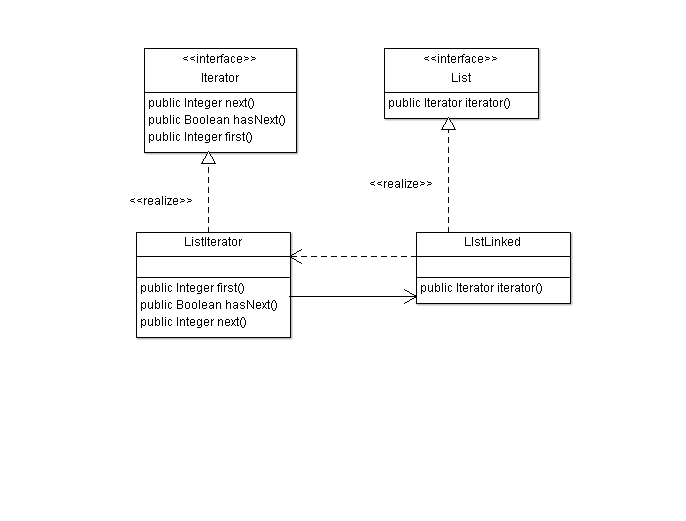
. Для кожного з них:

* вивчити Шаблон, його призначення, альтернативні назви, мотивацію, випадки коли його застосування є доцільним та результати такого застосування;
* знати особливості реалізації Шаблону, споріднені шаблони, відомі випадки його застосування в програмних додатках;
* вільно володіти структурою Шаблону, призначенням його класів та відносинами між ними;
* вміти розпізнавати Шаблон в UML діаграмі класів та будувати сирцеві коди Java-класів, що реалізують шаблон.

3. В підготованому проекті (ЛР1) створити програмний пакет com.lab111.labwork8. В пакеті розробити інтерфейси і класи, що реалізують завдання (згідно варіанту) з застосуванням одного чи декількох шаблонів (п.2). В розроблюваних класах повністю реалізувати методи, пов'язані з функціюванням Шаблону. Методи, що реалізують бізнес-логіку закрити заглушками з виводом на консоль інформації про викликаний метод та його аргументи.

4. За допомогою автоматизованих засобів виконати повне документування розроблених класів (також методів і полів), при цьому документація має в достатній мірі висвітлювати роль певного класу в загальній структурі Шаблону та особливості конкретної реалізації.

Варіант 8

Визначити специфікації класів, які інкапсулюють лінійний список цілих чисел та ітератор послідовного обходу у прямому напрямку в упорядкованій структурі.

**package** com.lab111;

**import** com.lab111.labwork8.\*;

**public** **final** **class** TestMain {

/\*\*

\* Constructor.

\*

\*/

**private** TestMain() {

**super**();

}

/\*\*

\* Invokes at application startup.

\*

\* **@param** args

\* Parameters from command line

\*/

**public** **static** **void** main(**final** String[] args) {

ListLinked list = **new** ListLinked();

Iterator iterator = list.iterator();

iterator.next();

}

}

**package** com.lab111.labwork8;

**public** **interface** Iterator {

**public** Integer next();

**public** Boolean hasNext();

**public** Integer first();

}

**package** com.lab111.labwork8;

**public** **interface** List {

**public** Iterator iterator();

}

**package** com.lab111.labwork8;

**public** **class** ListIterator **implements** Iterator {

**public** ListIterator() {

System.*out*.println("Создание итератора");

}

**public** ListLinked listLinked;

**public** Integer first() {

System.*out*.println("ListIterator.first");

**return** 0;

}

**public** Boolean hasNext() {

System.*out*.println("ListIterator.hasNext");

**return** **null**;

}

**public** Integer next() {

System.*out*.println("ListIterator.next");

**return** **null**;

}

}

**package** com.lab111.labwork8;

**public** **class** ListLinked **implements** List {

**public** Iterator iterator() {

**return** **new** ListIterator();

}

}