Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №2

З предмету: «Інженерія програмного забезпечення»

Виконав: студент групи ІО-22

Щербина М.В.

Перевірив: Абу Усбах А.Н.

Київ 2013р.

**Варіант**

Генералізація = 7

If3 <- If1

If1 <- If2

Cl3 <- Cl1

Агрегація = 0

If1 <- Cl1

Cl3 <- Cl2

Cl3 <- Cl3

**Завдання**

1. Ознайомитись з призначенням та видами діаграм мови UML. Вивчити діаграму класів, вільно володіти елементами та відношеннями між ними. Вміти будувати діаграми класів для сирцевого коду Java, а також генерувати сирцевий код еквівалентний заданій діаграмі класів.

2. Побудувати діаграму класів, що містить три інтерфейси If1, If2, If3 з методами meth1(), meth2(), meth3 та класи що їх реалізують Cl1, Cl2, Cl3 відповідно.

3. Згідно варіанту (нижче) реалізувати на діаграмі класів відношення генералізації та агрегації.

4. В підготованому проекті (ЛР1) створити програмний пакет com.lab111.labwork2.

В пакеті розробити інтерфейси і класи згідно діаграмі (п.3-4). Реалізація методів має виводити на консоль ім'я класу та назву методу).

5. Ознайомитись з засобами автоматизації UML-моделювання. Вміти використовувати середовища ArgoUML та Umbrello на базовому рівні для розробки діаграми класів та документування програмного забезпечення.

6. За допомогою середовища ArgoUML або Umbrello імпортувати сирцеві коди пакету com.lab111.labwork2 та перевірити відповідність побудованої діаграми класів з розробленою (п.3-4). Зберегти діаграму в каталозі документації проекту.

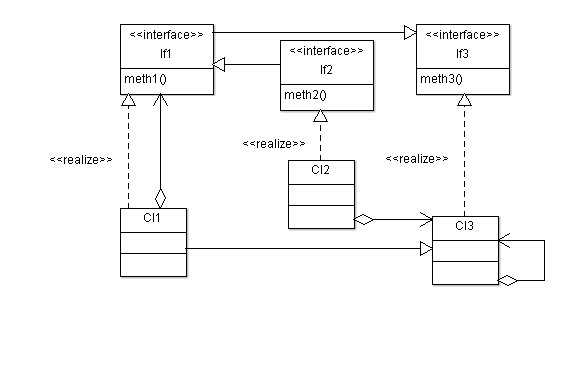
7. Ознайомитись з синтаксисом коментарів для засобу автоматизації

документації JavaDoc. Модифікувати сирцеві коди пакету com.lab111.labwork2 додавши коментарі у форматі JavaDoc.

8. Згенерувати JavaDoc за допомогою Eclipse (меню Project) у каталог документації проекту.

9. Розробити ціль ANT для генерації JavaDoc. Згенерувати JavaDoc за допомогою розробленої цілі ANT.

Діаграма класів



Код програми

**package** com.lab111.labwork2;

**import** java.util.Vector;

/\*\*

\* Class 1

\*/

**public** **class** Cl1 **extends** Cl3 **implements** If1 {

**public** Vector myIf1;

@Override

**public** **void** meth3() {

System.*out*.println("Cl1.meth3()");

}

@Override

**public** **void** meth1() {

System.*out*.println("Cl1.meth1()");

}

}

**package** com.lab111.labwork2;

**import** java.util.Vector;

/\*\*

\* Class 2

\*/

**public** **class** Cl2 **implements** If2 {

**public** Vector myCl3;

@Override

**public** **void** meth1() {

System.*out*.println("Cl2.meth1()");

}

@Override

**public** **void** meth3() {

System.*out*.println("Cl2.meth3()");

}

@Override

**public** **void** meth2() {

System.*out*.println("Cl2.meth2()");

}

}

**package** com.lab111.labwork2;

**import** java.util.Vector;

/\*\*

\* Class 3

\*/

**public** **class** Cl3 **implements** If3 {

**public** Vector myCl3;

@Override

**public** **void** meth3() {

System.*out*.println("Cl3.meth3()");

}

}

**package** com.lab111.labwork2;

/\*\*

\* Interface 1

\*/

**public** **interface** If1 **extends** If3 {

/\* {author=Nikita, version=1.0} \*/

**public** **void** meth1();

}

**package** com.lab111.labwork2;

/\*\*

\* Interface 2

\*/

**public** **interface** If2 **extends** If1 {

/\* {author=Nikita} \*/

**public** **void** meth2();

}

**package** com.lab111.labwork2;

/\*\*

\* Interface 3

\*/

**public** **interface** If3 {

/\* {author=Nikita} \*/

**public** **void** meth3();

}

com.lab111.labwork2

**Class Cl1**

* java.lang.Object
  + [com.lab111.labwork2.Cl3](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html)
    - com.lab111.labwork2.Cl1
* **All Implemented Interfaces:**

[If1](file:///C:\template\doc\ant\com\lab111\labwork2\If1.html), [If3](file:///C:\template\doc\ant\com\lab111\labwork2\If3.html)

public class **Cl1**

extends [Cl3](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html)

implements [If1](file:///C:\template\doc\ant\com\lab111\labwork2\If1.html)

Class 1

* + **Field Summary**

|  |  |
| --- | --- |
| **Fields** | |
| **Modifier and Type** | **Field and Description** |
| java.util.Vector | [**myIf1**](file:///C:\template\doc\ant\com\lab111\labwork2\Cl1.html#myIf1) |

* + - **Fields inherited from class com.lab111.labwork2.**[**Cl3**](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html)

[myCl3](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html#myCl3)

* + **Constructor Summary**

|  |
| --- |
| **Constructors** |
| **Constructor and Description** |
| [**Cl1**](file:///C:\template\doc\ant\com\lab111\labwork2\Cl1.html#Cl1())() |

* + **Method Summary**

|  |  |
| --- | --- |
| **Methods** | |
| **Modifier and Type** | **Method and Description** |
| void | [**meth1**](file:///C:\template\doc\ant\com\lab111\labwork2\Cl1.html#meth1())() |
| void | [**meth3**](file:///C:\template\doc\ant\com\lab111\labwork2\Cl1.html#meth3())() |

* + - **Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

* + **Field Detail**
    - **myIf1**

public java.util.Vector myIf1

* + **Constructor Detail**
    - **Cl1**

public Cl1()

* + **Method Detail**
    - **meth3**

public void meth3()

**Specified by:**

[meth3](file:///C:\template\doc\ant\com\lab111\labwork2\If3.html#meth3()) in interface [If3](file:///C:\template\doc\ant\com\lab111\labwork2\If3.html)

**Overrides:**

[meth3](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html#meth3()) in class [Cl3](file:///C:\template\doc\ant\com\lab111\labwork2\Cl3.html)

* + - **meth1**

public void meth1()

**Specified by:**

[meth1](file:///C:\template\doc\ant\com\lab111\labwork2\If1.html#meth1()) in interface [If1](file:///C:\template\doc\ant\com\lab111\labwork2\If1.html)