Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 2

з дисципліни «Інженерія програмного забезпечення»

на тему «Графічна нотація UML, документування проекту»

ВИКОНАВ:

Студент ІІ курсу ФІОТ

групи ІВ-83

Кирiяк ВП

Залікова - 8313

Варіант – 2, 4

ПЕРЕВІРИВ:

Доцент каф. ОТ

к.т.н., с.н.с.

Антонюк А.І.

Київ – 2019

**ЗАВДАННЯ**

**Мета**: Ознайомлення з видами діаграм UML. Отримання базових навичок з використання діаграми класів мови UML. Здобуття навичок з використання засобів автоматизації UML-моделювання на прикладі ArgoUML/Umbrello. Документування проекту за допомогою JavaDoc.

**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 докуменування програмного забезпечення.

6. За допомогою середовища ArgoUML або Umbrello імпортувати сирцеві коди пакету сom.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.

**Варіант** **завдання** **2:**

8309 mod 9 = 2 Генералізація (наслідування):

If1 <- If3

If3 <- If2

Cl1 <- Cl2

**Варіант** **завдання** **4:**

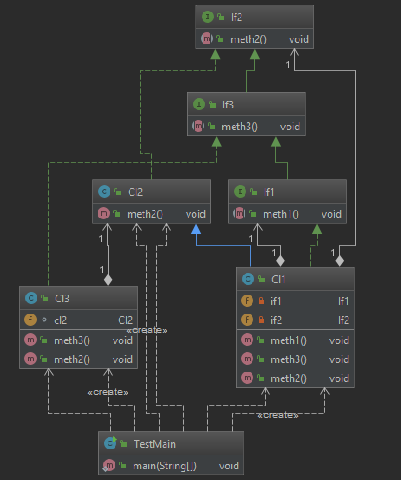
8309 mod 5 = 4 Агрегація:

If1 <- Cl1

If2 <- Cl1

Cl3 <- Cl2

**ДІАГРАМА КЛАСІВ**



**ПРОГРАМНИЙ КОД**

package com.lab111.labwork2;  
*/\*\*  
 \* Class which implements interface "If1", extends "Cl2", consist fields of "If1" and "If2" type and contain  
 \* methods "meth1()", "meth2()", "meth3()"  
 \*  
 \** ***@author Кирiяк ВП*** *\*  
 \*/*public class Cl1 extends Cl2 implements If1 {  
 */\*\*  
 \* fields "If1" and "If2" types  
 \*/* private If1 if1;  
 private If2 if2;  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth1() {  
 System.*out*.println("Meth1 is done! Cl1");  
 }  
  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth3() {  
 System.*out*.println("Meth3 is done! Cl1");  
  
 }  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth2() {  
 System.*out*.println("Meth2 is done! Cl1");  
  
 }  
}

package com.lab111.labwork2;  
*/\*\*  
 \* Class which implements interface "If2", consist fields of "If2" type and contain  
 \* methods "meth2()"  
 \*  
 \** ***@author*** *Drovnin Pavlo  
 \*  
 \*/*public class Cl2 implements If2{  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth2() {  
 System.*out*.println("Meth2 is done! Cl2");  
  
 }  
}

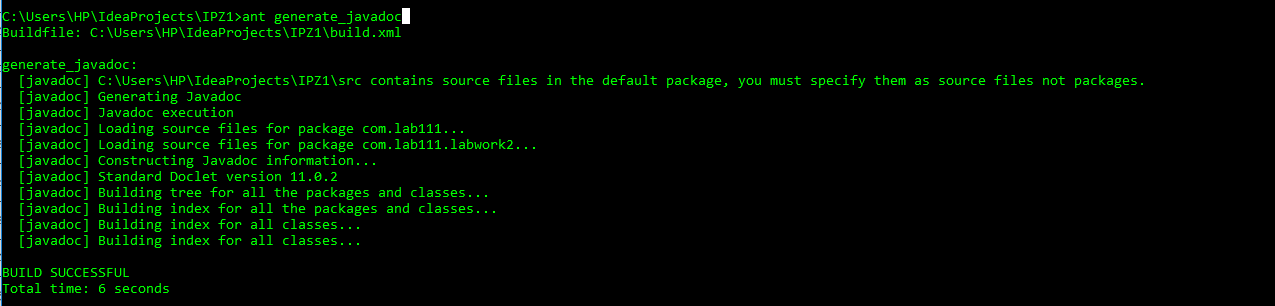
package com.lab111.labwork2;  
*/\*\*  
 \* Class which implements interface "If3",consist fields of "If3" type and contain  
 \* methods "meth2()", "meth3()"  
 \*  
 \** ***@author*** *Drovnin Pavlo  
 \*  
 \*/*public class Cl3 implements If3{  
  
 */\*\*  
 \* Field "Cl2" type  
 \*/* Cl2 cl2;  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth3() {  
 System.*out*.println("Meth2 is done! Cl3");  
  
 }  
  
 */\*\*  
 \* Method, which print information about himself in console  
 \*/* @Override  
 public void meth2() {  
 System.*out*.println("Meth3 is done! Cl3");  
  
 }  
}

package com.lab111.labwork2;  
*/\*\*  
 \* Interface If1 which extends If3,  
 \* consists of abstract method "meth1()" and implemented method "meth3"  
 \*  
 \** ***@author*** *Кирiяк ВП  
 \*  
 \*/*public interface If1 extends If3{  
 void meth1();  
}

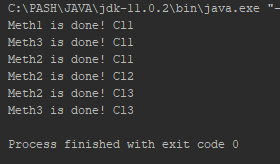
package com.lab111.labwork2;  
*/\*\*  
 \* Interface If2  
 \* consists of abstract method "meth2()"  
 \*  
 \** ***@author*** *Кирiяк ВП  
 \*  
 \*/*public interface If2 {  
 void meth2();  
}

package com.lab111.labwork2;  
*/\*\*  
 \* Interface If3 which extends If2,  
 \* consists of abstract method "meth3()" and implemented method "meth2"  
 \*  
 \** ***@author*** *Кирiяк ВП  
 \*  
 \*/*public interface If3 extends If2{  
 void meth3();  
}

**ГЕНЕРАЦІЯ JavaDoc**



**Результат програми:**



**ВИСНОВКИ**

В ході лабораторної роботи ми ознайомилися з видами діаграм UML, отримали базові навички з використання діаграми класів мови UML на прикладі ArgoUML, реалізували на діаграмі класів відношення генералізації та агрегації, задокументували проект за допомогою JavaDoc.

Побудована діаграма класів відповідає розробленій. Кінцева мета досягнута.