Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 4

З дисципліни «Інженерія програмного забезпечення»

На тему «Структурні шаблони проектування ПЗ - 2. Шаблони Flyweight, Adapter, Bridge, Facade.»

ВИКОНАВ:

студент 2 курсуФІОТ

групи ІО-41

Гаврилюк В.В

№ ЗК - 4107

Варіант – 3

ПЕРЕВІРИВ:

Старший викладач

к.т.н., с.н.с.

Антонюк А.І.

Київ – 2015

ЗАВДАННЯ

Визначити специфікації класів, які подають об'єкти-іконки для зображення елементів файлової системи при побудові графічного інтерфейсу користувача (GUI) - примітиви (файли) та їх композиції (директорії). Забезпечити ефективне використання пам'яті при роботі з великою кількістю графічних об'єктів. Реалізувати метод рисування графічного об'єкту.

ПРОГРАМНИЙ КОД

**package**com.lab111.labwork4;

/\*\*

\* **@author**Victor Havrylyuk

\*

\*/

**publicinterface** Point {

**publicvoid** draw(Coordinates coord);

}

**package com.lab111.labwork4;**

/\*\*

\* **@author**Victor Havrylyuk

\*

\* Class of pixels that implements the Point interface

\*

\*/

**publicclass** Points **implements** Point{

**public** String color;

/\*\*

\* Method that draws a pixel

\*/

@Override

**publicvoid** draw(Coordinates coord){

}

/\*\*

\* Constructor

\*/

**public** Points(String col){

**this**.color = col;

}

}

**package com.lab111.labwork4;**

/\*\*

\* **@author**Victor Havrylyuk

\*

\* Class of coordinates of a pixel

\*/

**publicclass** Coordinates {

**publicint** x;

**publicint** y;

**public** Coordinates(**int** x0, **int** y0){

**this**.x = x0;

**this**.y = y0;

}

}

**package com.lab111.labwork4;**

/\*\*

\* **@author**Victor Havrylyuk

\*

\* A factory for points, creates one point for each color

\*/

**publicclass** PointsFactory {

**publicfinalint** colorQuantity = 6;

**publicfinal** String[] colors = {"Black", "White", "Red", "Green", "Blue", "Octarine"};

**public** Points[] points = **new** Points[colorQuantity];

/\*\*

\* Method that adds a point of a new color

\*/

**public** Points addPoint(**int** col){

**if**(points[col] == **null**)

points[col] = **new** Points(colors[col]);

**return** points[col];

}

}

**package com.lab111.labwork4;**

/\*\*

\* **@author**Victor Havrylyuk

\*

\* Class of an image

\*

\*/

**publicclass** Image {

**publicint** height;

**publicint** width;

**public** PointsFactory pf;

**public** Points [][] pixels;

**publicint** randColor = 0;

/\*\*

\* Constructor that creates an image by corner coordinates and fills it with pixels

\*/

**public** Image(Coordinates p1, Coordinates p2){

**this**.height = p2.y - p1.y + 1;

**this**.width = p2.x - p1.x + 1;

pf = **new** PointsFactory();

pixels = **new** Points[width][];

**for**(**int** i = 0; i < width; i++){

pixels[i] = **new** Points[height];

}

**for**(**int** i = 0; i < width; i++)

**for**(**int** j = 0; j < height; j++){

randColor = (**int**)(Math.*random*()\*6);

pixels[i][j] = pf.addPoint(randColor);

}

}

/\*\*

\* Method that prints an image on the screen pixel by pixel

\*/

**publicvoid** printImage(){

**for**(**int** i = 0; i < width; i++){

**for**(**int** j = 0; j < height; j++)

System.*out*.print(pixels[i][j].color + " ");

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

}

}

}

**package com.lab111.labwork4;**

/\*\*

\* **@author**Victor Havrylyuk

\*

\* Main class that draws an image by given coordinates

\*

\*/

**publicclass**Runner {

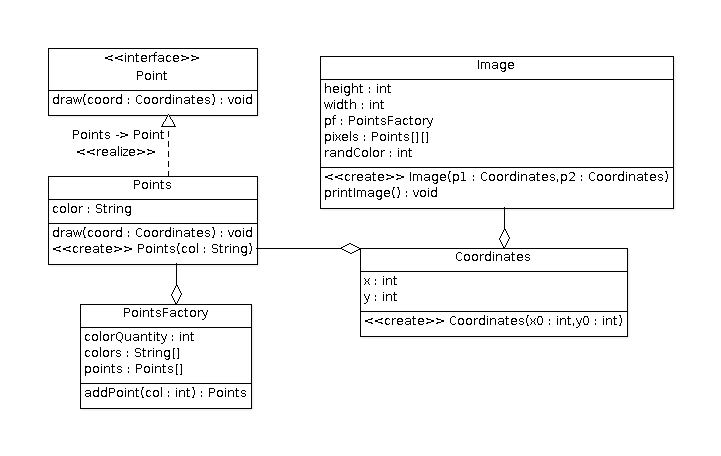
**publicstaticvoid** main(String [] args){

Image i = **new** Image(**new** Coordinates(2,3), **new** Coordinates(5,10));

i.printImage();

}

}

ДІАГРАМА КЛАСІВ

РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ

Black Black Blue Red Yelow Green Green Green

White Green White Red Yelow White Black Green

Yelow Blue White Black Yelow Red Yelow Blue

Black Yelow Blue Red Green Blue Yelow White

ВИСНОВКИ

Ознайомились із структурними шаблонами Flyweight, Adapter, Bridge, Facade.

Отримали навичкиіз застосування шаблонів.

Розроблена відповідна тестова програма.

Згенерований Javadoc.

Результати успішної роботи тестової програми наведені вище підтверджують правильність обраних рішень.