Міністерство освіти та науки України

Одеський Політехнічний Університет

Кафедра “Інформаційні технології”

Звіт з лабораторної роботи №8.2

з теми: **«**Створення графічного інтерфейсу користувача**»**

Варіант №3

Виконав:

cтудент групи АД-192

Гребенщиков Є.Г.

Перевірив:

Рудніченко М.Д.

Одеса 2020

ЗМІСТ

[ВСТУП 3](#_Toc58614765)

[ТЕОРЕТИЧНА ЧАСТИНА 4](#_Toc58614766)

[ПРАКТИЧНА ЧАСТИНА 5](#_Toc58614767)

[Завдання: 6](#_Toc58614768)

[Лістінг: 6](#_Toc58614769)

[Висновок: 8](#_Toc58614770)

# ВСТУП

Мета роботи: навчитися створювати свої класи. Навчитися відображати графічні об'єкти у вікні програми.

# ТЕОРЕТИЧНА ЧАСТИНА

У Java є три бібліотеки візуальних компонентів для створення графічного інтерфейсу користувача.

Найбільш рання з бібліотек для створення графічного інтерфейсу користувача називається AWT (Abstract Window Toolkit). Бібліотечні методи AWT створюють і використовують графічні компоненти операційного середовища. З одного боку, це добре, тому що програма на Java схожа на інші програми в рамках однієї ОС. Але при запуску її на іншій платформі можуть виникнути розбіжності в розмірах компонентів і шрифтів, які будуть псувати зовнішній вигляд програми.

Слідом за AWT була розроблена бібліотека Swing. Вона створена на базі AWT і замінює більшість її компонентів своїми, спроектованими більш ретельно і зручно.

Кожна бібліотека надає набір класів для роботи з кнопками, списками, вікнами, меню і т.д., але ці класи спроектовані по-різному: вони мають різний набір методів з різними параметрами.

**Бібліотека AWT.**

Використання класу Frame для реалізації вікна програми.

Клас Frame є підклас Window, що інкапсулює вікно програми.

Об'єкт Frame може містити рядок меню і відображати заголовок.

**Бібліотека Swing.**

Використання класу JFrame для реалізації вікна програми

Об'єкт класу JFrame є вікно з рамкою і рядком заголовка (з кнопками «Згорнути», «На весь екран» і «Закрити»). Воно може змінювати розміри і переміщатися по екрану.

# ПРАКТИЧНА ЧАСТИНА

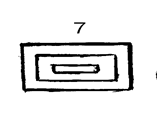
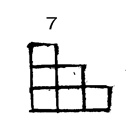
**Завдання:**

* Написати програму, що перетворює введену користувачем фразу типу «<Іменник> + <дієслово в минулому часі> + <наріччя>» одним із таких способів:
* Замінити іменник займенником (враховуючи рід і число).
* Вставити після дієслова «б».
* Замінити наріччя іншим, обраним зі списку.

Вибір одного з дій здійснюється за допомогою радіокнопок, текст вводиться і виводиться в текстове поле, яке повинно очищатися при натисканні відповідної кнопки, вихід з програми - при виборі відповідного пункту меню.

* Написати програму, яка ділить вікно на 4 частини і в кожній частині малює одну з чотирьох картинок, відповідно до варіанту.

Завдання 3.

* Створити додаток - графічний редактор, який дозволяє малювати за допомогою стрілок управління курсором.
* Написати додаток для перевірки вміння переводити число з десяткової в двійкову систему числення, числа задаються випадковим чином. У кутку вікна вивести дату і час запуску програми.

# Завдання:

Написати програму, що перетворює введену користувачем фразу типу «<Іменник> + <дієслово в минулому часі> + <наріччя>» одним із таких способів:

• Замінити іменник займенником (враховуючи рід і число).

• Вставити після дієслова «б».

• Замінити наріччя іншим, обраним зі списку.

Вибір одного з дій здійснюється за допомогою радіокнопок, текст вводиться і виводиться в текстове поле, яке повинно очищатися при натисканні відповідної кнопки, вихід з програми - при виборі відповідного пункту меню.

# Лістінг:

**Main:**

*package* sample;  
  
*import* javafx.application.Application;  
*import* javafx.scene.Group;  
*import* javafx.scene.Scene;  
  
*import* javafx.stage.Stage;  
*import* javafx.scene.shape.LineTo;  
*import* javafx.scene.shape.MoveTo;  
*import* javafx.scene.shape.Path;  
  
*public class* Main *extends* Application {  
  
 @Override  
 *public void* start(Stage primaryStage) *throws* Exception{  
  
 *//TODO: Star* Path path = *new* Path();  
  
 *//Moving to the starting point* MoveTo moveTo=*new* MoveTo(18, 90);  
  
 *//Creating 1st line* LineTo line1 = *new* LineTo(220, 90);  
  
 *//Creating 2nd line* LineTo line2 = *new* LineTo(36,210);  
  
 *//Creating 3rd line* LineTo line3 = *new* LineTo(120,12);  
  
 *//Creating 4th line* LineTo line4 = *new* LineTo(179, 210);  
  
 *//Creating 5th line* LineTo line5 = *new* LineTo(19, 90);  
  
 *//Adding all elements to the path* path.getElements().add(moveTo);  
 path.getElements().addAll(line1, line2, line3, line4, line5);  
 *//Creating a Group object* Group root = *new* Group(path);  
  
 *//Creating a scene Object* Scene scene = *new* Scene(root, 600,600);  
 primaryStage.setTitle("Drawing");  
 primaryStage.setScene(scene);  
 *//Displaying the contents of the stage* primaryStage.show();  
  
 *//TODO : Rocket* MoveTo moveTo1 = *new* MoveTo(490, 20);  
  
 LineTo line1\_1 = *new* LineTo(455, 60);  
 LineTo line1\_2 = *new* LineTo(455, 120);  
 LineTo line1\_3 = *new* LineTo(435, 130);  
 LineTo line1\_4 = *new* LineTo(435, 150);  
 LineTo line1\_5 = *new* LineTo(455,140);  
 LineTo line1\_6 = *new* LineTo(520, 140);  
 LineTo line1\_7 = *new* LineTo(540, 150);  
 LineTo line1\_8 = *new* LineTo(540, 130);  
 LineTo line1\_9 = *new* LineTo(520, 120);  
 LineTo line1\_10 = *new* LineTo(520,60);  
 LineTo line1\_11 = *new* LineTo(490, 20);  
 path.getElements().add(moveTo1);  
 path.getElements().addAll(  
 line1\_1, line1\_2, line1\_3, line1\_4,  
 line1\_5, line1\_6, line1\_7, line1\_8,  
 line1\_9, line1\_10, line1\_11);  
  
 *//TODO: some square with pattern  
 //TODO: square* MoveTo moveTo2 = *new* MoveTo(30, 500);  
 LineTo line2\_1 = *new* LineTo(30, 350);  
 LineTo line2\_2 = *new* LineTo(180, 350);  
 LineTo line2\_3 = *new* LineTo(180, 500);  
 LineTo line2\_4 = *new* LineTo(30,500);  
 path.getElements().add(moveTo2);  
 path.getElements().addAll(line2\_1, line2\_2, line2\_3, line2\_4);  
  
 *//TODO: pattern  
 //1st part* MoveTo moveTo3 = *new* MoveTo(105, 425);  
 LineTo line3\_1 = *new* LineTo(150,435);  
 LineTo line3\_2 = *new* LineTo(180,500);  
 LineTo line3\_3 = *new* LineTo(120,480);  
 LineTo line3\_4 = *new* LineTo(105, 425);  
 *//2nd part* LineTo line3\_5 = *new* LineTo(150, 405);  
 LineTo line3\_6 = *new* LineTo(180, 350);  
 LineTo line3\_7 = *new* LineTo(120, 370);  
 LineTo line3\_8 = *new* LineTo(105, 425);  
 *//3rd part* LineTo line3\_9 = *new* LineTo(90, 370);  
 LineTo line3\_10 = *new* LineTo(30, 350);  
 LineTo line3\_11 = *new* LineTo(60 ,405);  
 LineTo line3\_12 = *new* LineTo(105, 425);  
 *//4th part* LineTo line3\_13 = *new* LineTo(60 ,435);  
 LineTo line3\_14 = *new* LineTo(30, 500);  
 LineTo line3\_15 = *new* LineTo(90, 480);  
 LineTo line3\_16 = *new* LineTo(105, 425);  
  
 path.getElements().add(moveTo3);  
 path.getElements().addAll(  
 line3\_1, line3\_2,line3\_3, line3\_4,  
 line3\_5,line3\_6, line3\_7, line3\_8,  
 line3\_9, line3\_10, line3\_11, line3\_12,  
 line3\_13, line3\_14, line3\_15, line3\_16);  
  
 *//TODO: КВАДРАТИКИИИИИИ* MoveTo moveTo4 = *new* MoveTo(470, 425);  
 LineTo line4\_1 = *new* LineTo(510, 425);  
 LineTo line4\_2 = *new* LineTo(510, 465);  
 LineTo line4\_3 = *new* LineTo(550, 465);  
 LineTo line4\_4 = *new* LineTo(550, 425);  
 LineTo line4\_5 = *new* LineTo(510, 425);  
 LineTo line4\_6 = *new* LineTo(510, 385);  
 LineTo line4\_7 = *new* LineTo(470, 385);  
 LineTo line4\_8 = *new* LineTo(470, 345);  
 LineTo line4\_9 = *new* LineTo(430, 345);  
 LineTo line4\_10 = *new* LineTo(430, 385);  
 LineTo line4\_11 = *new* LineTo(470, 385);  
 LineTo line4\_12 = *new* LineTo(470, 425);  
 LineTo line4\_13 = *new* LineTo(430, 425);  
 LineTo line4\_14 = *new* LineTo(430, 465);  
 LineTo line4\_15 = *new* LineTo(470, 465);  
 LineTo line4\_16 = *new* LineTo(470, 425);  
  
 path.getElements().add(moveTo4);  
 path.getElements().addAll(  
 line4\_1, line4\_2, line4\_3, line4\_4,  
 line4\_5, line4\_6, line4\_7, line4\_8,  
 line4\_9, line4\_10, line4\_11, line4\_12,  
 line4\_13, line4\_14, line4\_15, line4\_16  
 );  
  
 }  
  
  
 *public static void* main(String[] args) {  
 launch(args);  
 }  
}

# Висновок:

Створив додаток - графічний редактор, який дозволяє малювати за допомогою стрілок управління курсором.