

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислюальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота № 1

з дисципліни «Технології та засоби розробки комп'ютерної графіки та
мультимедіа»

Тема: «Базові поняття комп'ютерної

графіки. Ініціалізація графічного середовища»

Виконала:

студентка групи IC-34

Ященко Олександра.

Дата здачі 08.10.25

Захищено з балом _____

Перевірила:

ст. вик. кафедри ICT

Хмельюк Марина Сергіївна

Київ 2025

Завдання

Створити графічний інтерфейс користувача за допомогою бібліотеки PyQt5, який виводить прізвище студента за допомогою графічних примітивів (ліній, дуг, еліпсів). Кожна літера має бути намальована окремим кольором з використанням базових графічних функцій.

Опис реалізації

Структура програми

Програма складається з двох основних класів:

1. GraphicsForm (QMainWindow) — головне вікно програми

- Розмір: 600×350 пікселів
- Чорний фон
- Містить заголовок, інформацію про студента, область малювання та кнопку закриття
- Кастомна іконка вікна (синє коло)

2. DrawingWidget (QWidget) — віджет для малювання

- Розмір: 700×300 пікселів
- Білий фон з рамкою
- Реалізує метод paintEvent() для відтворення графіки

Технічні рішення

Використані графічні примітиви:

- drawLine() — для літер Y, A, H, E, N, K
- drawArc() — для літер S, C
- drawEllipse() — для літери O

Кольорова схема: Кожна літера прізвища "YASHCHENKO" має унікальний колір, що формує плавний градієнт від червоного до рожевого через спектр веселки.

Антиаліасинг: Застосовано QPainter.Antialiasing для згладжування країв і покращення якості відображення.

Код програми

```
import sys
from PyQt5.QtWidgets import *
from PyQt5.QtCore import Qt
from PyQt5.QtGui import *

class GraphicsForm(QMainWindow):
    def __init__(self):
        super().__init__()
        self.initUI()

    def initUI(self):
        self.setWindowTitle("Lab №1")
        self.setFixedSize(600, 350)

        self.setWindowIcon(self.createIcon())

        self.setStyleSheet("background-color: black")

        central_widget = QWidget()
        self.setCentralWidget(central_widget)

        layout = QVBoxLayout()
        layout.setAlignment(Qt.AlignCenter)

        title_label = QLabel("Lab №1")
        title_label.setFont(QFont("Arial", 20, QFont.Bold))
        title_label.setStyleSheet("color: white; margin: 10px;")
        title_label.setAlignment(Qt.AlignCenter)

        student_label = QLabel("Student: Yashchenko Oleksandra, Group IC-34")
        student_label.setFont(QFont("Arial", 12, QFont.StyleItalic))
        student_label.setStyleSheet("color: #bdc3c7; margin: 5px;")
        student_label.setAlignment(Qt.AlignCenter)

        self.drawing_widget = DrawingWidget()

        close_button = QPushButton("Close")
        close_button.setFont(QFont("Arial", 12))
        close_button.setStyleSheet("""
            QPushButton {
                background-color: red;
                color: white;
                margin: 5px;
                border: 1px solid red;
                border-radius: 20px;
                padding: 10px 30px;
            }
            QPushButton:hover {
                background-color: white;
                color: black;
                border: none;
            }
        """)

        layout.addWidget(title_label)
        layout.addWidget(student_label)
        layout.addWidget(drawing_widget)
        layout.addWidget(close_button)

        central_widget.setLayout(layout)
```

```

        """)

close_button.clicked.connect(self.close)

layout.addWidget(title_label)
layout.addWidget(student_label)
layout.addWidget(self.drawing_widget)
layout.addWidget(close_button)

central_widget.setLayout(layout)

def createIcon(self):
    pixmap = QPixmap(32, 32)
    pixmap.fill(QColor("#3498db"))

    painter = QPainter(pixmap)
    painter.setPen(QPen(QColor("white"), 2))
    painter.drawEllipse(8, 8, 16, 16)
    painter.end()

    return QIcon(pixmap)

class DrawingWidget(QWidget):
    def __init__(self):
        super().__init__()
        self.setFixedSize(700, 300)
        self.setStyleSheet("background-color: white; border: 2px solid #2c3e50;")

    def paintEvent(self, event):
        painter = QPainter(self)
        painter.setRenderHint(QPainter.Antialiasing)

        self.drawSurname(painter)

    def drawSurname(self, painter):
        painter.setFont(QFont("Arial", 12, QFont.Bold))
        painter.setPen(QColor("#2c3e50"))

        # Y
        painter.setPen(QPen(QColor("#cc6155"), 3))
        painter.drawLine(50, 50, 65, 90)
        painter.drawLine(80, 50, 65, 90)
        painter.drawLine(65, 90, 65, 120)

        # A
        painter.setPen(QPen(QColor("#e6935c"), 3))
        painter.drawLine(100, 120, 115, 50)
        painter.drawLine(115, 50, 130, 120)
        painter.drawLine(108, 85, 122, 85)

        # S
        painter.setPen(QPen(QColor("#f0e54c"), 3))
        painter.drawArc(150, 50, 25, 35, 0, 180*16)
        painter.drawArc(150, 50, 25, 35, 270*9, 315*6)
        painter.drawArc(150, 85, 25, 35, 90*1, 180*7)
        painter.drawArc(150, 85, 25, 35, 180*16, 180*16)

        # H
        painter.setPen(QPen(QColor("#8fd44d"), 3))

```

```

painter.drawLine(200, 50, 200, 120)
painter.drawLine(225, 50, 225, 120)
painter.drawLine(200, 85, 225, 85)

# C
painter.setPen(QPen(QColor("#67d1bc"), 3))
painter.drawArc(250, 50, 30, 70, 30*16, 300*16)

# H
painter.setPen(QPen(QColor("#6d9feb"), 3))
painter.drawLine(300, 50, 300, 120)
painter.drawLine(325, 50, 325, 120)
painter.drawLine(300, 85, 325, 85)

# E
painter.setPen(QPen(QColor("#194BEE"), 3))
painter.drawLine(350, 50, 350, 120)
painter.drawLine(350, 50, 375, 50)
painter.drawLine(350, 85, 370, 85)
painter.drawLine(350, 120, 375, 120)

# N
painter.setPen(QPen(QColor("#a55fe7"), 3))
painter.drawLine(400, 50, 400, 120)
painter.drawLine(425, 50, 425, 120)
painter.drawLine(400, 85, 425, 120)

# K
painter.setPen(QPen(QColor("#e575f0"), 3))
painter.drawLine(450, 50, 450, 120)
painter.drawLine(450, 85, 475, 50)
painter.drawLine(450, 85, 475, 120)

# O
painter.setPen(QPen(QColor("#f049aa"), 3))
painter.drawEllipse(500, 50, 30, 70)

def main():
    app = QApplication(sys.argv)

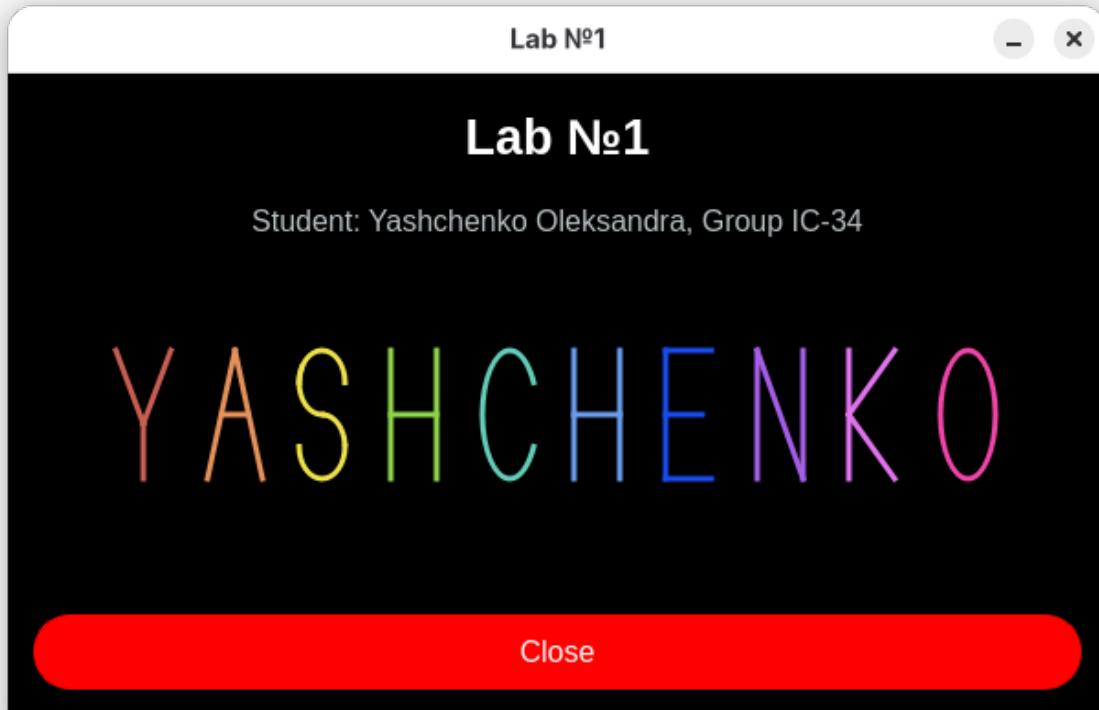
    form = GraphicsForm()
    form.show()

    sys.exit(app.exec_())

if __name__ == "__main__":
    main()

```

Результат



Висновки

В ході виконання лабораторної роботи було успішно освоєно:

- Створення графічних інтерфейсів за допомогою PyQt5
- Робота з класами QPainter, QPen, QColor
- Використання графічних примітивів для малювання складних форм
- Застосування стилізації елементів через CSS-подібний синтаксис
- Організація коду з використанням ООП принципів

Програма відповідає всім вимогам завдання: відображає прізвище студента за допомогою графічних примітивів, має зручний інтерфейс та коректно працює.