Міністерство освіти і науки України

**Прикарпатський національний університет**

**імені В.Стефаника**

*Факультет математики та інформатики*

*Кафедра інформаційних технологій*

*Людино-машинна взаємодія*

Лабораторна робота № 4

Тема: Основи роботи з QT бібліотекою. Створення консольних та Qt Widgets додатків у інтегрованому середовищі розробки QT-Сreator

Варіант: 14

Виконав: ***Пикало М.І***

Група ІПЗ-23

Дата: 14 листопада 2024р.

Викладач: Пікуляк М.В.

Івано-Франківськ - 2024

**Завдання для виконання**

Реалізувати додатки з графічним інтерфейсом користувача, використавши при цьому наявні можливості по роботі з шрифтами, кольором, фоном і т.д.

**Варіант 2**

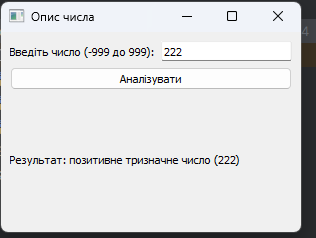
Задача 1. Число, що лежить в діапазоні від -999 до 999, вводиться в lineedit. Реалізувати додаток з графічним інтерфейсом для виведення інформаційного повідомлення про словесний опис даного числа виду «негативне двозначне число», «нульове число», «позитивне однозначне число» і т.д.

Задача 2. Дано три змінні: X, Y, Z (змінні вводяться в lineedit). Якщо їх значення впорядковані за зростанням або зменшенням, то подвоїти їх; в іншому випадку замінити значення кожної змінної на протилежне.

Задача 3. Реалізувати додаток з графічним інтерфейсом вікна авторизації студента (за логіном і паролем).

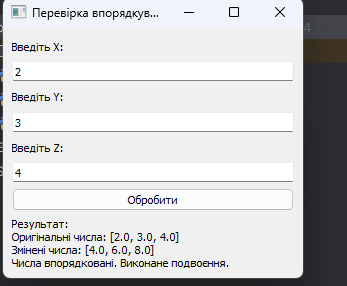
Завдання 1

import sys  
from PyQt5.QtWidgets import (QApplication, QMainWindow, QWidget, QVBoxLayout,  
 QHBoxLayout, QLabel, QLineEdit, QPushButton, QMessageBox)  
  
  
class NumberDescriptionApp(QMainWindow):  
 def \_\_init\_\_(self):  
 super().\_\_init\_\_()  
 self.setWindowTitle('Опис числа')  
 self.setGeometry(100, 100, 300, 200)  
  
 central\_widget = QWidget()  
 self.setCentralWidget(central\_widget)  
 layout = QVBoxLayout()  
 central\_widget.setLayout(layout)  
  
 # Введення числа  
 input\_layout = QHBoxLayout()  
 self.number\_input = QLineEdit()  
 input\_layout.addWidget(QLabel('Введіть число (-999 до 999):'))  
 input\_layout.addWidget(self.number\_input)  
 layout.addLayout(input\_layout)  
  
 # Кнопка аналізу  
 analyze\_btn = QPushButton('Аналізувати')  
 analyze\_btn.clicked.connect(self.analyze\_number)  
 layout.addWidget(analyze\_btn)  
  
 # Результат  
 self.result\_label = QLabel('Результат: ')  
 layout.addWidget(self.result\_label)  
  
 def analyze\_number(self):  
 try:  
 number = int(self.number\_input.text())  
  
 if number < -999 or number > 999:  
 raise ValueError("Число поза допустимим діапазоном")  
  
 if number == 0:  
 description = "нульове число"  
 elif number < 0:  
 if -10 <= number < 0:  
 description = f"негативне однозначне число ({number})"  
 elif -100 <= number < -10:  
 description = f"негативне двозначне число ({number})"  
 else:  
 description = f"негативне тризначне число ({number})"  
 else:  
 if 0 < number < 10:  
 description = f"позитивне однозначне число ({number})"  
 elif 10 <= number < 100:  
 description = f"позитивне двозначне число ({number})"  
 else:  
 description = f"позитивне тризначне число ({number})"  
  
 self.result\_label.setText(f'Результат: {description}')  
  
 except ValueError as e:  
 QMessageBox.warning(self, 'Помилка', str(e))  
  
  
def main():  
 app = QApplication(sys.argv)  
 window = NumberDescriptionApp()  
 window.show()  
 sys.exit(app.exec\_())  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()



Завдання 2

import sys  
from PyQt5.QtWidgets import (QApplication, QMainWindow, QWidget, QVBoxLayout,  
 QLabel, QLineEdit, QPushButton, QMessageBox)  
  
  
class NumberProcessingApp(QMainWindow):  
 def \_\_init\_\_(self):  
 super().\_\_init\_\_()  
 self.setWindowTitle('Перевірка впорядкування чисел')  
 self.setGeometry(100, 100, 300, 250)  
  
 central\_widget = QWidget()  
 self.setCentralWidget(central\_widget)  
 layout = QVBoxLayout()  
 central\_widget.setLayout(layout)  
  
 # Введення чисел  
 self.x\_input = QLineEdit()  
 self.y\_input = QLineEdit()  
 self.z\_input = QLineEdit()  
  
 layout.addWidget(QLabel('Введіть X:'))  
 layout.addWidget(self.x\_input)  
 layout.addWidget(QLabel('Введіть Y:'))  
 layout.addWidget(self.y\_input)  
 layout.addWidget(QLabel('Введіть Z:'))  
 layout.addWidget(self.z\_input)  
  
 # Кнопка обробки  
 process\_btn = QPushButton('Обробити')  
 process\_btn.clicked.connect(self.process\_numbers)  
 layout.addWidget(process\_btn)  
  
 # Результат  
 self.result\_label = QLabel('Результат: ')  
 layout.addWidget(self.result\_label)  
  
 def process\_numbers(self):  
 try:  
 x = float(self.x\_input.text())  
 y = float(self.y\_input.text())  
 z = float(self.z\_input.text())  
  
 original\_values = [x, y, z]  
 modified\_values = original\_values.copy()  
  
 # Перевірка впорядкованості  
 if x <= y <= z or x >= y >= z:  
 # Подвоєння чисел  
 modified\_values = [val \* 2 for val in modified\_values]  
 description = "Числа впорядковані. Виконане подвоєння."  
 else:  
 # Зміна на протилежні  
 modified\_values = [-val for val in modified\_values]  
 description = "Числа не впорядковані. Змінені на протилежні."  
  
 result\_text = (  
 f"Результат:\n"  
 f"Оригінальні числа: {original\_values}\n"  
 f"Змінені числа: {modified\_values}\n"  
 f"{description}"  
 )  
  
 self.result\_label.setText(result\_text)  
  
 except ValueError:  
 QMessageBox.warning(self, 'Помилка', 'Введіть коректні числові значення')  
  
  
def main():  
 app = QApplication(sys.argv)  
 window = NumberProcessingApp()  
 window.show()  
 sys.exit(app.exec\_())  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()



Завдання 3

import sys  
from PyQt5.QtWidgets import (QApplication, QMainWindow, QWidget, QVBoxLayout,  
 QHBoxLayout, QLabel, QLineEdit, QPushButton)  
  
  
class StudentAuthorizationApp(QMainWindow):  
 def \_\_init\_\_(self):  
 super().\_\_init\_\_()  
 self.setWindowTitle('Авторизація студента')  
 self.setGeometry(100, 100, 300, 200)  
  
 central\_widget = QWidget()  
 self.setCentralWidget(central\_widget)  
 layout = QVBoxLayout()  
 central\_widget.setLayout(layout)  
  
 # Логін  
 login\_layout = QHBoxLayout()  
 login\_layout.addWidget(QLabel('Логін:'))  
 self.login\_input = QLineEdit()  
 login\_layout.addWidget(self.login\_input)  
 layout.addLayout(login\_layout)  
  
 # Пароль  
 password\_layout = QHBoxLayout()  
 password\_layout.addWidget(QLabel('Пароль:'))  
 self.password\_input = QLineEdit()  
 self.password\_input.setEchoMode(QLineEdit.Password)  
 password\_layout.addWidget(self.password\_input)  
 layout.addLayout(password\_layout)  
  
 # Кнопка входу  
 login\_btn = QPushButton('Вхід')  
 login\_btn.clicked.connect(self.check\_authorization)  
 layout.addWidget(login\_btn)  
  
 # Результат  
 self.result\_label = QLabel('')  
 layout.addWidget(self.result\_label)  
  
 def check\_authorization(self):  
 # Жорстко закодовані дані для прикладу  
 correct\_login = 'student'  
 correct\_password = '12345'  
  
 login = self.login\_input.text()  
 password = self.password\_input.text()  
  
 if login == correct\_login and password == correct\_password:  
 self.result\_label.setText('Вхід успішний!')  
 self.result\_label.setStyleSheet('color: green')  
 else:  
 self.result\_label.setText('Невірний логін або пароль')  
 self.result\_label.setStyleSheet('color: red')  
  
  
def main():  
 app = QApplication(sys.argv)  
 window = StudentAuthorizationApp()  
 window.show()  
 sys.exit(app.exec\_())  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

