**AMALIY MASHG‘ULOT UCHUN O‘QUV MATERIALLARI**

**2-Mavzu:** PyQt5 paketi va QtDesigner dasturi yordamida GUI dasturlarini yaratish.

**4-mashg‘ulot.** PyQt5 modal dialog. QMessageBox vidgeti bilan ishlash.

**O‘quv savollari:**

1. QMessageBox vidgetining vazifasi.
2. QMessageBox vidgetining asosiy xususiyatlari.
3. Statik funksiyalari.
4. Piktogramma va Pixmap xususiyatlari.

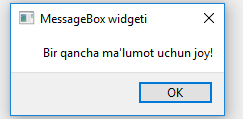
**1. QMessageBox vidgetining vazifasi**

QMessageBox - bu dialoglar yaratish uchun ishlatiladigan foydali PyQt5 vidjeti. Ushbu dialog oynalari turli maqsadlarda ishlatilishi va turli xil xabarlar va tugmalarni ko‘rsatish uchun ko‘p jihatdan moslashtirilgan bo‘lishi mumkin.

**Quyida oddiy QMessageBox** vidjetini yaratmoqda. Ammo bu hech qanday qo‘shimcha funksiyalar yoki sozlashlarsiz faqat MessageBox vidjetining o‘zining tinchlik xolatidagi ko‘rinishi bo‘ladi.

|  |  |
| --- | --- |
|  | def display():  msg = QMessageBox(wind)  msg.setWindowTitle("MessageBox widgeti")  msg.setText("Bir qancha ma’lumot uchun joy!")    x = msg.exec\_() |

Va quyida chiqish. Agar siz ekranda oddiy xabarni ko‘rsatmoqchi bo‘lsangiz, buni qilishning yo‘li.



*61-rasm. QmessageBox vidjetining o‘zining ko‘rinishi*

**2. QMessageBox vidgetining asosiy xususiyatlari**

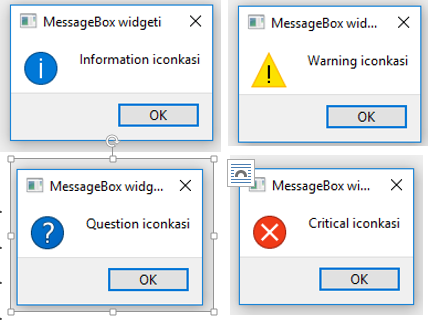
**QMessageBox –** da .**setIcon()** metodini chaqirish orqali foydalanish mumkin bo‘lgan 4 xil turdagi piktogramma mavjud bo‘lib, ko‘rsatmoqchi bo‘lgan xabar turiga qarab qaysi birini ishlatishni hal qilinadi.

* QMessageBox.Question
* QMessageBox.Information
* QMessageBox.Warning
* QMessageBox.Critical

Yuqoridagilardan birini quyida ko‘rsatilganidek **setIcon()** funksiyasiga o‘tkazish kifoya .

|  |  |
| --- | --- |
|  | msg.setIcon(QMessageBox.Question) |

Quyida ko‘rsatilishi mumkin bo‘lgan to‘rt xil xabarlar mavjud.



*62-rasm. QMessageBox pictogrammalarining turlari*

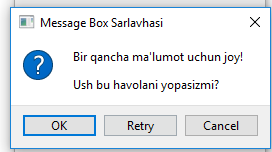
**QMessageBox tugmalari**

**Hozirgacha biz QMessageBox taqdim etgan standart “OK**” tugmasidan foydalanib kelmoqdamiz. Biroq, QMessageBox foydalanish uchun o‘ndan ortiq turli xil tugmalar taklif qiladi. Quyida ba’zi tugmalardan foydalanish misoli keltirilgan.

**setStandardButtons**() funktsiyadan foydalanib xohlagan tugma turini o‘ratish mumkin.

|  |  |
| --- | --- |
|  | def display():  msg = QMessageBox(wind)  msg.setWindowTitle("Message Box Sarlavhasi")  msg.setText("Bir qancha ma’lumot uchun joy!")  msg.setIcon(QMessageBox.Question)  msg.setStandardButtons(QMessageBox.Cancel  |QMessageBox.Ok  |QMessageBox.Retry)  msg.setInformativeText("Ush bu havolani yopasizmi?")  x = msg.exec\_() |

Ushbu dasturning natijasi quyida keltirilgan:



*63-rasm. QMessageBox ga doir oddiy dastur natijasi*

Buni allaqachon payqagan bo‘lishingiz mumkin, lekin tugmalar tartibi ularning xabarlar oynasida qanday ko‘rinishiga ta’sir qilmaydi.

**Default Button**

Yuqoridagi rasmga qarab “OK” tugmasi atrofida ko‘k kontur borligini payqash mumkin . Bu ushbu tugma **Default Button** ekanligini bildiradi.

**setDefaultButton**() funktsiyasidan foydalanib, standart tugmani o‘zgartirish mumkin. Kodga quyidagi qatorni qo‘shish ko‘k rangni "**Cancel**" tugmasi atrofida bo‘lishiga olib keladi.

|  |  |
| --- | --- |
|  | msg.setDefaultButton(QMessageBox.Cancel) |

Bu yerda **QmessageBox** vidjetida foydalanish mumkin bo‘lgan turli xil tugma turlarining to‘liq ro‘yxati keltirilgan:

* QMessageBox.Ok
* QMessageBox.No
* QMessageBox.Yes
* QMessageBox.Cancel
* QMessageBox.Close
* QMessageBox.Abort
* QMessageBox.open
* QMessageBox.Retry
* QMessageBox.Ignore
* QMessageBox.Save
* QMessageBox.Retry
* QMessageBox.Apply
* QMessageBox.Help
* QMessageBox.Reset
* QMessageBox.SaveAll
* QMessageBox.YesToAll
* QMessageBox.NoToAll

**3. Statik funksiyalari**

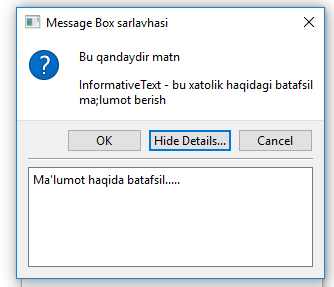
Odatda QMessageBox-da Matnni ko‘rsatadigan faqat bitta maydon mavjud. Aslida esa, qo‘shimcha matn qo‘shish uchun yana ikkita qo‘shimcha maydon mavjud.

Birinchisi, QMessageBox oynasining o‘zida qo‘shimcha matn bo‘limi. Bu qo‘shish mumkin bo‘lgan qo‘shimcha matn qatoriga o‘xshaydi. Ushbu yangi matn maydonini qo‘shish uchun faqat setInformativeText() funksiyasiga murojaat qilish kerak.

Ikkinchisi QMessageBox-dan kengaytirilgan maydonda ko‘rsatiladi. Bu **Detailed Text** – ya’ni “Batafsil matn” deb ataladi. Ushbu bo‘limni o‘rnatish avtomatik ravishda ushbu maydonni ko‘rsatish uchun ishlatiladigan tugmani qo‘shadi. Ushbu batafsil matn yoziladigan maydonni qo‘shish uchun faqat **setDetailedText()** funksiya kerak bo‘ladi.

|  |  |
| --- | --- |
|  | from PyQt5 import QtWidgets  from PyQt5.QtWidgets import \*  import sys    def show\_popup():  msg = QMessageBox(win)  msg.setWindowTitle("Message Box sarlavhasi")  msg.setText("Bu qandaydir matn")  msg.setIcon(QMessageBox.Question)    msg.setStandardButtons(QMessageBox.Cancel|QMessageBox.Ok)  msg.setDefaultButton(QMessageBox.Ok)    msg.setDetailedText("Ma’lumot haqida batafsil.....")  msg.setInformativeText("InformativeText - bu xatolik haqidagi batafsil ma;lumot berish")  x = msg.exec\_()    app = QApplication(sys.argv)  win = QMainWindow()  win.setGeometry(400,400,300,300)  win.setWindowTitle("QMessageBox widgeti haqida")    button = QtWidgets.QPushButton(win)  button.setText("A Button")  button.clicked.connect(show\_popup)  button.move(100,100)    win.show()  sys.exit(app.exec\_()) |

Yuqoridagi dasturning ishga tushirilganidan keyin quyidagi messagebox oynasi paydo bo‘ladi:



*QMessageBox widgetining to‘liq ko‘rinishi*

Hide Details/Show Details tugmasini bosish mos ravishda xabarlar qutisi ostidagi qo‘shimcha maydonni yashirish/ko‘rsatish mumkin.

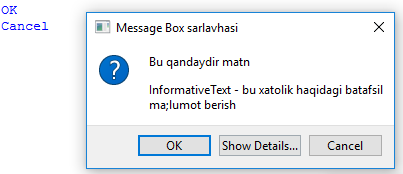
**QMessageBox qiymatlarini olish**

Yuqorida juda ko‘p turli xil sozlashlar va xususiyatlarni muhokama qilindi, ammo aslida bu turli xususiyat va tugmalarning hech birini kodga bog‘lanmadi.

Misol uchun, agar MessageBoxda 3 xil tugma bo‘lsa, qaysi biri bosilganligini qanday bilish mumkin?  Buning uchun quyida ko‘rsatilgan kodyaqqol javob bo‘la oladi:

|  |  |
| --- | --- |
|  | from PyQt5 import QtWidgets  from PyQt5.QtWidgets import \*  import sys    def show\_popup():  msg = QMessageBox(win)  msg.setWindowTitle("Message Box sarlavhasi")  msg.setText("Bu qandaydir matn")  msg.setIcon(QMessageBox.Question)  msg.setStandardButtons(QMessageBox.Cancel|QMessageBox.Ok)  msg.setDefaultButton(QMessageBox.Ok)    msg.setDetailedText("Ma’lumot haqida batafsil.....")  msg.setInformativeText("InformativeText - bu xatolik haqidagi batafsil ma;lumot berish")  msg.buttonClicked.connect(popup)  x = msg.exec\_()    def popup(i):  print(i.text())    app = QApplication(sys.argv)  win = QMainWindow()  win.setGeometry(400,400,300,300)  win.setWindowTitle("QMessageBox widgeti haqida")    button = QtWidgets.QPushButton(win)  button.setText("A Button")  button.clicked.connect(show\_popup)  button.move(100,100)    win.show()  sys.exit(app.exec\_()) |

**buttonClicked.connect()** usulidan foydalanib  MessageBoxdagi birorta tugma bosilganda funktsiyani ishga tushirish mumkin. Bunday holda, messageboxni **popup** deb nomlangan funktsiyaga bog‘landi. popup() funsiyas bitta parametr qabul qiladi.



*QMessageBox vidjetining to‘liq imkoniyatlari bilan ko‘rinishi*

MessageBoxning tugmasi bosilganda, u avtomatik ravishda bosilgan tugmani **buttonClicked.connect()** da e’lon qilingan funksiyaga uzatadi. Ushbu tugmadagi text() funktsiyasini chaqirish esa tugmaning matnini qaytaradi.

**Nazorat savollari:**

1. QMessageBox vidjeti qanday vazifani bajaradi?
2. QMessageBox qanday ishga tushiriladi?
3. QMessageBox ning qanday xususiyatlari mavjud?
4. QMessageBox vidgetida qanday pictogrammalar mavjud?
5. QmessageBoxga o‘rnatish mumkin bo‘lhan standart buttonlarni keltiring?