

[소프트웨어프로젝트 II]

[PyQt_GUI 기반 ScoreDB 관리]

20171661_이다운_class04

이시윤 교수님

1. 라벨 생성

23	+	#labels
22	24	name = QLabel("Name: ")
23	25	age = QLabel("Age: ")
24	26	score = QLabel("Score: ")
25	27	amount = QLabel("Amount: ")
26	28	key = QLabel("Key: ")
27	29	result = QLabel("Result: ")
28	30	

- QLabel 클래스에 대한 각 인스턴스들을 만들어서 변수에 넣습니다.

2. 입력창 생성

31	+	#lineEdit
29	32	self.nameEdit = QLineEdit()
30	33	self.ageEdit = QLineEdit()
31	34	self.scoreEdit = QLineEdit()
32	35	self.amountEdit = QLineEdit()
33	36	self.keyCombo = QComboBox()

- QLineEdit 클래스에 대한 인스턴스들을 만들어서 변수에 넣습니다.
- self.변수로 나타내줌으로써 다른 함수에서 이 변수들을 사용할 수 있습니다.

3. ComboBox 생성

33	36	self.keyCombo = QComboBox()
34	37	self.keyCombo.addItem(["Name", "Age", "Score"])

- QComboBox 클래스를 사용하여 인스턴스를 생성합니다.
- addItem 를 사용하여 ComboBox 의 원소들을 추가해 줍니다.

4. 버튼생성

40	+	#Buttons
37	41	addButton = QPushButton("Add")
38	42	delButton = QPushButton("Del")
39	43	findButton = QPushButton("Find")
40	44	incButton = QPushButton("Inc")
41	45	showButton = QPushButton("show")

- QPushButton 클래스에 대한 인스턴스들을 만들어서 변수에 넣습니다.
- 아직은 생성된 버튼에 대한 이벤트가 없으므로 클릭해도 아무런 반응이 없습니다.

5. layout 생성

	47	+	#layout
43	48		hbox = QHBoxLayout()
44	49		hbox2 = QHBoxLayout()
45	50		hbox2.addStretch(1)
46	51		hbox3 = QHBoxLayout()
47	52		hbox3.addStretch(1)
48	53		hbox4 = QHBoxLayout()
49	54		hbox5 = QHBoxLayout()
50	55		
51	56		hbox.addWidget(name)
52	57		hbox.addWidget(self.nameEdit)
53	58		hbox.addWidget(age)
54	59		hbox.addWidget(self.ageEdit)
55	60		hbox.addWidget(score)
56	61		hbox.addWidget(self.scoreEdit)
57	62		hbox2.addWidget(amount)
58	63		hbox2.addWidget(self.amountEdit)
59	64		hbox2.addWidget(key)
60	65		hbox2.addWidget(self.keyCombo)
61	66		hbox3.addWidget(addButton)
62	67		hbox3.addWidget(delButton)
63	68		hbox3.addWidget(findButton)
64	69		hbox3.addWidget(incButton)
65	70		hbox3.addWidget(showButton)
66	71		hbox4.addWidget(result)
67	72		hbox5.addWidget(self.resultTextEdit)
68	73		
69	74		vbox = QVBoxLayout()
70	75		
71	76		vbox.addLayout(hbox)
72	77		vbox.addLayout(hbox2)
73	78		vbox.addLayout(hbox3)
74	79		vbox.addLayout(hbox4)
75	80		vbox.addLayout(hbox5)

- QHBoxLayout 클래스를 사용해서 hbox1 - 5 의 수평 레이아웃을 5 개 생성합니다.
- line 50, 52 에서 addStretch 를 사용함으로써 window 에서 hbox2, 3 가 오른쪽으로 정렬합니다.
- 각 행에 라벨과 입력창을 추가합니다.
- QVBoxLayout 클래스를 사용해서 vbox 수직 레이아웃을 생성합니다.
- vbox 에 hbox1 -5 를 추가시킵니다. 그러면 하나의 수직 레이아웃에 5 개의 수평 레이아웃이 추가됩니다.

6. 버튼 클릭 시 이벤트 연결

```
79 - addButton.clicked.connect(self.addDB)
80 - delButton.clicked.connect(self.delDB)
81 - findButton.clicked.connect(self.findDB)
82 - incButton.clicked.connect(self.incDB)
83 - showButton.clicked.connect(self.showDB)

84 + #connect
85 + self.addButton.clicked.connect(self.addDB)
86 + self.delButton.clicked.connect(self.delDB)
87 + self.findButton.clicked.connect(self.findDB)
88 + self.incButton.clicked.connect(self.incDB)
89 + self.showButton.clicked.connect(self.showDB)
```

- 버튼 클릭 시 대응하는 함수를 실행시킵니다.

- 예로, line85 에서 addButton 을 클릭 시 addDB 함수를 실행시킵니다.

7. 변화된 데이터 저장

```
189 + def closeEvent(self, event):
190 +
191 +     self.writeScoreDB()
```

- 변화된 데이터 값을 저장하여 window 를 닫은 후 다시 열어 값을 읽을 때 변화된 값이 나오게 합니다.