



Seminario de Algoritmia

CLAVE: I59556

NRC: 59556

2022B

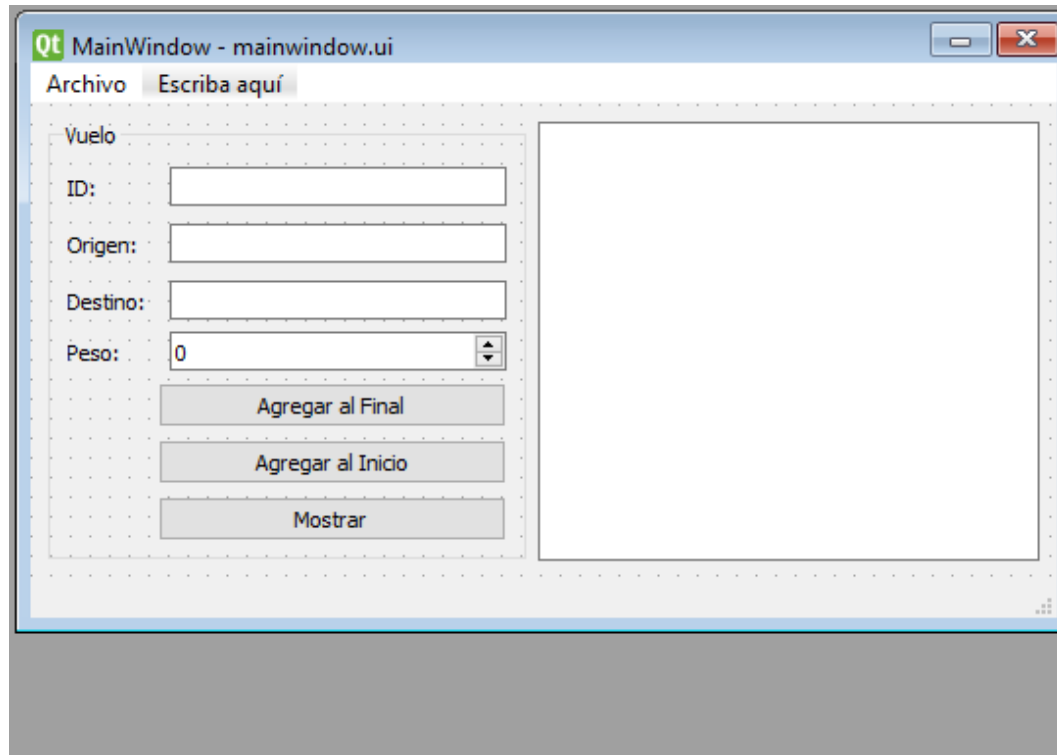
D14

Interfaz 4

Arellano Granados Angel Mariano

218123444

Interfaz Mejorada con nombre en los widget:



| Objeto | Clase |
|--------------------------|----------------|
| ▼ MainWindow | QMainWindow |
| ▼ centralwidget | QWidget |
| ▼ groupBox | QGroupBox |
| agrega_inicio_pushButton | QPushButton |
| agregar_final_pushButton | QPushButton |
| destino_lineEdit | QLineEdit |
| id_lineEdit | QLineEdit |
| label | QLabel |
| label_2 | QLabel |
| label_3 | QLabel |
| label_4 | QLabel |
| mostrar_pushButton | QPushButton |
| origen_lineEdit | QLineEdit |
| peso_spinBox | QSpinBox |
| salida | QPlainTextEdit |
| ▼ menubar | QMenuBar |
| ▼ menuArchivo | QMenu |
| actionAbrir | QAction |
| actionGuardar | QAction |
| statusbar | QStatusBar |

Mainwindow.py:

```

mainwindow.py 8 X
mainwindow.py > ...
1  from PySide2.QtWidgets import QMainWindow
2  from PySide2.QtCore import Slot
3  from ui_mainwindow import Ui_MainWindow
4  from aeropuerto import Aeropuerto
5  from vuelo import Vuelo
6  from PySide2.QtWidgets import QMainWindow, QFileDialog, QMessageBox
7
8  class MainWindow(QMainWindow):
9      def __init__(self):#reserva memoria para mostrar ventana
10         super(MainWindow,self).__init__()#constructor para reservar memoria
11         #clase hija
12
13         self.aeropuerto = Aeropuerto()
14
15         self.ui = Ui_MainWindow()
16         self.ui.setupUi(self)
17
18         self.ui.agregar_final_pushButton.clicked.connect(self.click_agregar)
19         self.ui.agrega_inicio_pushButton.clicked.connect(self.click_agregar_inicio)
20         self.ui.mostrar_pushButton.clicked.connect(self.click_mostrar)
21
22         self.ui.actionAbrir.triggered.connect(self.action_Abrir_Archivo)
23         self.ui.actionGuardar.triggered.connect(self.action_Guardar_Archivo)
24
25         @Slot( )
26         def click_agregar(self):
27             id = self.ui.id_lineEdit.text()
28             origen = self.ui.origen_lineEdit.text()
29             destino = self.ui.destino_lineEdit.text()
30             peso = self.ui.peso_spinBox.value()
31
32             vuelo = Vuelo(id,origen,destino,peso)
33             self.aeropuerto.agrega_final(vuelo)
34             #print(id,origen,destino,peso)
35             #self.ui.salida.insertPlainText(id + origen + destino + str(peso))
36
37         @Slot( )
38         def click_agregar_inicio(self):
39             id = self.ui.id_lineEdit.text()

```

```

39     id = self.ui.id_lineEdit.text()
40     origen = self.ui.origen_lineEdit.text()
41     destino = self.ui.destino_lineEdit.text()
42     peso = self.ui.peso_spinBox.value()
43
44     vuelo = Vuelo(id,origen,destino,peso)
45     self.aeropuerto.agrega_inicio(vuelo)
46
47     @Slot( )
48     def click_mostrar(self):
49         self.ui.salida.clear()
50         self.ui.salida.insertPlainText(str(self.aeropuerto))
51         #self.aeropuerto.mostrar()
52
53     @Slot( )
54     def action Abrir Archivo(self):
55         ubicacion = QFileDialog.getOpenFileName(
56             self,
57             'Guardar Archivo',
58             '.',
59             'JSON (*.json)'
60         )[0]
61         if self.aeropuerto.abrir(ubicacion):
62             QMessageBox.information(
63                 self,
64                 "Exito",
65                 "Se pudo abrir el archivo " + ubicacion
66             )
67         else:
68             QMessageBox.information(
69                 self,
70                 "Error",
71                 "No se pudo abrir el archivo " + ubicacion
72             )
73         self.aeropuerto.abrir(ubicacion)
74         #print("Abrir_Archivo")

```

```

75
76     @Slot( )
77     def action Guardar Archivo(self):
78         ubicacion = QFileDialog.getSaveFileName(
79             self,
80             'Guardar Archivo',
81             '.',
82             'JSON (*.json)'
83         )[0]
84         print(ubicacion)
85         if self.aeropuerto.guardar(ubicacion):
86             QMessageBox.information(
87                 self,
88                 "Exito",
89                 "Se pudo crear el archivo " + ubicacion
90             )
91         else:
92             QMessageBox.information(
93                 self,
94                 "Error",
95                 "No se pudo crear el archivo " + ubicacion
96             )
97         self.aeropuerto.guardar(ubicacion)
98         #print("Guardar_Archivo")
99

```

Función str en la clase aeropuerto:

```
aeropuerto.py X
aeropuerto.py > ...
1  from asyncore import write
2  #from vuelo import Vuelo
3  import json
4
5  from vuelo import Vuelo
6
7  class Aeropuerto:
8      def __init__(self):
9          self.__vuelos = []
10
11     def agrega_final(self,vuelo:Vuelo):
12         self.__vuelos.append(vuelo)
13
14     def agrega_inicio (self,vuelo:Vuelo):
15         self.__vuelos.insert(0,vuelo)
16
17     def mostrar(self):
18         for v in self.__vuelos:
19             print(v)
20
21     def __str__(self) -> str:
22         return"".join(
23             str(v) + "\n" for v in self.__vuelos #programacion Funcional
24         )
```

Prueba de la interfaz:

The image shows a Python IDE with a dark theme. The left pane displays the code for `main.py`, which is a Qt application. The code imports `QMainWindow` and `QApplication` from `PySide2.QtWidgets`, and `MainWindow` from a local `mainwindow` module. It also imports `sys`. The application is created with `app = QApplication()`, a `MainWindow` instance is created and shown with `window.show()`, and the Qt loop is started with `sys.exit(app.exec_())`.

The right pane shows a running window titled `MainWindow`. The window has a title bar with standard minimize, maximize, and close buttons. The main content area is titled `Archivo` and contains a form for adding flight records. The form has four input fields: `ID:` (text box with `9012`), `Origen:` (text box with `Vuelo 3`), `Destino:` (text box with `Vuelo 3`), and `Peso:` (spin box with `3`). Below these fields are three buttons: `Agregar al Final`, `Agregar al Inicio`, and `Mostrar`. To the right of the form, there is a list of three flight records, each with its identifier, origin, destination, and weight.

```
main.py 3 X
main.py > ...
1  from PySide2.QtWidgets import QMainWindow, QApplication
2  from mainwindow import MainWindow
3  import sys
4
5  #Aplicacion de Qt
6  app = QApplication()
7
8  #Se crea una ventana vacia
9  window = MainWindow()
10
11 #Se hace visible la ventana
12 window.show()
13
14 #Qt loop
15 sys.exit(app.exec_())
```

MainWindow

Archivo

Vuelo

ID: 9012

Origen: Vuelo 3

Destino: Vuelo 3

Peso: 3

Agregar al Final

Agregar al Inicio

Mostrar

Identificador: 5678
origen: Vuelo 2
Destino: Vuelo 2
Peso: 2

Identificador: 1234
origen: Vuelo 1
Destino: Vuelo 1
Peso: 1

Identificador: 9012
origen: Vuelo 3
Destino: Vuelo 3
Peso: 3