



UNIVERSIDAD DE GUAYAQUIL

FACULTAD DE CIENCIAS ADMINISTRATIVAS



CARRERA:
“GESTION DE LA INFORMACION GERENCIAL”

NOMBRE
REYES OBANDO DANIELA ALEJANDRA

MATERIA:
PROGRAMACION ORIENTADA A OBJETOS

TITULO:
IMPLEMENTAR LA VALIDACIÓN DEL EMAIL

DOCENTE:
ING. GUILLERMO VALAREZO GUZMÁN

CURSO:
3-1



Widget Box

Filter

- Layouts
 - Vertical Layout
 - Horizontal Layout
 - Grid Layout
 - Form Layout
- Spacers
- Buttons
 - Push Button
 - Tool Button
 - Radio Button
 - Check Box
 - Command Link Button
 - Dialog Button Box
- Item Views (Model-Based)
 - List View
 - Tree View
 - Table View
 - Column View
- Item Widgets (Item-Based)
 - List Widget
 - Tree Widget
 - Table Widget
- Containers
 - Group Box
 - Scroll Area
 - Tool Box
 - Tab Widget
 - Stacked Widget
- Frame
- Widget
- MDI Area
- Dock Widget
- Input Widgets
 - Combo Box
 - Font Combo Box
 - Line Edit

MainWindow - vtnPrincipal.ui

Type Here

NOMBRES: APELLIDOS: CEDULA: SEXO: SeleccionarEMAIL:

GUARDAR

LIMPIAR

Object Inspector

Object	Class
vtnPrincipal	QMainWindow
centralwidget	QWidget
LbApellido	QLabel
LbCedula	QLabel
LbEmail	QLabel
LbNombre	QLabel

Property Editor

Property	Value
objectName	vtnPrincipal
enabled	<input checked="" type="checkbox"/>
geometry	[0, 0, 726 x 397]
sizePolicy	[Preferred, Preferred, 0, 0]
minimumSize	0 x 0
maximumSize	16777215 x 16777215
sizeIncrement	0 x 0
baseSize	0 x 0
palette	Inherited
font	A [MS Shell Dlg 2, 8]
cursor	Arrow
mouseTracking	<input type="checkbox"/>
tabletTracking	<input type="checkbox"/>

Resource Browser

Filter
<resource root>

Signal/Slot Editor Action Editor Resource Browser



PR PROYECTODAN master Start Free Trial

Project + ⌂ X : SERVICIOS persona.py main.py DOMINIO persona.py 2 35

PROYECTODAN C:\POO\QT DESIGNER\PROYECTODAN .venv DATOS DOMINIO persona.py SERVICIOS persona.py UI vtnPrincipal.py vtnPrincipal.ui main.py External Libraries Scratches and Consoles

```
from PySide6.QtGui import QIntValidator
from PySide6.QtWidgets import QMainWindow, QMessageBox
from DOMINIO.persona import Persona
from UI.vtnPrincipal import Ui_vtnPrincipal

class PersonaServicio(QMainWindow): 3 usages & Daniela Reyes
    """ clase que genera la logica de los objetos persona """

    def __init__(self): & Daniela Reyes
        super(PersonaServicio, self).__init__()
        self.ui = Ui_vtnPrincipal()
        self.ui.setupUi(self)
        self.ui.btnGuardar.clicked.connect(self.guardar)
        self.ui.btnLimpiar.clicked.connect(self.limpiar)
        self.ui.txtCedula.setValidator(QIntValidator())

    def guardar(self): 1 usage & Daniela Reyes
        nombre = self.ui.txtNombre.text()
        apellido = self.ui.txtApellido.text()
        cedula = self.ui.txtCedula.text()
        sexo = self.ui.cbSexo.currentText()
        email = self.ui.txtEmail.text()

        # validacion de datos del formulario
        if nombre == "":
            QMessageBox.warning(self, title: 'Advertencia', text: 'Debe ingresar el nombre')
        elif apellido == "":
            QMessageBox.warning(self, title: 'Advertencia', text: 'Debe ingresar el apellido')
```

PROYECTODAN > UI > vtnPrincipal.py 8:1 CRLF UTF-8 4 spaces Python 3.13 (PROYECTODAN)

30°C Mayor nublado Buscar 16:33 24 ene 2026

PR PROYECTODAN master Start Free Trial

Project + ⊕ ^ X : SERVICIOS persona.py main.py DOMINIO persona.py

PROYECTODAN .venv DATOS DOMINIO SERVICIOS UI vtnPrincipal.py vtnPrincipal.ui main.py External Libraries Scratches and Consoles

```
9     class PersonaServicio(QMainWindow): 3 usages & Daniela Reyes
10    def guardar(self): 1 usage & Daniela Reyes
11        elif len(cedula) < 10:
12            QMessageBox.warning(self, title: 'Advertencia', text: 'Debe ingresar la cedula')
13        elif sexo == "Seleccionar":
14            QMessageBox.warning(self, title: 'Advertencia', text: 'Debe seleccionar el sexo')
15        elif email == "":
16            QMessageBox.warning(self, title: 'Advertencia', text: 'Debe ingresar el email')
17        else:
18            try:
19                persona = Persona(cedula=cedula, nombre=nombre, apellido=apellido, sexo=sexo, email=email)
20                print(persona)
21
22                # mostrar confirmacion de que se guardo
23                self.ui.statusbar.showMessage('Se guardó la información', 3000)
24
25
26                # Borrar los campos del formulario
27                self.ui.txtNombre.setText('')
28                self.ui.txtApellido.setText('')
29                self.ui.txtCedula.setText('')
30                self.ui.txtEmail.setText('')
31                self.ui.cbSexo.setCurrentText('Seleccionar')
32
33
34            except ValueError as e:
35                # Captura errores de validación de la clase Persona
36                QMessageBox.warning(self, title: 'Error de validación', str(e))
37
38
39        def limpiar(self): 1 usage & Daniela Reyes
40            self.ui.txtNombre.clear()
41            self.ui.txtApellido.clear()
42            self.ui.txtCedula.clear()
43            self.ui.txtEmail.clear()
```

PROYECTODAN > UI > vtnPrincipal.py 8:1 CRLF UTF-8 4 spaces Python 3.13 (PROYECTODAN)

30°C Mayor m. nublado Buscar PC Q ESP LAA 16:34 24 ene 2026

PR PROYECTODAN master Start Free Trial

Project

- PROYECTODAN (POO\QT DESIGNER\PROYECTODAN)
 - .venv
 - DATOS
 - DOMINIO
 - persona.py
 - SERVICIOS
 - persona.py
 - UI
 - vtnPrincipal.py
 - vtnPrincipal.ui
 - main.py
- External Libraries
- Scratches and Consoles

SERVICIOS\persona.py main.py DOMINIO\persona.py

```
3     class Persona: 2 usages & Daniela Reyes
7         def __init__(self, cedula: str, nombre: str, apellido: str, sexo: str, email: str = None): & Daniela Reyes
11            self.sexo = sexo
12            self.email = email
13
14        # Cedula
15        @property 2 usages & Daniela Reyes
16        def cedula(self):
17            return self._cedula
18
19        @cedula.setter 1 usage & Daniela Reyes
20        def cedula(self, valor):
21            if not valor.isdigit():
22                raise ValueError("La cédula debe contener solo números")
23            self._cedula = valor
24
25        # Nombre
26        @property 2 usages & Daniela Reyes
27        def nombre(self):
28            return self._nombre
29
30        @nombre.setter 1 usage & Daniela Reyes
31        def nombre(self, valor):
32            if not valor.isalpha():
33                raise ValueError("El nombre debe contener solo letras")
34            self._nombre = valor.capitalize()
35
36        # Apellido
37        @property 2 usages & Daniela Reyes
38        def apellido(self):
39            return self._apellido
40
```

PROYECTODAN > DOMINIO > persona.py 71:32 CRLF UTF-8 4 spaces Python 3.13 (PROYECTODAN)

30°C Buscar 16:34
Mayorm. nublado ESP LAA 24 ene 2026

A screenshot of a Windows desktop environment. The main window is a code editor for Python, specifically PyCharm, showing the file `DOMINIO\persona.py`. The code defines a class `Persona` with properties for apellido, sexo, and email, each with its own setter method that includes validation logic. The PyCharm interface includes a left sidebar for navigating the project structure, which shows a folder named `PROYECTODAN` containing subfolders like `.venv`, `DATOS`, `DOMINIO`, `SERVICIOS`, and `UI`, along with files such as `vtnPrincipal.py` and `main.py`. The status bar at the bottom provides information about the current file, encoding, and system status.

```
3     class Persona: 2 usages & Daniela Reyes
40
41         @apellido.setter 1 usage & Daniela Reyes
42         def apellido(self, valor):
43             if not valor.isalpha():
44                 raise ValueError("El apellido debe contener solo letras")
45             self._apellido = valor.capitalize()
46
47         # Sexo
48         @property 2 usages & Daniela Reyes
49         def sexo(self):
50             return self._sexo
51
52         @sexo.setter 1 usage & Daniela Reyes
53         def sexo(self, valor):
54             if valor.lower() not in ["masculino", "femenino", "prefiero no decirlo"]:
55                 raise ValueError("Sexo debe ser 'Masculino', 'Femenino' o 'Prefiero no decirlo'")
56             self._sexo = valor.lower()
57
58         # Email
59         @property 2 usages & Daniela Reyes
60         def email(self):
61             return self._email
62
63         @email.setter 1 usage & Daniela Reyes
64         def email(self, valor):
65             if valor is None:
66                 self._email = None
67             else:
68                 patron = r'^[\w\.-]+@[\\w\.-]+\.\w+$'
69                 if not re.match(patron, valor):
70                     raise ValueError("Email invalido, debe tener formato correcto")
```

PR PROYECTODAN master Start Free Trial

Project ▾

PROYECTODAN C:\POO\QT DESIGNER\PROYECTODAN

- .venv
- DATOS
- DOMINIO

 - persona.py

- SERVICIOS

 - persona.py

- UI

 - vtnPrincipal.py
 - vtnPrincipal.ui

External Libraries

Scratches and Consoles

SERVICIOS\persona.py main.py DOMINIO\persona.py

```
class Persona: 2 usages & Daniela Reyes
    def sexo(self, valor):
        if valor.lower() not in ["masculino", "femenino", "prefiero no decirlo"]:
            raise ValueError("Sexo debe ser 'Masculino', 'Femenino' o 'Prefiero no decirlo'")
        self._sexo = valor.lower()

    # Email
    @property 2 usages & Daniela Reyes
    def email(self):
        return self._email

    @email.setter 1 usage & Daniela Reyes
    def email(self, valor):
        if valor is None:
            self._email = None
        else:
            patron = r'^[\w\.-]+@[\\w\.-]+\.\w+$'
            if not re.match(patron, valor):
                raise ValueError("Email inválido, debe tener formato correcto")
            self._email = valor

    # Representación en cadena
    def __str__(self): & Daniela Reyes
        return f'Persona: {self.nombre} {self.apellido}, Cédula: {self.cedula}, Sexo: {self.sexo}, Email: {self.email}'
```

PROYECTODAN > DOMINIO > persona.py

71:32 CRLF UTF-8 4 spaces Python 3.13 (PROYECTODAN)

30°C Mayom, nublado Buscar 16:36

Start Free Trial

36

1 24 ene 2026

