

# Proyecto Fundamentos Básicos de Python

Loidher Steward Jaimes Delgado  
Docente: Alejandro Rodas Vásquez  
Universidad Tecnológica de Pereira

28 de febrero de 2025

## Introducción

En este proyecto usted pondrá en práctica los conceptos básicos vistos en esta unidad especialmente *arreglos*, *funciones*, *módulos*. Se deberá consumir los recursos suministrados por una API suministrada por el portal Nacional de [Datos Abiertos](#) y crear una Arquitectura de Software basada en módulos.

Durante esta pandemia la adquisición, almacenamiento y consulta de datos relacionado con el reporte de casos de COVID-19 se ha convertido en un insumo imprescindible para la creación de aplicaciones software que ayuden en la prevención y trazabilidad de la enfermedad.

El portal de Datos Abierto pone a disposición el conjunto de datos llamado [Casos positivos de COVID-19 en Colombia](#) (se recomienda que sea examinado) el cual puede ser consultado por aplicaciones web o de escritorio a través de consultas a su API. A continuación, se encuentra el código fuente en Python que permite realiza dicha consulta.

```

1 import pandas as pd
2 from sodapy import Socrata
3
4 # Unauthenticated client only works with public data sets. Note 'None'
5 # in place of application token, and no username or password:
6 client = Socrata("www.datos.gov.co", None)
7
8 # Example authenticated client (needed for non-public datasets):
9 # client = Socrata(www.datos.gov.co,
10 #                  MyAppToken,
11 #                  username="user@example.com",
12 #                  password="AFakePassword")
13
14 # First 2000 results, returned as JSON from API / converted to Python
15 # list of dictionaries by sodapy.
16 results = client.get("gt2j-8ykr", limit=limite_registros, departamento =
17                     nombre_departamento)
18
19 # Convert to pandas DataFrame
20 results_df = pd.DataFrame.from_records(results)

```

En las líneas 1 y 2 se hace la llamada a las librerías *Pandas* y *sodapy*. *Pandas* es reconocida por ser empleada en *Análisis de Datos*. En la línea 16 se encuentran los parámetros *limite\_registros* y *nombre\_departamento*. Estos parámetros son requeridos y como su nombre lo dice, permiten consultar el Departamento y el Número de Casos que se quieren extraer. Para mayor conocimiento sobre el API se tiene el sitio [Documentación sodapy](#).

## 1. Requerimientos Funcionales

Usted debe de crear una aplicación que le permita al usuario ingresar el Departamento (*nombre\_departamento*) que desea consultar y el Número de Registro (*limite\_registros*) que quiere obtener de la consulta (este parámetro es importante puesto que si selecciona un número grande e.j 1000 registros, la consulta puede “colgarse”).

El resultado obtenido debe de poder visualizarse en la pantalla en un formato (investigar la función *format*) que contenga solo las siguientes columnas: *Ciudad de ubicación*, *Departamento*, *Edad*, *Tipo*, *Estado* y *País de procedencia*.

## 2. Requerimientos de la Arquitectura de Software

El software debe de estar construido aplicando el concepto de *modularidad*. Cada módulo debe de tener una tarea en específico

- Módulo para UI
- Módulo API
- Como archivo separado el *main*

### 3. Evidencias para la entrega

1. Usted debe de subir el código fuente a su repositorio invidual de [github](#).
2. Pantallazos donde se corrobore el funcionamiento del software (consultas realizadas y resultado obtenido) y se compruebe que los Requerimientos Funcionales ([Sección 1](#)) han sido complidos.
3. Para que esta actividad esté enmarcada dentro del contexto de la Arquitectura de Software y tengo un fundamento teórico válido. Usted debe de consultar ¿Qué es un Diagrama de Componentes? ¿Para qué se utiliza? ¿Qué es un componente?

#### 3.1. Estructura del Proyecto

A continuación, se muestra la arquitectura que debe de tener el software ([Figura 1](#)). Sin embargo, en esta imagen solo se muestra la estructura base, se debe agregar los archivos Python que usted crea.

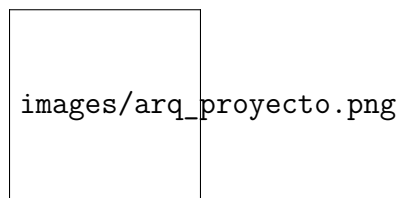


Figura 1: Estructura de la Arquitectura

#### 3.2. Pantallazos

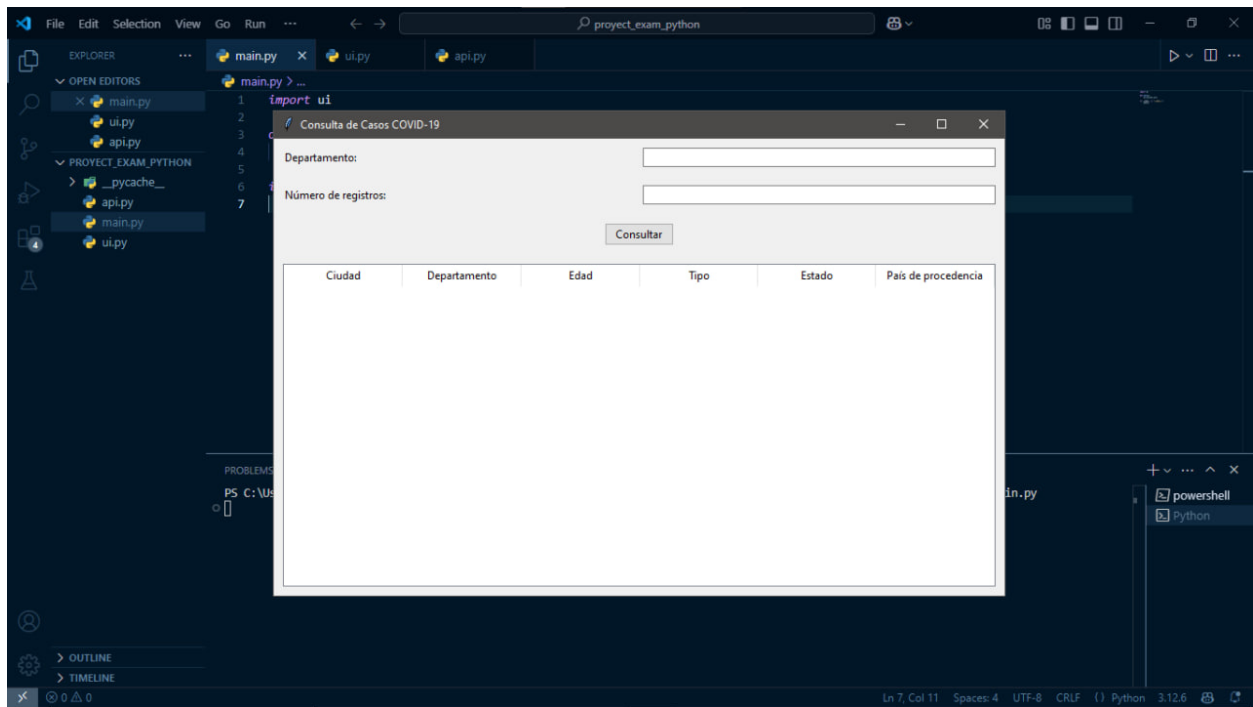


Figura 2: Ventana Principal (UI)

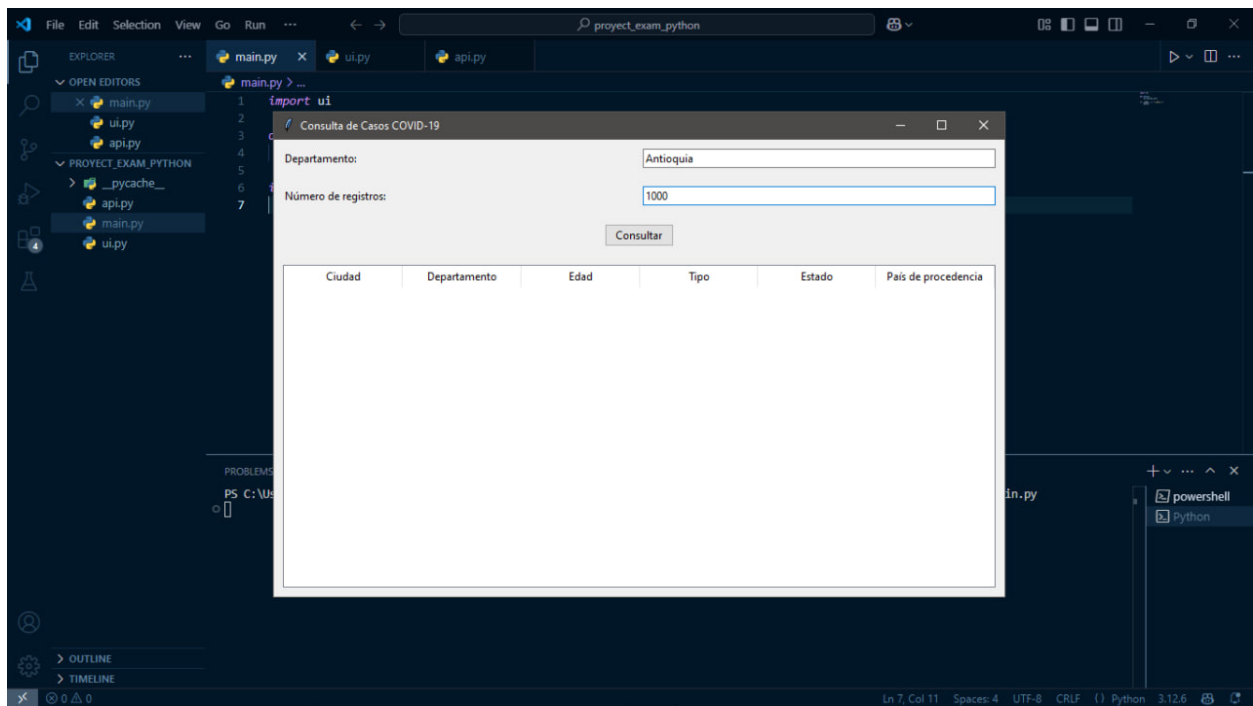


Figura 3: Entrada de datos

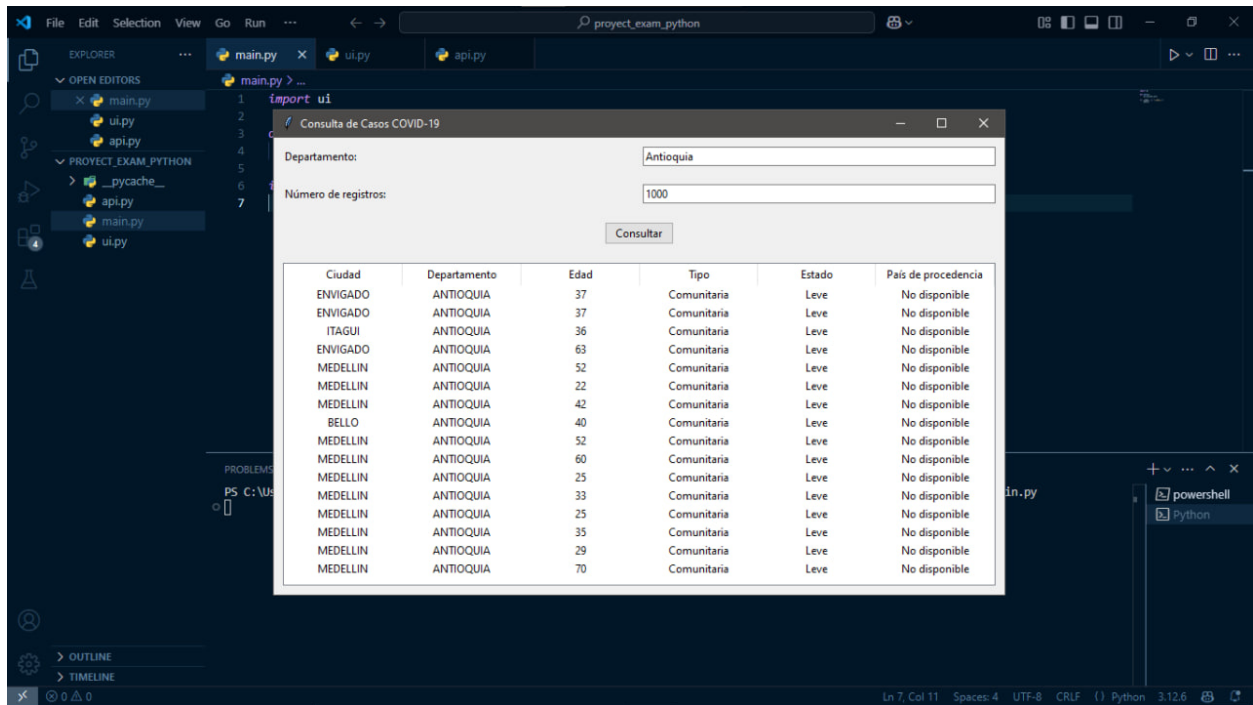


Figura 4: Consulta de Datos

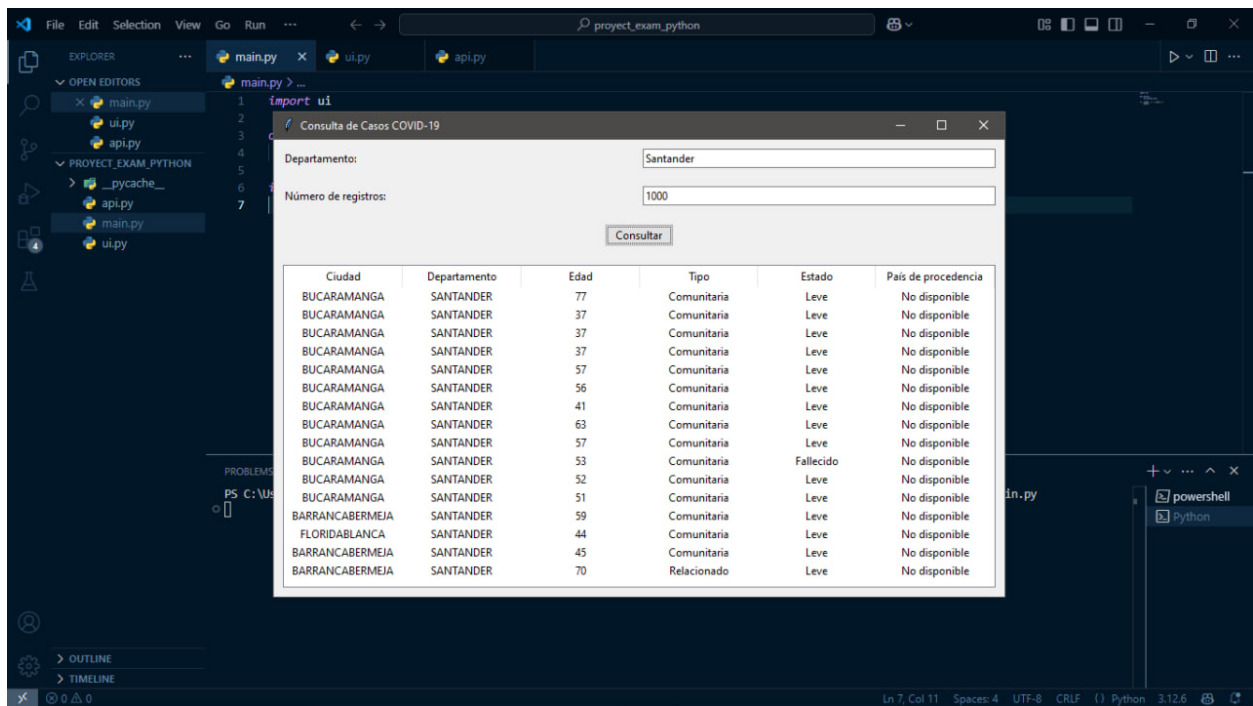


Figura 5: Segunda Consulta de Datos

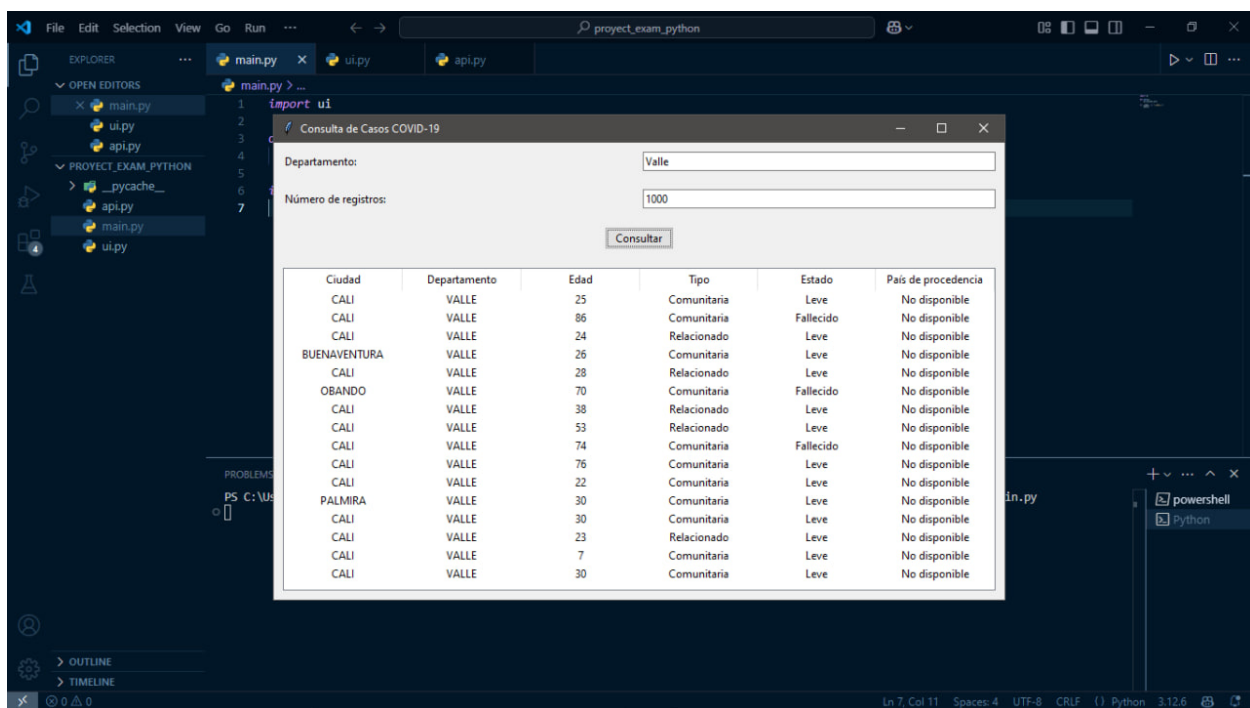


Figura 6: Tercera Consulta de Datos