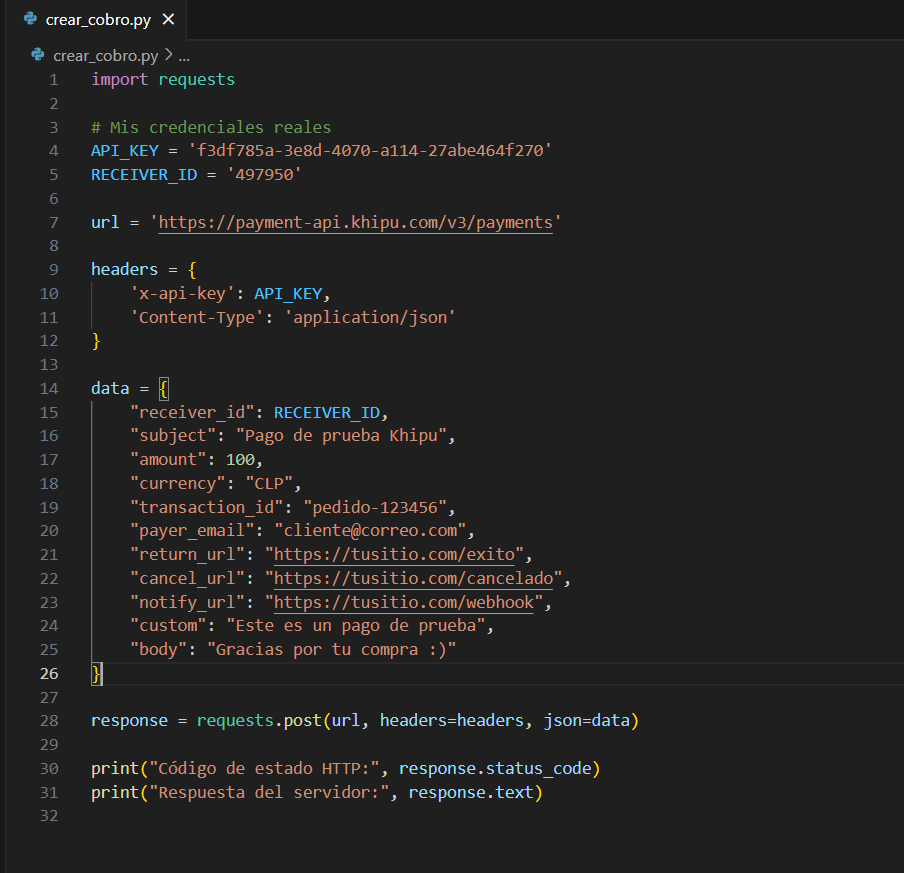
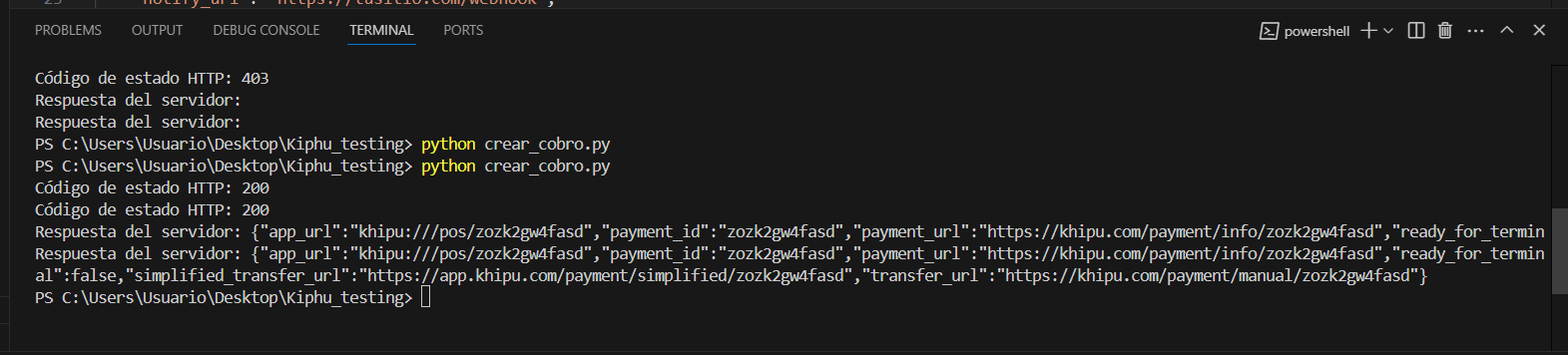
Documentación Integración Kiphu a través de llamadas a la API

Crear cobro

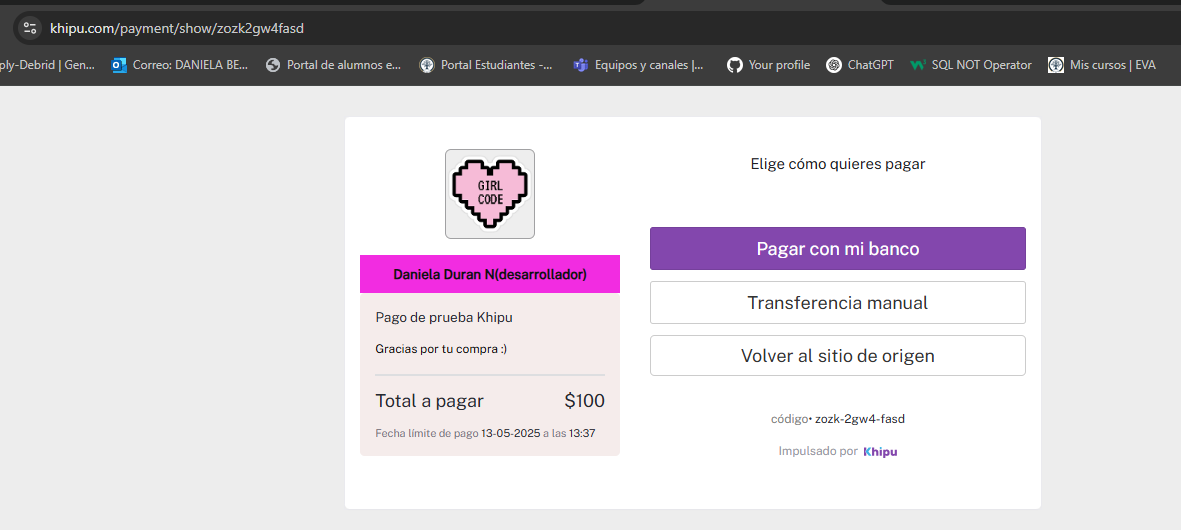


Creación de pago. Código exitoso

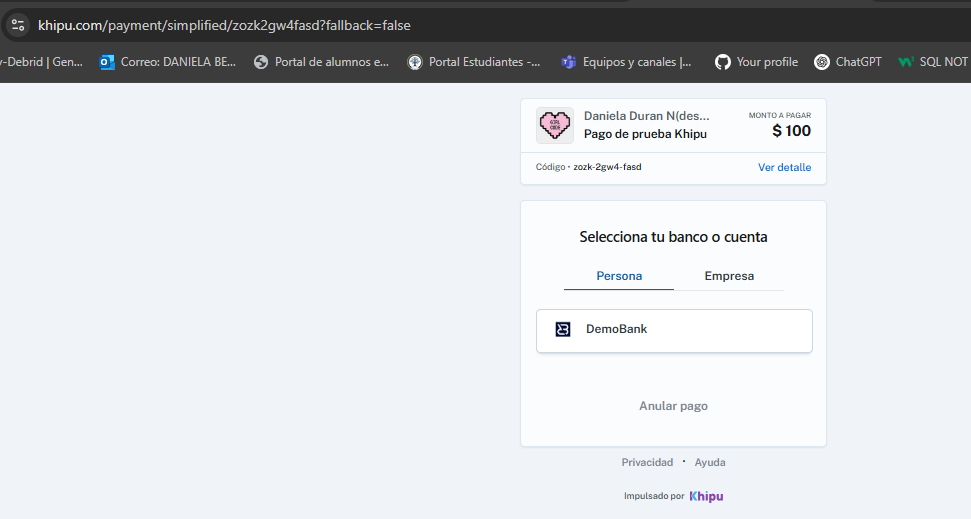


Probamos URL devuelta por la terminal

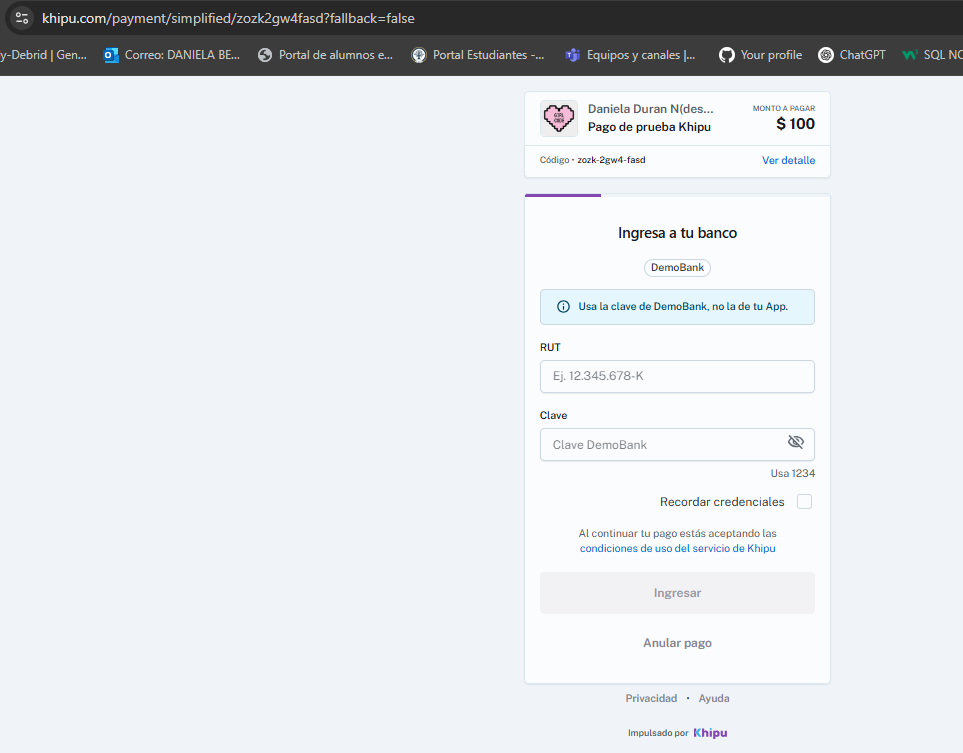
<https://khipu.com/payment/info/zozk2gw4fasd>



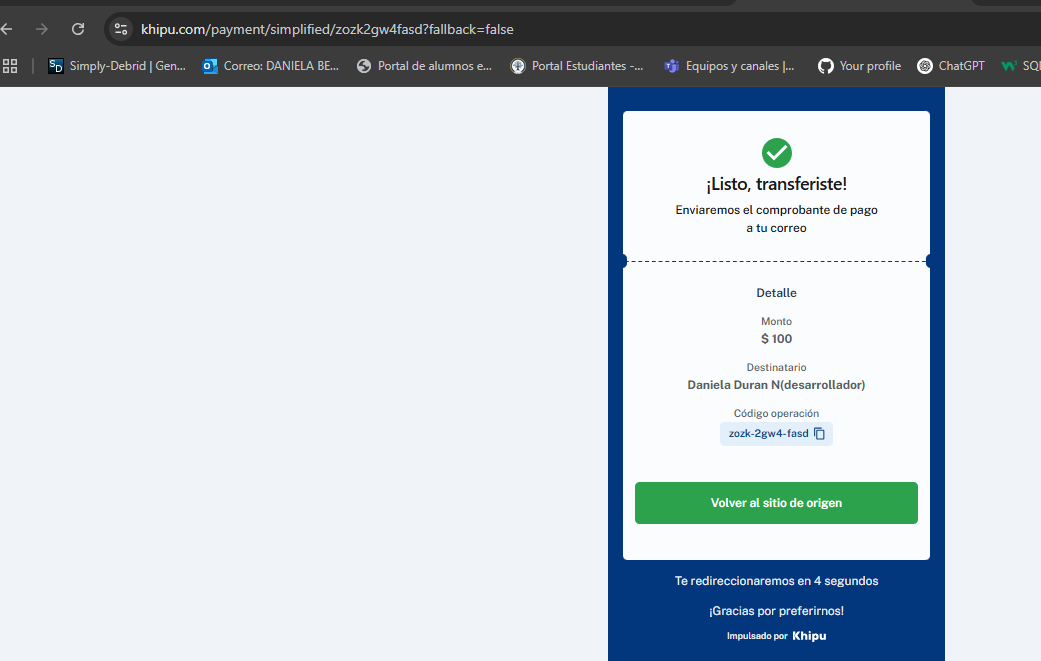
Opción pagar con mi banco:



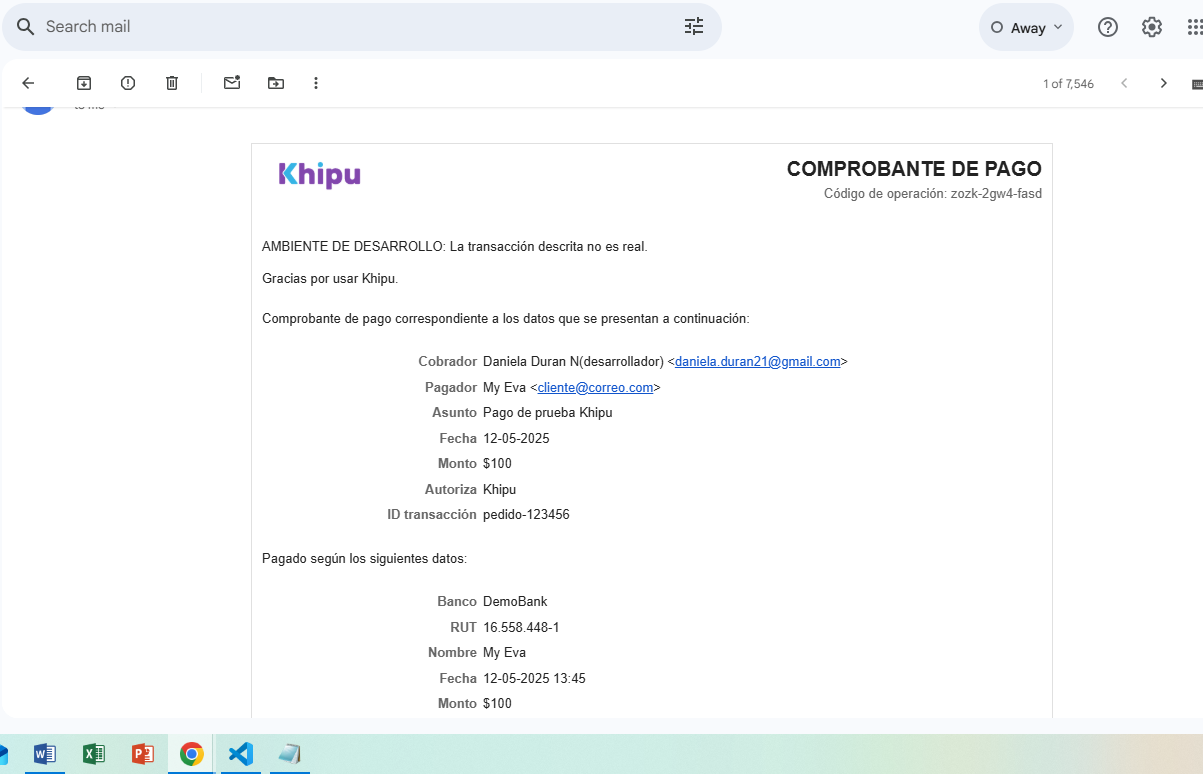
Elegimos opción demoBank nos redirige solicitud de credenciales para ingresar a DemoBank:



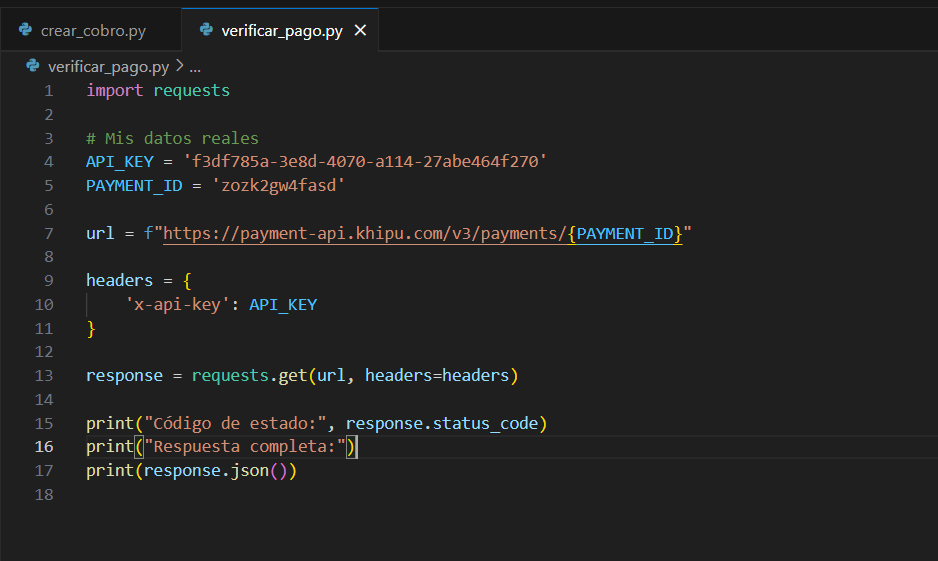
Ingreso datos y transfiero exitosamente:



Me llega un mensaje de correo notificando el pago exitoso:



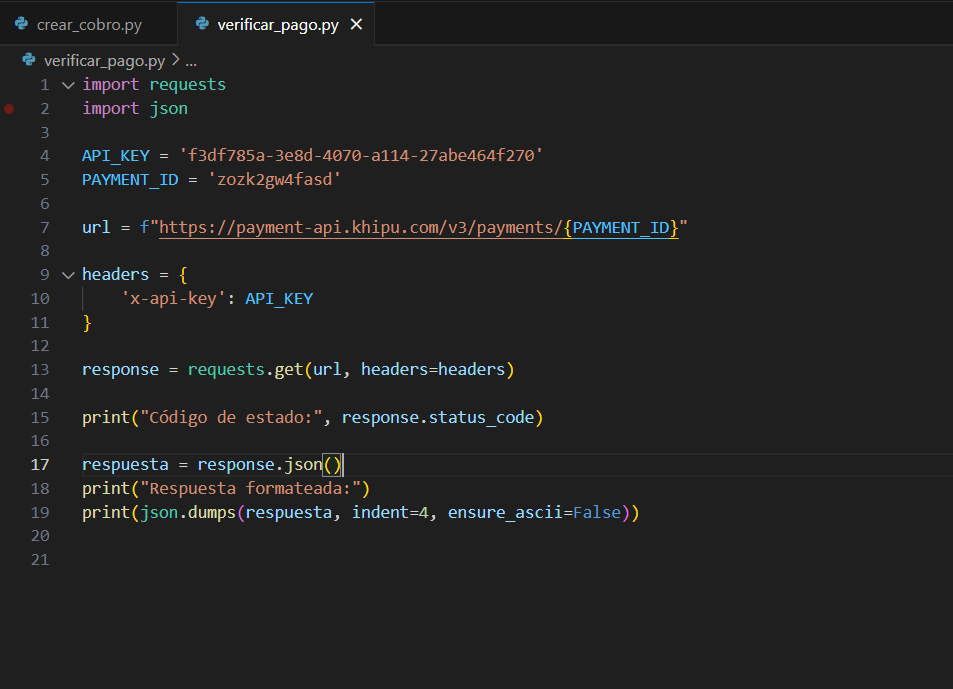
Para verificar pago creamos verificar\_pago.py

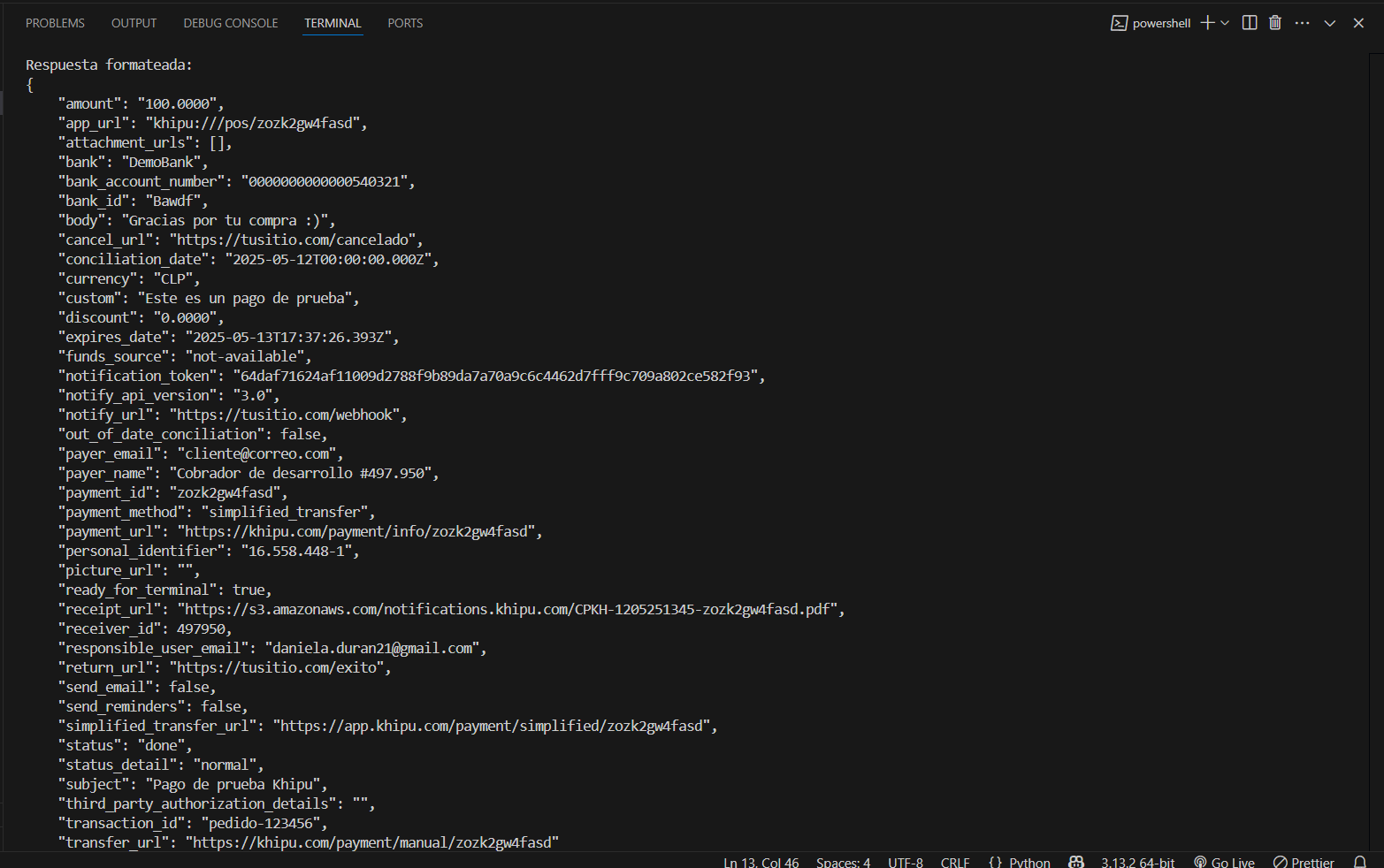


Respuesta en la terminal con un código 200 y status: done nos confirma el paso exitoso

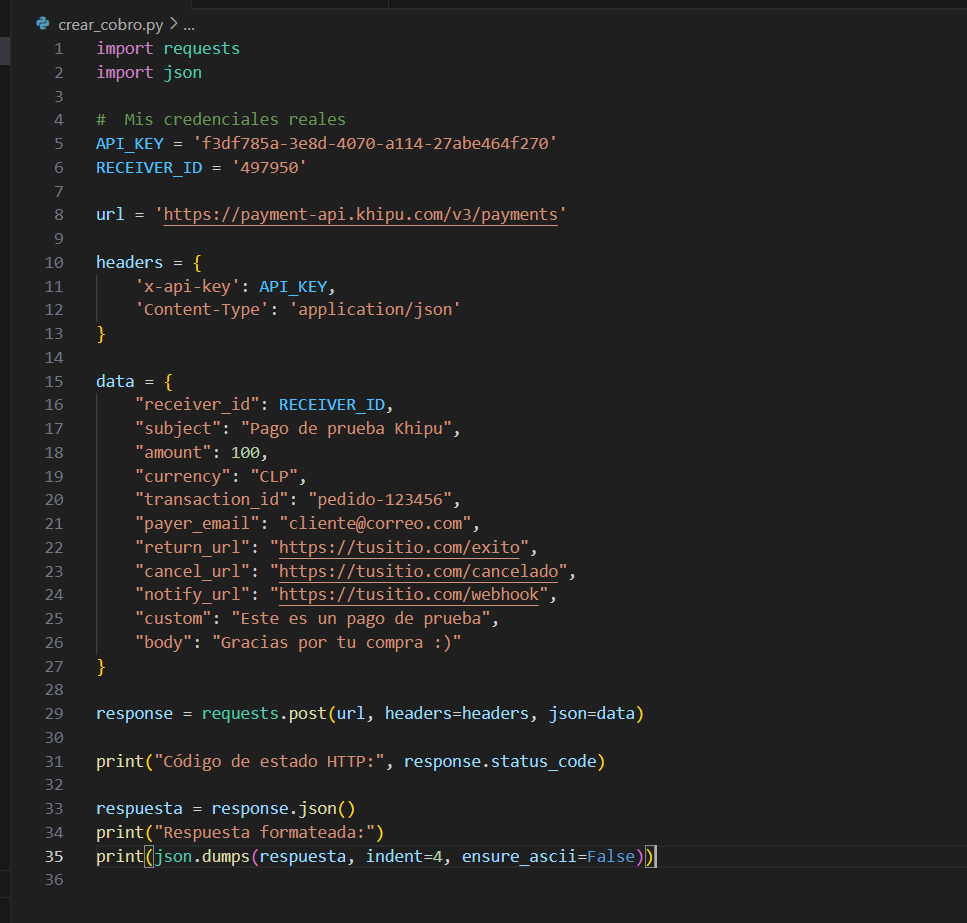


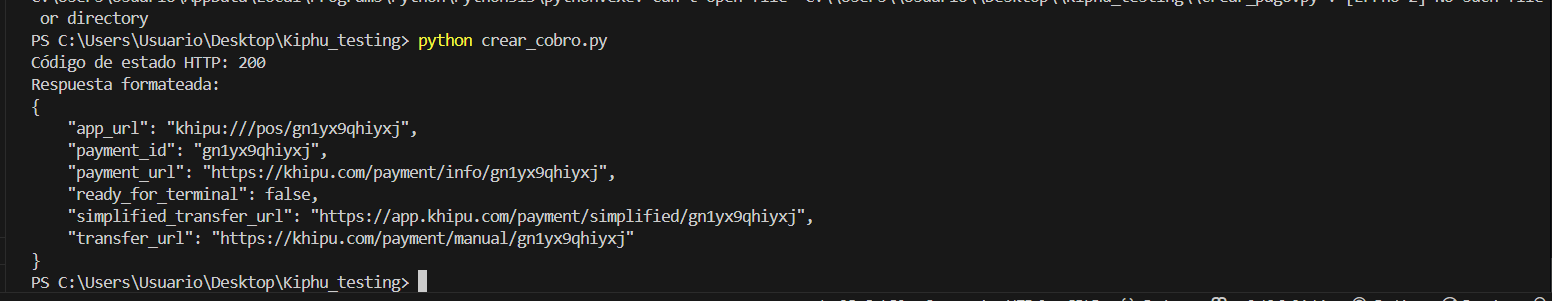
Mejoramos el aspecto visual de verificar\_pago.py con JSON





Lo mismo para crear\_cobro.py nos ordena mucho mejor la vista en la terminal.





En resumen:

1. Se creó cuenta en modo desarrollador en el portal de Kiphu siguiendo las instrucciones señaladas en el kit de implementación, se utilizó DemoBank como banco de pruebas y se obtuvieron las credenciales receiver\_id y API key válida para llamadas a la API (x-api-key).

2. Se implementó Python en la llamada POST/PAYMENTS para crear un cobro. No se utilizaron herramientas manuales desde el portal para generar datos. Se verificó que la integración responde con código HTTP 200 en la terminal.

3. Simulación de pago, se generó un payment\_url válido que se probó en el navegador y se simularon pagos con Demobank.

4. Se verifica el estado del pago al crear un segundo script verificar\_pago.py para consultar GET/payments/{payment\_id} de lo cual la API respondió con estado “done” lo que indica que el pago fue simulado con éxito.

Se mejoraron errores como 403 Forbidden y se corrigieron los headers y también se añadió formato JSON para una mayor legibilidad en la terminal con las respuestas.