

# Reporte de Desarrollo

## Introducción

Este reporte describe el proceso de desarrollo de un script en Python que lee una lista de correos de un .csv y que envíe un saludo en un correo electrónico

Se instalo las librerias pandas y smtplib para poder leer el archivo csv se utiliza la librería pandas y smtp para el envio de correos a continuación se describe el codigo realizado:

primero se configura variables necesarias para la integracion con gmail  
el password se recupera de una variable de entorno por seguridad

```
# Configuración del correo
smtp_server = "smtp.gmail.com" # Servidor SMTP de Gmail
smtp_port = 587                # Puerto de Gmail para TLS
sender_email = "crilobra@gmail.com" # Correo del remitente
sender_password = os.getenv('password') # Contraseña del remitente
```

lectura de datos usando libreria pandas

```
# Leer la lista de correos electrónicos desde el archivo CSV
contactos = pd.read_csv("lista.csv")
```

conexion servidor de correos

```
# Iniciar la conexión con el servidor SMTP
server = smtplib.SMTP(smtp_server, smtp_port)
server.starttls() # Protocolo de encriptación
server.login(sender_email, sender_password)
```

envio de correo personalizado creando un archivo pdf

```
def crearPdf(nombre, mensaje, nombreArchivo):
    # Configurar el lienzo del PDF
    c = canvas.Canvas(nombreArchivo, pagesize=letter)
    ancho, alto = letter

    # Añadir título de felicitación
    c.setFont("Helvetica-Bold", 24)
    c.setFillColor(colors.darkblue)
```

```

    c.drawCentredString(ancho / 2, alto - 100, f";Feliz Cumpleaños,
{nombre}!")

    # Añadir mensaje personalizado
    c.setFont("Helvetica", 14)
    c.setFill_color(colors.black)
    texto = c.beginText(100, alto - 150)
    texto.setFont("Helvetica", 14)
    texto.setFill_color(colors.black)
    texto.textLines(mensaje)
    c.drawText(texto)

    # Cerrar el PDF
    c.save()

```

## envio de correos

```

# Enviar el correo a cada contacto
for index, row in contactos.iterrows():
    msg = MIMEMultipart()
    msg['From'] = sender_email
    msg['To'] = row['email']
    msg['Subject'] = subject

    fecha = datetime.strptime(row['fecha'], "%Y/%m/%d")

    if fecha.day == fechaActual.day and fecha.month ==
fechaActual.month:
        # Nombre del archivo PDF
        nombreArchivo = "felicitacion.pdf"
        # Crear el PDF
        crearPdf(row['nombre'], row['mensaje'], nombreArchivo)
        # Adjuntar el PDF
        with open(nombreArchivo, "rb") as adjunto:
            part = MIMEBase("application", "octet-stream")
            part.set_payload(adjunto.read())
            encoders.encode_base64(part)
            part.add_header(
                "Content-Disposition",

```

```
        f"attachment; filename= {nombreArchivo}"
    )
    msg.attach(part)
    # Enviar el correo
    server.sendmail(sender_email, row['email'], msg.as_string())
    print(f"Correo enviado a {row['email']}")
    with open("registro_envios.csv", "a") as log:
        log.write(f"{row['email']}, Enviado\n")
    os.remove(nombreArchivo)
# Cerrar la conexión
server.quit()
print("Todos los correos han sido enviados.")
```

## Conclusiones

El script desarrollado permite leer un archivo csv y enviar un correos y registrar un log de envios