



Dependency Methods and Functions Explained

1. pandas

Used for **reading and processing the Excel file** (`employees.xlsx`).

✓ Common Methods:

- **`pd.read_excel('filename.xlsx')`**
Reads data from an Excel file into a DataFrame.

Python

```
df = pd.read_excel("employees.xlsx")
```

- **`df.iterrows()`**
Loops through each row in the DataFrame.

Python

```
for index, row in df.iterrows():  
    print(row['Name'], row['Salary'])
```

- **`df['column_name']`**
Accesses a column by name.

Python

```
salaries = df['Gross Salary']
```

2. **fpdf.FPDF** (or *reportlab*)

Used for **creating the payslip PDF**.

✓ Common Methods:

- **FPDF()**
Initializes a new PDF object.
- **.add_page()**
Adds a new page to the PDF document.
- **.set_font(font, style, size)**
Sets the font for the text.
- **.cell(width, height, txt, ln, align)**
Adds a text cell to the page (a line of text).

✓ Example:

```
Python
from fpdf import FPDF

pdf = FPDF()
pdf.add_page()
pdf.set_font("Arial", size=12)
pdf.cell(200, 10, txt="Payslip for John Doe", ln=True, align="C")
pdf.output("john_payslip.pdf")
```

3. **smtplib**

Used for **sending the payslip via email**.

✓ Common Functions:

- **smtplib.SMTP(server, port)**
Connects to the SMTP server (e.g. Gmail).

- **.starttls()**
Encrypts the connection.
- **.login(email, password)**
Logs into the SMTP server.
- **.sendmail(from, to, message)**
Sends the actual email.

✓ **Example:**

Python

```
import smtplib

server = smtplib.SMTP('smtp.gmail.com', 587)
server.starttls()
server.login("youremail@gmail.com", "yourpassword")
server.sendmail(from_addr, to_addr, message)
server.quit()
```

4. **email** module

Used for **creating and attaching content to your email**.

✓ **Useful Classes and Functions:**

- **EmailMessage()**
Used to construct the email (subject, body, attachment).
- **.set_content()**
Sets the body of the email.
- **.add_attachment()**
Attaches files like PDFs to the email.

✓ **Example:**

Python

```
from email.message import EmailMessage
```

```
msg = EmailMessage()
msg["Subject"] = "Payslip"
msg["From"] = sender_email
msg["To"] = receiver_email
msg.set_content("Attached is your payslip.")
with open("john_payslip.pdf", "rb") as f:
    msg.add_attachment(f.read(), maintype="application", subtype="pdf",
filename="john_payslip.pdf")
```

5. **yagmail** (*optional, simplified email sending*)

An easier alternative to `smtplib` + `email`.

✓ Main Functions:

- **yagmail.SMTP(email, password)**
Logs in with your credentials.
- **.send(to, subject, contents, attachments)**
Sends the email with the specified content and files.

✓ Example:

Python

```
import yagmail
```

```
yag = yagmail.SMTP("youremail@gmail.com", "yourpassword")
yag.send(
    to="someone@example.com",
    subject="Your Payslip",
    contents="Please find your payslip attached.",
    attachments="john_payslip.pdf"
)
```

6. **os**

Used for **managing folders and paths**.

✓ **Common Methods:**

- **os.mkdir("folder_name")**
Creates a new directory.
- **os.path.exists(path)**
Checks if a path or folder exists.

✓ **Example:**

```
Python
import os

if not os.path.exists("payslips"):
    os.mkdir("payslips")
```

7. **dotenv** (optional)

To load environment variables from a **.env** file for security.

✓ **Common Functions:**

- **load_dotenv()**
Loads all variables from the **.env** file.
- **os.getenv("VAR_NAME")**
Gets the value of the environment variable.

✓ **Example:**

```
Python
from dotenv import load_dotenv
import os
```

```
load_dotenv()  
email = os.getenv("EMAIL_ADDRESS")  
password = os.getenv("EMAIL_PASSWORD")
```
