```
In [1]: # Extract data from amazon invoice pdfs and save each pdf file data into the seprate excel file

        import fitz
        import pandas as pd
        import re

        k = {}
        def pdf_to_excel(pdf_path, excel_path):
            # Open the PDF file
            pdf_document = fitz.open(pdf_path)

            # Iterate through each page in the PDF

            page = pdf_document[0]
            text = page.get_text()
            ts = text

            sold_by_pattern = re.compile(r'Sold By\s*:\s*([\s\S]*?)\*')
            sold_by_match = sold_by_pattern.search(text)
            sold_by = sold_by_match.group(1).strip() if sold_by_match else None

            sold_address_pattern = re.compile(r'Sold By\s*:\s*([\s\S]*?)IN')
            sold_address_match = sold_address_pattern.search(text)
            address = sold_address_match.group(1).strip() if sold_address_match else None

            so = sold_by+address
            k['Sold By'] =  so.replace('\n','')


            Billing_Address_pattern = re.compile(r'Billing Address\s*:\s*([\s\S]*?)IN')
            Billing_Address_match = Billing_Address_pattern.search(text)
            Billing_Address = Billing_Address_match.group(1).strip() if Billing_Address_match else None

            k['Billing_Address'] = Billing_Address.replace('\n','')


            PAN_No_pattern = re.compile(r'PAN No:\s*([A-Z0-9]+)')
            PAN_No_match =PAN_No_pattern.search(text)
            PAN_No = PAN_No_match.group(1).strip() if PAN_No_match else None

            k['PAN_No'] = PAN_No

            GST_Registration_No_pattern = re.compile(r'GST Registration No:\s*([A-Z0-9]+)')
            GST_Registration_No_match =GST_Registration_No_pattern.search(text)
            GST_Registration_No = GST_Registration_No_match.group(1).strip()

            k['GST_Registration_No'] = GST_Registration_No

            Order_Number_pattern = re.compile(r'Order Number:\s*([A-Z0-9-]+)')
            Order_Number_match =Order_Number_pattern.search(text)
            Order_Number = Order_Number_match.group(1).strip()

            k['Order_Number'] = Order_Number

            Order_Date_pattern = re.compile(r'Order Date:\s*([A-Z0-9.]+)')
            Order_Date_match =Order_Date_pattern.search(text)
            Order_Date = Order_Date_match.group(1).strip()

            k['Order_Date'] = Order_Date

            Invoice_Number_pattern = re.compile(r'Invoice Number\s*:\s*([A-Z0-9-]+)')
            Invoice_Number_match =Invoice_Number_pattern.search(text)
            Invoice_Number = Invoice_Number_match.group(1).strip()
```

```
    k['Invoice_Number'] = Invoice_Number

    Invoice_Details_pattern = re.compile(r'Invoice Details\s*:\s*([A-Z0-9-]+)')
    Invoice_Details_match =Invoice_Details_pattern.search(text)
    Invoice_Details = Invoice_Details_match.group(1).strip()

    k['Invoice_Details']  = Invoice_Details


    item_pattern = re.compile(r'(\d+)\s(.+?)\s\|\sB[0-9A-Z]+\s\(\s([^\s]+)\s\)\s+HSN:(\d+)\s+₹([\d,.]+)\s(\d+)\s₹([\d,.]+)\s(\d+)%\s(\w+)\s₹([\d,.]+)\s₹([\d,.]+)')

    # Extract information using regular expressions
    item_match = item_pattern.findall(text)

    for sl_no, description, product_code, hsn, unit_price, quantity, net_amount, tax_rate, tax_type, tax_amount, total_amount in item_match:
        k['Sl. No']=sl_no
        k['Description']= description
        k['Product Code']= product_code
        k['HSN']= hsn
        k['Unit Price']= unit_price
        k['Quantity']= quantity
        k['Net Amount']= net_amount
        k['Tax Rate']= tax_rate
        k['Tax Type']= tax_type
        k['Tax Amount']= tax_amount
        k['Total Amount']=total_amount

    df = pd.DataFrame([k])
    df.to_excel(excel_file_path, index=False)


pdf_file_path = 'img1.pdf'
excel_file_path = 'output.xlsx'

# Convert PDF to Excel
for i in range(1,3):
    pdf_file_path = 'img'+str(i)+'.pdf'
    excel_file_path = 'output'+ str(i) + '.xlsx'
    print(pdf_file_path)
    pdf_to_excel(pdf_file_path, excel_file_path)
```

```
img1.pdf
img2.pdf
```

In [2]:

```
pd.read_excel('output1.xlsx')
```

Out[2]:

	Sold By	Billing_Address	PAN_No	GST_Registration_No	Order_Number	Order_Date	Invoice_Number	Invoice_Details	Sl. No	Description	Product Code	HSN	Unit Price	Quantity	Net Amount	Tax Rate	Tax Type	Tax Amount
0	Darshita Aashiyana Private LimitedDarshita Aas...	karthikC1001 ace city, sector1GREATER NOIDA, U...	AAFCD6883Q	06AAFCD6883Q1ZU	407-8153595-7245952	10.08.2019	DEL2-68786	HR-DEL2-179184911-1920	1	OnePlus 7 (Mirror Blue, 6GB RAM, 128GB Storage)	OP7-NBLUE-6-128GB	8517	29,463.39	1	29,463.39	12	IGST	3,535.61

In [3]:

```
pd.read_excel('output2.xlsx')
```

Out[3]:

	Sold By	Billing_Address	PAN_No	GST_Registration_No	Order_Number	Order_Date	Invoice_Number	Invoice_Details	Sl. No	Description	Product Code	HSN	Unit Price	Quantity	Net Amount	Tax Rate	Tax Type
0	Cloudtail India Private LimitedCloudtail India...	R.Shyamala108,sree Ayyappa Nagar, First main r...	AAQCS4259Q	06AAQCS4259Q1ZE	403-1565911-5936322	05.09.2021	DEL5-12841370	HR-DEL5-1004-2122	1	Cello Classic Plastic Pedal Dustbin, 6 Liters,...	B00XYEIYXE	39241090	448.31	1	448.31	18	IGST

Combine all the excel file into the single excel file

In [4]:

```
merged_df = pd.DataFrame()
dfs = []
for i in range(1,3):
    excel_path = 'output'+str(i)+'.xlsx'
    df = pd.read_excel(excel_path)
    dfs.append(df)
```

In [5]:

```
merged_df = pd.concat(dfs, ignore_index=True)
```

In [6]:

```
merged_df.to_excel('Final_Excel.xlsx', index=False)
```

In [7]:

```
merged_df
```

Out[7]:

	Sold By	Billing_Address	PAN_No	GST_Registration_No	Order_Number	Order_Date	Invoice_Number	Invoice_Details	Sl. No	Description	Product Code	HSN	Unit Price	Quantity	Net Amount	Tax Rate	T Ty
0	Darshita Aashiyana Private LimitedDarshita Aas...	karthikC1001 ace city, sector1GREATER NOIDA, U...	AAFCD6883Q	06AAFCD6883Q1ZU	407-8153595-7245952	10.08.2019	DEL2-68786	HR-DEL2-179184911-1920	1	OnePlus 7 (Mirror Blue, 6GB RAM, 128GB Storage)	OP7-NBLUE-6-128GB	8517	29,463.39	1	29,463.39	12	IG
1	Cloudtail India Private LimitedCloudtail India...	R.Shyamala108,sree Ayyappa Nagar, First main r...	AAQCS4259Q	06AAQCS4259Q1ZE	403-1565911-5936322	05.09.2021	DEL5-12841370	HR-DEL5-1004-2122	1	Cello Classic Plastic Pedal Dustbin, 6 Liters,...	B00XYEIYXE	39241090	448.31	1	448.31	18	IG

In [ ]:

In [ ]: