```
import pandas as pd
In [1]:
          import numpy as np
In [2]:
          df=pd.read csv("OnlineRetail.csv", encoding='latin-1')
          df
In [3]:
                                                              Quantity InvoiceDate UnitPrice CustomerID
Out[3]:
                   InvoiceNo
                             StockCode
                                                  Description
                                                                                                             Country
                                              WHITE HANGING
                                                                           12/1/2010
                                                                                                               United
               0
                                                HEART T-LIGHT
                                                                      6
                                                                                          2.55
                                                                                                    17850.0
                      536365
                                 85123A
                                                                                8:26
                                                                                                             Kingdom
                                                     HOLDER
                                                 WHITE METAL
                                                                           12/1/2010
                                                                                                               United
               1
                     536365
                                  71053
                                                                      6
                                                                                          3.39
                                                                                                    17850.0
                                                    LANTERN
                                                                                8:26
                                                                                                             Kingdom
                                                CREAM CUPID
                                                                           12/1/2010
                                                                                                               United
                                                 HEARTS COAT
                                                                                                    17850.0
               2
                      536365
                                 84406B
                                                                      8
                                                                                          2.75
                                                                                                             Kingdom
                                                                                8:26
                                                     HANGER
                                          KNITTED UNION FLAG
                                                                           12/1/2010
                                                                                                               United
                                 84029G
               3
                      536365
                                                                      6
                                                                                          3.39
                                                                                                    17850.0
                                            HOT WATER BOTTLE
                                                                                                             Kingdom
                                                                                8:26
                                          RED WOOLLY HOTTIE
                                                                           12/1/2010
                                                                                                               United
               4
                     536365
                                 84029E
                                                                      6
                                                                                          3.39
                                                                                                    17850.0
                                                WHITE HEART.
                                                                                8:26
                                                                                                             Kingdom
                                                   PACK OF 20
                                                                           12/9/2011
          541904
                      581587
                                  22613
                                                                     12
                                                                                          0.85
                                                                                                    12680.0
                                                                                                               France
                                           SPACEBOY NAPKINS
                                                                               12:50
                                                                           12/9/2011
                                           CHILDREN'S APRON
          541905
                      581587
                                  22899
                                                                      6
                                                                                          2.10
                                                                                                    12680.0
                                                                                                               France
                                                   DOLLY GIRL
                                                                               12:50
                                           CHILDRENS CUTLERY
                                                                           12/9/2011
          541906
                      581587
                                  23254
                                                                      4
                                                                                          4.15
                                                                                                    12680.0
                                                                                                               France
                                                   DOLLY GIRL
                                                                               12:50
                                           CHILDRENS CUTLERY
                                                                           12/9/2011
          541907
                     581587
                                  23255
                                                                      4
                                                                                          4.15
                                                                                                    12680.0
                                                                                                               France
                                               CIRCUS PARADE
                                                                               12:50
                                            BAKING SET 9 PIECE
                                                                           12/9/2011
                                                                      3
          541908
                      581587
                                  22138
                                                                                          4.95
                                                                                                    12680.0
                                                                                                               France
                                                  RETROSPOT
                                                                               12:50
         541909 rows × 8 columns
In [4]: print(df.isnull().sum())
         InvoiceNo
                                   0
         StockCode
                                   0
         Description
                               1454
         Quantity
                                   0
         InvoiceDate
                                   0
                                   0
         UnitPrice
         CustomerID
                            135080
         Country
                                   0
         dtype: int64
In [5]:
          df.shape
          (541909, 8)
Out[5]:
In [6]:
          df.info()
```

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 541909 entries, 0 to 541908 Data columns (total 8 columns): # Column Non-Null Count Dtype --------0 InvoiceNo 541909 non-null object 1 StockCode 541909 non-null object 2 Description 540455 non-null object 3 Quantity 541909 non-null int64 4 InvoiceDate 541909 non-null object 5 UnitPrice 541909 non-null float64 6 CustomerID 406829 non-null float64 Country 541909 non-null object 7 dtypes: float64(2), int64(1), object(5)

memory usage: 33.1+ MB

df=df.dropna() In [7]:

df In [8]:

Out[8]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
•••							<b></b>	
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0	France

406829 rows × 8 columns

```
In [9]: df1=df.duplicated().sum()
```

df1 In [10]:

5225

Out[10]:

In [11]: df=df.drop\_duplicates()

InvoiceNo StockCode

In [15]: num\_orders = df['InvoiceNo'].nunique()

WHITE HANGING

Out[12]:

	0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	United Kingdom		
	1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	United Kingdom		
	2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	United Kingdom		
	3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	United Kingdom		
	4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	United Kingdom		
	541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0	France		
	541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0	France		
	541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0	France		
	541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0	France		
	541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0	France		
[13]:	<pre>df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate']) df['TotalPrice'] = df['Quantity'] * df['UnitPrice'] total_revenue = df['TotalPrice'].sum() print("Total sales revenue:", total revenue)</pre>										
	Total sa	les rever	nue: 8278	519.4240000015							
	C:\Users\DELL\AppData\Local\Temp\ipykernel_6072\174943563.py:1: SettingWithCopyWarning A value is trying to be set on a copy of a slice from a DataFrame.  Try using .loc[row_indexer,col_indexer] = value instead										
				et on a copy of a	slice f	rom a DataFi		igwreiicop	yWarning		
	See the guide/ir df['Ir C:\Users A value	caveats indexing.ht avoiceDate by DELL\App is trying	ow_indexed.n the documl#reture'] = pd. DData\Loog to be s	et on a copy of a	slice fralue ins: //panda -a-copy voiceDat 6072\174 slice fr	com a DataFi stead as.pydata.or te']) 1943563.py:2 com a DataFi	came. cg/pandas 2: Settir	s-docs/st	able/use:		
	Try using See the guide/in df['In C:\Users A value Try using See the guide/in the second seco	caveats indexing.htm.  convoiceDate  convoic	ow_indexed.n the documl#reture!] = pd. Data\Loog to be sow_indexed.n the documl#reture	et on a copy of a r, col_indexer] = vocumentation: https: cumentation: https: cumentat	slice firalue ins ://panda -a-copy voiceDat 6072\174 slice firalue ins ://panda -a-copy	com a DataFratead  as.pydata.or  ce'])  1943563.py:2  com a DataFratead  as.pydata.or	came. cg/pandas 2: Settir came.	s-docs/st	able/use: yWarning		
[14]:	Try using See the guide/in df['Ir C:\Users A value Try using See the guide/in df['To unique_o	caveats indexing.htm. s\DELL\App is trying ig .loc[ro caveats indexing.htm. talPrice' customers	ow_indexed.n the documl#reture!] = pd. Data\Loog to be sow_indexed.n the documl#reture] = df['Cu	et on a copy of a er, col_indexer] = volumentation: https: cumentation: https: cumentation: https: col_datetime(df['Include on a copy of a er, col_indexer] = volumentation: https: cumentation: https: cumentation-a-view-versus	slice fralue ins ://panda -a-copy voiceDat 6072\174 slice fralue ins ://panda -a-copy nitPrice ()	com a DataFratead as.pydata.or te']) 4943563.py:2 com a DataFratead as.pydata.or	came. cg/pandas 2: Settir came.	s-docs/st	able/use: yWarning		

Description Quantity InvoiceDate UnitPrice CustomerID Country

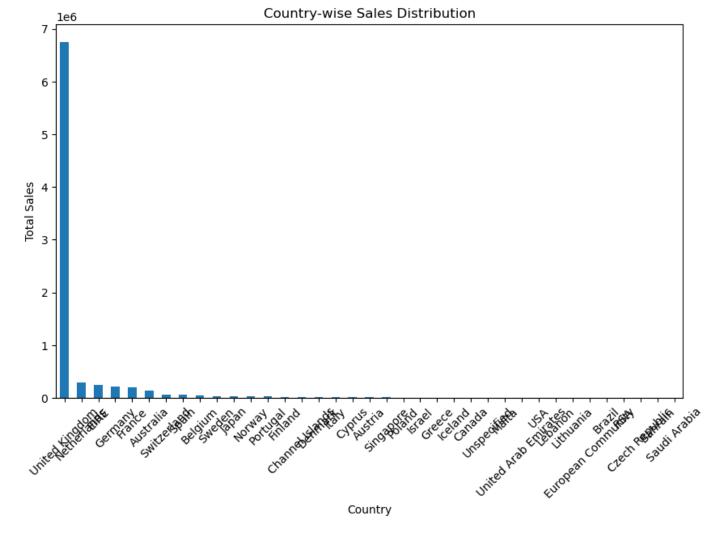
```
print("Number of orders:", num orders)
        Number of orders: 22190
In [16]: avg_order_value = total revenue / num orders
        print("Average order value:", avg order value)
        Average order value: 373.0743318611988
In [17]: popular products = df['Description'].value counts().head(5)
        print("Most popular products:\n", popular products)
        Most popular products:
         WHITE HANGING HEART T-LIGHT HOLDER
                                              2058
                                              1894
        REGENCY CAKESTAND 3 TIER
        JUMBO BAG RED RETROSPOT
                                              1659
        PARTY BUNTING
                                              1409
        ASSORTED COLOUR BIRD ORNAMENT
                                              1405
        Name: Description, dtype: int64
In [18]: customer with most purchases = df['CustomerID'].value counts().idxmax()
        print("Customer with the highest number of purchases:", customer with most purchases)
        Customer with the highest number of purchases: 17841.0
In [19]: | monthly sales = df.groupby(df['InvoiceDate'].dt.to period('M')).sum()['TotalPrice']
        print("Monthly sales trends:\n", monthly sales)
        Monthly sales trends:
         InvoiceDate
        2010-12 552372.860
        2011-01
                  473731.900
        2011-02
                  435534.070
        2011-03 578576.210
2011-04 425222.671
2011-05 647011.670
        2011-06
                   606862.520
        2011-07
                   573112.321
        2011-08
                  615078.090
                  929356.232
        2011-09
        2011-10 973306.380
        2011-11 1126815.070
2011-12 341539.430
        Freq: M, Name: TotalPrice, dtype: float64
        C:\Users\DELL\AppData\Local\Temp\ipykernel 6072\3226995229.py:1: FutureWarning: The defa
        ult value of numeric only in DataFrameGroupBy.sum is deprecated. In a future version, nu
        meric only will default to False. Either specify numeric only or select only columns whi
        ch should be valid for the function.
         monthly sales = df.groupby(df['InvoiceDate'].dt.to period('M')).sum()['TotalPrice']
In [20]: sales distribution = df.groupby('Country').sum()['TotalPrice'].sort values(ascending=Fal
        print("Country-wise sales distribution:\n", sales distribution)
        Country-wise sales distribution:
         Country
        United Kingdom
                               6747156.154
        Netherlands
                                 284661.540
        EIRE
                                 250001.780
        Germany
                                221509.470
                                196626.050
        France
                                137009.770
        Australia
        Switzerland
                                 55739.400
                                 54756.030
        Spain
        Belgium
                                  40910.960
        Sweden
                                  36585.410
        Japan
                                 35340.620
                                  35163.460
        Norway
```

```
Finland
                                  22326.740
        Channel Islands
                                 20076.390
                                 18768.140
        Denmark
        Italy
                                  16890.510
                                 12858.760
        Cyprus
        Austria
                                 10154.320
        Singapore
                                  9120.390
                                  7213.140
        Poland
        Israel
                                  6988.400
        Greece
                                  4710.520
        Iceland
                                   4310.000
        Canada
                                  3666.380
        Unspecified
                                  2660.770
                                  2505.470
        Malta
        United Arab Emirates
                                1902.280
        USA
                                  1730.920
        Lebanon
                                  1693.880
        Lithuania
                                   1661.060
                                  1291.750
        European Community
        Brazil
                                  1143.600
        RSA
                                  1002.310
        Czech Republic
                                    707.720
        Bahrain
                                    548.400
        Saudi Arabia
                                   131.170
        Name: TotalPrice, dtype: float64
        C:\Users\DELL\AppData\Local\Temp\ipykernel 6072\1016603934.py:1: FutureWarning: The defa
        ult value of numeric only in DataFrameGroupBy.sum is deprecated. In a future version, nu
        meric only will default to False. Either specify numeric only or select only columns whi
        ch should be valid for the function.
          sales distribution = df.groupby('Country').sum()['TotalPrice'].sort values(ascending=F
        alse)
In [21]: df['PurchaseDate'] = df['InvoiceDate'].dt.date
        average purchase frequency = df.groupby('CustomerID')['PurchaseDate'].nunique().mean()
        print("Average purchase frequency per customer:", average purchase frequency)
        C:\Users\DELL\AppData\Local\Temp\ipykernel 6072\21020352.py:1: SettingWithCopyWarning:
        A value is trying to be set on a copy of a slice from a DataFrame.
        Try using .loc[row indexer,col indexer] = value instead
        See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user
        guide/indexing.html#returning-a-view-versus-a-copy
         df['PurchaseDate'] = df['InvoiceDate'].dt.date
        Average purchase frequency per customer: 4.413540713632205
In [22]: import matplotlib.pyplot as plt
        plt.figure(figsize=(10, 6))
        sales distribution.plot(kind='bar')
        plt.title('Country-wise Sales Distribution')
        plt.xlabel('Country')
        plt.ylabel('Total Sales')
        plt.xticks(rotation=45)
```

28995.760

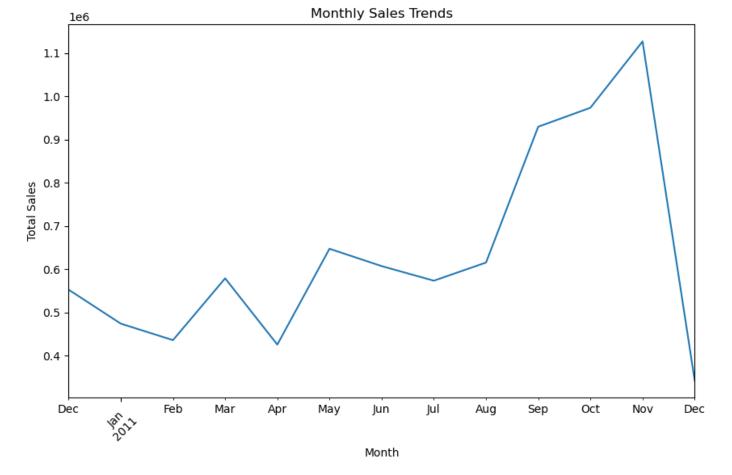
Portugal

plt.show()



```
In [23]: import matplotlib.pyplot as plt

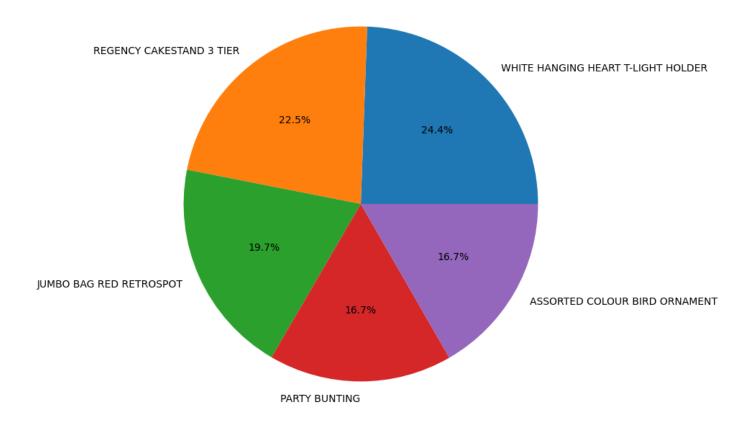
plt.figure(figsize=(10, 6))
    monthly_sales.plot(kind='line')
    plt.title('Monthly Sales Trends')
    plt.xlabel('Month')
    plt.ylabel('Total Sales')
    plt.xticks(rotation=45)
    plt.show()
```



```
In [24]: import matplotlib.pyplot as plt

plt.figure(figsize=(8, 8))
    popular_products.plot(kind='pie', autopct='%1.1f%%')
    plt.title('Most Popular Products')
    plt.ylabel('')
    plt.show()
```

## Most Popular Products



```
In [25]: import matplotlib.pyplot as plt

purchase_frequency = df.groupby('CustomerID')['PurchaseDate'].nunique()

plt.figure(figsize=(10, 6))
 plt.hist(purchase_frequency, bins=20)
 plt.title('Purchase Frequency Distribution')
 plt.xlabel('Number of Purchases')
 plt.ylabel('Frequency')
 plt.show()
```

