analysis-report-maker

[2]:
```
!pip install fpdf
```

```
Collecting fpdf
  Downloading fpdf-1.7.2.tar.gz (39 kB)
  Preparing metadata (setup.py) … done
Building wheels for collected packages: fpdf
  Building wheel for fpdf (setup.py) … done
  Created wheel for fpdf: filename=fpdf-1.7.2-py2.py3-none-any.whl size=40704
sha256=cbc1c3cbcd5cce70c1feae79a9f140dd71d3850e4e004f22e10c7b0cb780ca8c
  Stored in directory: /root/.cache/pip/wheels/f9/95/ba/f418094659025eb9611f17cb
caf2334236bf39a0c3453ea455
Successfully built fpdf
Installing collected packages: fpdf
Successfully installed fpdf-1.7.2
```

[3]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from fpdf import FPDF
import os
```

[4]:
```
df = pd.read_csv('/content/employee.csv')
```

[6]:
```
def list_missing_values(df):
    missing_values = df.isnull().sum()
    return missing_values[missing_values > 0]

list_missing_values(df)
```

[6]:
```
Series([], dtype: int64)
```

[7]:
```
def categorize_columns(df):
    numeric_cols = df.select_dtypes(include=['number']).columns.tolist()
    categorical_cols = df.select_dtypes(exclude=['number']).columns.tolist()
    return numeric_cols, categorical_cols

categorize_columns(df)
```

[7]:
```
(['EMPLOYEE_ID', 'SALARY', 'DEPARTMENT_ID'],
 ['FIRST_NAME',
  'LAST_NAME',
  'EMAIL',
  'PHONE_NUMBER',
  'HIRE_DATE',
  'JOB_ID',
  'COMMISSION_PCT',
  'MANAGER_ID'])
```

[8]:
```
def list_and_remove_duplicates(df):
    duplicates = df[df.duplicated()]
    df_no_duplicates = df.drop_duplicates()
    return duplicates, df_no_duplicates

list_and_remove_duplicates(df)
```

[8]:
```
(Empty DataFrame
 Columns: [EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, PHONE_NUMBER, HIRE_DATE,
JOB_ID, SALARY, COMMISSION_PCT, MANAGER_ID, DEPARTMENT_ID]
 Index: [],
    EMPLOYEE_ID   FIRST_NAME    LAST_NAME     EMAIL  PHONE_NUMBER  HIRE_DATE  \
 0          198      Donald     OConnell  DOCONNEL  650.507.9833  21-JUN-07
 1          199     Douglas        Grant   DGRANT  650.507.9844  13-JAN-08
 2          200    Jennifer       Whalen   JWHALEN  515.123.4444  17-SEP-03
 3          201     Michael    Hartstein  MHARTSTE  515.123.5555  17-FEB-04
 4          202         Pat          Fay      PFAY  603.123.6666  17-AUG-05
 5          203       Susan       Mavris   SMAVRIS  515.123.7777  07-JUN-02
 6          204     Hermann         Baer     HBAER  515.123.8888  07-JUN-02
 7          205     Shelley      Higgins  SHIGGINS  515.123.8080  07-JUN-02
 8          206     William        Gietz    WGIETZ  515.123.8181  07-JUN-02
 9          100      Steven         King     SKING  515.123.4567  17-JUN-03
 10         101        Neena      Kochhar  NKOCHHAR  515.123.4568  21-SEP-05
 11         102          Lex      De Haan   LDEHAAN  515.123.4569  13-JAN-01
 12         103    Alexander       Hunold   AHUNOLD  590.423.4567  03-JAN-06
 13         104        Bruce        Ernst    BERNST  590.423.4568  21-MAY-07
 14         105        David       Austin   DAUSTIN  590.423.4569  25-JUN-05
 15         106        Valli    Pataballa  VPATABAL  590.423.4560  05-FEB-06
 16         107        Diana      Lorentz  DLORENTZ  590.423.5567  07-FEB-07
 17         108        Nancy    Greenberg  NGREENBE  515.124.4569  17-AUG-02
 18         109       Daniel       Faviet   DFAVIET  515.124.4169  16-AUG-02
 19         110         John         Chen     JCHEN  515.124.4269  28-SEP-05
 20         111       Ismael      Sciarra  ISCIARRA  515.124.4369  30-SEP-05
 21         112  Jose Manuel        Urman   JMURMAN  515.124.4469  07-MAR-06
 22         113         Luis         Popp     LPOPP  515.124.4567  07-DEC-07
 23         114          Den     Raphaely  DRAPHEAL  515.127.4561  07-DEC-02
 24         115    Alexander         Khoo     AKHOO  515.127.4562  18-MAY-03
```

25	116	Shelli	Baida	SBAIDA	515.127.4563	24-DEC-05	
26	117	Sigal	Tobias	STOBIAS	515.127.4564	24-JUL-05	
27	118	Guy	Himuro	GHIMURO	515.127.4565	15-NOV-06	
28	119	Karen	Colmenares	KCOLMENA	515.127.4566	10-AUG-07	
29	120	Matthew	Weiss	MWEISS	650.123.1234	18-JUL-04	
30	121	Adam	Fripp	AFRIPP	650.123.2234	10-APR-05	
31	122	Payam	Kaufling	PKAUFLIN	650.123.3234	01-MAY-03	
32	123	Shanta	Vollman	SVOLLMAN	650.123.4234	10-OCT-05	
33	124	Kevin	Mourgos	KMOURGOS	650.123.5234	16-NOV-07	
34	125	Julia	Nayer	JNAYER	650.124.1214	16-JUL-05	
35	126	Irene	Mikkilineni	IMIKKILI	650.124.1224	28-SEP-06	
36	127	James	Landry	JLANDRY	650.124.1334	14-JAN-07	
37	128	Steven	Markle	SMARKLE	650.124.1434	08-MAR-08	
38	129	Laura	Bissot	LBISSOT	650.124.5234	20-AUG-05	
39	130	Mozhe	Atkinson	MATKINSO	650.124.6234	30-OCT-05	
40	131	James	Marlow	JAMRLOW	650.124.7234	16-FEB-05	
41	132	TJ	Olson	TJOLSON	650.124.8234	10-APR-07	
42	133	Jason	Mallin	JMALLIN	650.127.1934	14-JUN-04	
43	134	Michael	Rogers	MROGERS	650.127.1834	26-AUG-06	
44	135	Ki	Gee	KGEE	650.127.1734	12-DEC-07	
45	136	Hazel	Philtanker	HPHILTAN	650.127.1634	06-FEB-08	
46	137	Renske	Ladwig	RLADWIG	650.121.1234	14-JUL-03	
47	138	Stephen	Stiles	SSTILES	650.121.2034	26-OCT-05	
48	139	John	Seo	JSEO	650.121.2019	12-FEB-06	
49	140	Joshua	Patel	JPATEL	650.121.1834	06-APR-06	

	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
0	SH_CLERK	2600	-	124	50
1	SH_CLERK	2600	-	124	50
2	AD_ASST	4400	-	101	10
3	MK_MAN	13000	-	100	20
4	MK_REP	6000	-	201	20
5	HR_REP	6500	-	101	40
6	PR_REP	10000	-	101	70
7	AC_MGR	12008	-	101	110
8	AC_ACCOUNT	8300	-	205	110
9	AD_PRES	24000	-	-	90
10	AD_VP	17000	-	100	90
11	AD_VP	17000	-	100	90
12	IT_PROG	9000	-	102	60
13	IT_PROG	6000	-	103	60
14	IT_PROG	4800	-	103	60
15	IT_PROG	4800	-	103	60
16	IT_PROG	4200	-	103	60
17	FI_MGR	12008	-	101	100
18	FI_ACCOUNT	9000	-	108	100
19	FI_ACCOUNT	8200	-	108	100

```
20  FI_ACCOUNT    7700             -         108            100
21  FI_ACCOUNT    7800             -         108            100
22  FI_ACCOUNT    6900             -         108            100
23      PU_MAN   11000             -         100             30
24    PU_CLERK    3100             -         114             30
25    PU_CLERK    2900             -         114             30
26    PU_CLERK    2800             -         114             30
27    PU_CLERK    2600             -         114             30
28    PU_CLERK    2500             -         114             30
29      ST_MAN    8000             -         100             50
30      ST_MAN    8200             -         100             50
31      ST_MAN    7900             -         100             50
32      ST_MAN    6500             -         100             50
33      ST_MAN    5800             -         100             50
34    ST_CLERK    3200             -         120             50
35    ST_CLERK    2700             -         120             50
36    ST_CLERK    2400             -         120             50
37    ST_CLERK    2200             -         120             50
38    ST_CLERK    3300             -         121             50
39    ST_CLERK    2800             -         121             50
40    ST_CLERK    2500             -         121             50
41    ST_CLERK    2100             -         121             50
42    ST_CLERK    3300             -         122             50
43    ST_CLERK    2900             -         122             50
44    ST_CLERK    2400             -         122             50
45    ST_CLERK    2200             -         122             50
46    ST_CLERK    3600             -         123             50
47    ST_CLERK    3200             -         123             50
48    ST_CLERK    2700             -         123             50
49    ST_CLERK    2500             -         123             50  )
```

[16]:
```python
def list_and_remove_constants(df):
    constant_cols = [col for col in df.columns if df[col].nunique() == 1]
    df_no_constants = df.drop(columns=constant_cols)
    return constant_cols, df_no_constants

list_and_remove_constants(df)
```

[16]:
```
(['COMMISSION_PCT'],
     EMPLOYEE_ID   FIRST_NAME    LAST_NAME     EMAIL  PHONE_NUMBER  HIRE_DATE  \
0            198       Donald     OConnell  DOCONNEL  650.507.9833  21-JUN-07
1            199      Douglas        Grant    DGRANT  650.507.9844  13-JAN-08
2            200     Jennifer       Whalen   JWHALEN  515.123.4444  17-SEP-03
3            201      Michael    Hartstein  MHARTSTE  515.123.5555  17-FEB-04
4            202          Pat          Fay      PFAY  603.123.6666  17-AUG-05
5            203        Susan       Mavris   SMAVRIS  515.123.7777  07-JUN-02
6            204      Hermann         Baer     HBAER  515.123.8888  07-JUN-02
```

7	205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-02
8	206	William	Gietz	WGIETZ	515.123.8181	07-JUN-02
9	100	Steven	King	SKING	515.123.4567	17-JUN-03
10	101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05
11	102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01
12	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06
13	104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07
14	105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05
15	106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06
16	107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07
17	108	Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02
18	109	Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02
19	110	John	Chen	JCHEN	515.124.4269	28-SEP-05
20	111	Ismael	Sciarra	ISCIARRA	515.124.4369	30-SEP-05
21	112	Jose Manuel	Urman	JMURMAN	515.124.4469	07-MAR-06
22	113	Luis	Popp	LPOPP	515.124.4567	07-DEC-07
23	114	Den	Raphaely	DRAPHEAL	515.127.4561	07-DEC-02
24	115	Alexander	Khoo	AKHOO	515.127.4562	18-MAY-03
25	116	Shelli	Baida	SBAIDA	515.127.4563	24-DEC-05
26	117	Sigal	Tobias	STOBIAS	515.127.4564	24-JUL-05
27	118	Guy	Himuro	GHIMURO	515.127.4565	15-NOV-06
28	119	Karen	Colmenares	KCOLMENA	515.127.4566	10-AUG-07
29	120	Matthew	Weiss	MWEISS	650.123.1234	18-JUL-04
30	121	Adam	Fripp	AFRIPP	650.123.2234	10-APR-05
31	122	Payam	Kaufling	PKAUFLIN	650.123.3234	01-MAY-03
32	123	Shanta	Vollman	SVOLLMAN	650.123.4234	10-OCT-05
33	124	Kevin	Mourgos	KMOURGOS	650.123.5234	16-NOV-07
34	125	Julia	Nayer	JNAYER	650.124.1214	16-JUL-05
35	126	Irene	Mikkilineni	IMIKKILI	650.124.1224	28-SEP-06
36	127	James	Landry	JLANDRY	650.124.1334	14-JAN-07
37	128	Steven	Markle	SMARKLE	650.124.1434	08-MAR-08
38	129	Laura	Bissot	LBISSOT	650.124.5234	20-AUG-05
39	130	Mozhe	Atkinson	MATKINSO	650.124.6234	30-OCT-05
40	131	James	Marlow	JAMRLOW	650.124.7234	16-FEB-05
41	132	TJ	Olson	TJOLSON	650.124.8234	10-APR-07
42	133	Jason	Mallin	JMALLIN	650.127.1934	14-JUN-04
43	134	Michael	Rogers	MROGERS	650.127.1834	26-AUG-06
44	135	Ki	Gee	KGEE	650.127.1734	12-DEC-07
45	136	Hazel	Philtanker	HPHILTAN	650.127.1634	06-FEB-08
46	137	Renske	Ladwig	RLADWIG	650.121.1234	14-JUL-03
47	138	Stephen	Stiles	SSTILES	650.121.2034	26-OCT-05
48	139	John	Seo	JSEO	650.121.2019	12-FEB-06
49	140	Joshua	Patel	JPATEL	650.121.1834	06-APR-06

	JOB_ID	SALARY	MANAGER_ID	DEPARTMENT_ID
0	SH_CLERK	2600	124	50
1	SH_CLERK	2600	124	50

2	AD_ASST	4400	101	10
3	MK_MAN	13000	100	20
4	MK_REP	6000	201	20
5	HR_REP	6500	101	40
6	PR_REP	10000	101	70
7	AC_MGR	12008	101	110
8	AC_ACCOUNT	8300	205	110
9	AD_PRES	24000	-	90
10	AD_VP	17000	100	90
11	AD_VP	17000	100	90
12	IT_PROG	9000	102	60
13	IT_PROG	6000	103	60
14	IT_PROG	4800	103	60
15	IT_PROG	4800	103	60
16	IT_PROG	4200	103	60
17	FI_MGR	12008	101	100
18	FI_ACCOUNT	9000	108	100
19	FI_ACCOUNT	8200	108	100
20	FI_ACCOUNT	7700	108	100
21	FI_ACCOUNT	7800	108	100
22	FI_ACCOUNT	6900	108	100
23	PU_MAN	11000	100	30
24	PU_CLERK	3100	114	30
25	PU_CLERK	2900	114	30
26	PU_CLERK	2800	114	30
27	PU_CLERK	2600	114	30
28	PU_CLERK	2500	114	30
29	ST_MAN	8000	100	50
30	ST_MAN	8200	100	50
31	ST_MAN	7900	100	50
32	ST_MAN	6500	100	50
33	ST_MAN	5800	100	50
34	ST_CLERK	3200	120	50
35	ST_CLERK	2700	120	50
36	ST_CLERK	2400	120	50
37	ST_CLERK	2200	120	50
38	ST_CLERK	3300	121	50
39	ST_CLERK	2800	121	50
40	ST_CLERK	2500	121	50
41	ST_CLERK	2100	121	50
42	ST_CLERK	3300	122	50
43	ST_CLERK	2900	122	50
44	ST_CLERK	2400	122	50
45	ST_CLERK	2200	122	50
46	ST_CLERK	3600	123	50
47	ST_CLERK	3200	123	50
48	ST_CLERK	2700	123	50

```
 49    ST_CLERK    2500        123             50  )
```

[11]:
```
def create_box_plots(df, output_dir):
    numeric_cols = df.select_dtypes(include=['number']).columns
    for col in numeric_cols:
        plt.figure(figsize=(10, 6))
        sns.boxplot(x=df[col])
        plt.title(f'Box plot of {col}')
        plt.savefig(os.path.join(output_dir, f'boxplot_{col}.png'))
        plt.close()

create_box_plots(df, '/content/')
```

[12]:
```
def create_distribution_charts(df, output_dir):
    sample_cols = df.columns[:6]
    for col in sample_cols:
        plt.figure(figsize=(10, 6))
        sns.histplot(df[col], kde=True)
        plt.title(f'Distribution of {col}')
        plt.savefig(os.path.join(output_dir, f'distribution_{col}.png'))
        plt.close()

create_distribution_charts(df, '/content/')
```

[17]:
```
def generate_report(df, output_dir='/content/report'):
    if not os.path.exists(output_dir):
        os.makedirs(output_dir)

    missing_values = list_missing_values(df)
    numeric_cols, categorical_cols = categorize_columns(df)
    duplicates, df_no_duplicates = list_and_remove_duplicates(df)
    constant_cols, df_no_constants = list_and_remove_constants(df)

    create_box_plots(df, output_dir)
    create_distribution_charts(df, output_dir)

    pdf = FPDF()
    pdf.add_page()
    pdf.set_font("Arial", size=12)

    pdf.cell(200, 10, txt="Data Report", ln=True, align='C')

    pdf.cell(200, 10, txt="Missing Values:", ln=True)
    for col, val in missing_values.items():
        pdf.cell(200, 10, txt=f"{col}: {val}", ln=True)

    pdf.cell(200, 10, txt="Numeric Columns:", ln=True)
```

```
    for col in numeric_cols:
        pdf.cell(200, 10, txt=col, ln=True)

    pdf.cell(200, 10, txt="Categorical Columns:", ln=True)
    for col in categorical_cols:
        pdf.cell(200, 10, txt=col, ln=True)

    pdf.cell(200, 10, txt="Duplicates Before Removal:", ln=True)
    pdf.cell(200, 10, txt=str(duplicates), ln=True)

    pdf.cell(200, 10, txt="Duplicates After Removal:", ln=True)
    pdf.cell(200, 10, txt=str(df_no_duplicates), ln=True)

    pdf.cell(200, 10, txt="Constant Columns:", ln=True)
    for col in constant_cols:
        pdf.cell(200, 10, txt=col, ln=True)

    pdf.output(os.path.join(output_dir, "data_report.pdf"))
```

[18]:
```
# Generate the report
generate_report(df)
```

[ ]:
```

```