Exploring the customers data and writing a report.

In [1]:

```
import pandas as pd
import re
```

In [2]:

```
# Loading the data

data = pd.read_csv("../data/raw/customers.csv")

df = data.copy()
df.head()
```

Out[2]:

	Customer Id	First Name	Last Name	Company	City	Country	Phone 1	Phone 2	Email	Subscription Date	Website
0	dE014d010c7ab0c	Andrew	Goodman	Stewart-Flynn	Rowlandberg	Macao	846-790-4623x4715	(422)787-2331x71127	marieyates@gomez-spencer.info	7/26/2021	http://www.shea.biz/
1	2B54172c8b65eC3	Alvin	Lane	Terry, Proctor and Lawrence	Bethside	Papua New Guinea	124-597-8652x05682	321.441.0588x6218	alexandra86@mccoy.com	6/24/2021	http://www.pena-cole.com/
2	d794Dd48988d2ac	Jenna	Harding	Bailey Group	Moniquemouth	China	(335)987-3085x3780	001-680-204-8312	justincurtis@pierce.org	4/5/2020	http://www.booth-reese.biz/
3	3b3Aa4aCc68f3Be	Fernando	Ford	Moss-Maxwell	Leeborough	Macao	(047)752-3122	048.779.5035x9122	adeleon@hubbard.org	11/29/2020	http://www.hebert.com/
4	D60df62ad2ae41E	Kara	Woods	Mccarthy-Kelley	Port Jacksonland	Nepal	+1-360-693-4419x19272	163-627-2565	jesus90@roberson.info	4/22/2022	http://merritt.com/

In [3]:

```
# Viewing data types and missing values

df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   Customer Id        999 non-null    object
 1   First Name         1000 non-null   object
 2   Last Name          1000 non-null   object
 3   Company            1000 non-null   object
 4   City               1000 non-null   object
 5   Country            999 non-null    object
 6   Phone 1            1000 non-null   object
 7   Phone 2            1000 non-null   object
 8   Email              1000 non-null   object
 9   Subscription Date  1000 non-null   object
 10  Website            999 non-null    object
dtypes: object(11)
memory usage: 86.1+ KB
```

In [4]:

```
# Viewing sum and pecentage of missing values

missing = {
    "Sum": df.isna().sum(),
    "Percentage": df.isna().mean()*100
}

null = pd.DataFrame(missing)

null
```

Out[4]:

	Sum	Percentage
Customer Id	1	0.1
First Name	0	0.0
Last Name	0	0.0
Company	0	0.0
City	0	0.0
Country	1	0.1
Phone 1	0	0.0
Phone 2	0	0.0
Email	0	0.0
Subscription Date	0	0.0
Website	1	0.1

In [5]:
```
# Viewing the three rows with null data
df[df.isnull().any(axis=1)]
```

Out[5]:

	Customer Id	First Name	Last Name	Company	City	Country	Phone 1	Phone 2	Email	Subscription Date	Website
328	7b7A3BaF1d132C2	JERMAINE	HODGES	FIELDS-FREDERICK	LAKE MARILYNHAVEN	CAYMAN ISLANDS	001-531-138-1723x53933	128-891-8417	barry83@beltran-tyler.biz	5/4/2022	NaN
822	NaN	Rachel	Watts	Holloway-Nolan	Aaronville	Gibraltar	001-884-328-3072	-5400	kelliroy@sawyer-barker.com	5/12/2021	https://www.jordan-wolf.info/
972	F9F39Bfe4f410fB	Jorge	Mcgrath	Patterson PLC	Estradahaven	NaN	350-818-9538	(063)970-6806x7912	hchambers@barrera.biz	11/5/2021	https://www.morales.com/

In [6]:
```
# Viewing duplicates

df.duplicated().sum()
```

Out[6]:
```
np.int64(4)
```

In [7]:
```
# Viewing the duplicated rows

df[df.duplicated()]
```

Out[7]:

	Customer Id	First Name	Last Name	Company	City	Country	Phone 1	Phone 2	Email	Subscription Date	Website
622	D270EFAd9D76A76	Rick	Barrera	Wells, Gallagher and Robles	Jermainetown	Samoa	1246308068	7544522601	cmercado@reed.com	6/22/2021	http://strong.com/
657	4E100fFA492E3fC	Sylvia	Lin	Rivas-Alexander	Lake Juanport	Nauru	(737)463-4946x28020	449-929-1096x654	stevensmaureen@watts-tapia.biz	5/9/2021	http://www.mcgee-hood.net/
677	08a3cA5AcB14199	James	Ward	Marsh and Sons	Michaelashire	Anguilla	(674)232-5443x309	718.475.4045x08506	sara28@singleton.net	2/25/2021	http://www.ali.net/
999	8F952d03DDC9EDa	Claire	Shaw	Ayala, Krause and Hendrix	Anthonyville	Bulgaria	(232)325-5438x4420	001-056-775-0843x4935	chaynes@rasmussen.com	10/19/2021	http://leach.com/

In [8]:
```
# Viewing the text data to check inconsistent cases
text_cols = ['First Name', 'Last Name', 'Company', 'City', 'Country', 'Email', 'Website']

df[text_cols].head(30)
```

Out[8]:

	First Name	Last Name	Company	City	Country	Email	Website
0	Andrew	Goodman	Stewart-Flynn	Rowlandberg	Macao	marieyates@gomez-spencer.info	http://www.shea.biz/
1	Alvin	Lane	Terry, Proctor and Lawrence	Bethside	Papua New Guinea	alexandra86@mccoy.com	http://www.pena-cole.com/
2	Jenna	Harding	Bailey Group	Moniquemouth	China	justincurtis@pierce.org	http://www.booth-reese.biz/
3	Fernando	Ford	Moss-Maxwell	Leeborough	Macao	adeleon@hubbard.org	http://www.hebert.com/
4	Kara	Woods	Mccarthy-Kelley	Port Jacksonland	Nepal	jesus90@roberson.info	http://merritt.com/
5	Marissa	Gamble	Cherry and Sons	Webertown	Sudan	katieallison@leonard.com	http://www.kaufman.org/
6	Julie	Cooley	Yu, Norman and Sharp	West Sandra	Japan	priscilla88@stephens.info	http://www.sexton-chang.com/
7	Lauren	Villa	French, Travis and Hensley	New Yolanda	Fiji	colehumphrey@austin-caldwell.com	https://www.kerr.com/
8	Emily	Bryant	Moon, Strickland and Combs	East Normanchester	Seychelles	buckyvonne@church-lutz.com	http://grimes.com/
9	Marie	Estrada	May Inc	Welchton	United Arab Emirates	christie44@mckenzie.biz	https://www.salinas.net/
10	NICHOLE	CANNON	RIOS AND SONS	WEST DEVON	BURUNDI	blandry@henson-harris.biz	http://www.humphrey.org/
11	BERNARD	RITTER	BRADFORD AND SONS	WEST FRANCISCO	PALAU	tammiepope@arroyo-baldwin.com	http://sellers.biz/
12	DARRYL	ARCHER	KERR-CHERRY	HOLTFURT	UGANDA	woodalejandro@skinner-sloan.biz	https://www.daniels.com/
13	RYAN	LEE	HOOPER, CROSS AND HOLT	BATESVILLE	LIECHTENSTEIN	lmassey@duke.com	http://nunez.com/
14	VICKI	NUNEZ	LEONARD, GALVAN AND BLACKBURN	BARBARABOROUGH	HAITI	zgrant@sweeney.com	https://reynolds.com/
15	SEAN	TOWNSEND	PRESTON-SOSA	VELASQUEZBERG	IRAN	lkline@maxwell.info	http://www.vargas.biz/
16	SOPHIA	MATHIS	RICHARD-VELASQUEZ	TODDHAVEN	SWITZERLAND	brockmason@faulkner-may.com	http://www.vaughn.com/
17	HELEN	POTTS	RANGEL, LIVINGSTON AND PATEL	DOUGLASLAND	SEYCHELLES	carrollmia@donovan-keith.com	http://www.kennedy-edwards.biz/
18	JOANN	FINLEY	HARVEY PLC	BARRETTSHIRE	MONTSERRAT	gabriela86@sampson.com	http://www.harrell.com/
19	THOMAS	WALSH	BEST-THOMAS	ROBLESPORT	KIRIBATI	timcoleman@frank-king.org	http://www.kane.com/
20	CRISTINA	LAM	WATTS-ALLISON	WEST JOCELYNFORT	KOREA	charlotte16@hood-zhang.org	http://whitehead.net/
21	VICKI	HEATH	CHERRY, SCHULTZ AND RUIZ	PORT CAMERONBURY	BANGLADESH	alan46@benjamin.com	https://www.bird.com/
22	GLENN	WANG	WARNER-HODGE	WEST RACHAEL	GABON	anna80@mata.com	http://brooks-kerr.com/
23	DARIUS	BENITEZ	GILES LLC	MEJIASHIRE	JERSEY	garrettdurham@olsen.com	https://washington.com/
24	XAVIER	CRUZ	WILEY LTD	MINDYBOROUGH	LATVIA	andersongrant@pugh.com	https://www.cohen.info/
25	DOUGLAS	GALLOWAY	DUFFY INC	EILEENBURY	MONGOLIA	caleb11@velazquez.com	http://www.mcneil.net/

	First Name	Last Name	Company	City	Country	Email	Website
26	PHYLLIS	BECKER	ONEAL AND SONS	EAST ANDRE	BOUVET ISLAND (BOUVETOYA)	darrylshort@bright-tucker.com	https://www.farrell.com/
27	EBONY	MURPHY	BARRY-MARTINEZ	ATKINSFURT	VANUATU	vpowers@moyer.com	http://www.dorsey.com/
28	TYLER	STEVENSON	BURNS AND SONS	NORTH JOANNASHIRE	SRI LANKA	mmayo@gilbert.com	http://www.fry.org/
29	CESAR	BERNARD	POTTER-HO	MCCORMICKVILLE	IRAQ	damon31@grant-morrison.com	https://bryan-walters.com/

In [9]:

```
# Viewing the phone numbers

df[['Phone 1', 'Phone 2']].head(20)
```

Out[9]:

	Phone 1	Phone 2
0	846-790-4623x4715	(422)787-2331x71127
1	124-597-8652x05682	321.441.0588x6218
2	(335)987-3085x3780	001-680-204-8312
3	(047)752-3122	048.779.5035x9122
4	+1-360-693-4419x19272	163-627-2565
5	001-645-334-5514x0786	(751)980-3163
6	+1-675-243-7422x9177	(703)337-5903
7	081.226.1797x647	186.540.9690x605
8	430-401-5228x35091	115-835-3840
9	001-648-790-9244	973-767-3611
10	647787401	139.476.1068
11	292.313.1902	(065)075-0554
12	(389)437-1716	092.364.7349x593
13	001-119-787-0125x4500	001-477-254-3645
14	(217)474-0312	(098)195-0840x79579
15	001-534-283-5153	5786415664
16	001-858-762-7896x916	274-147-4185x15182
17	(140)862-2659	-8339
18	(941)715-8720x950	155.433.4824x955
19	679.326.0724	001-305-038-6009

In [10]:

```
# Validating email
def is_valid_email_basic(email):
    if pd.isna(email):
        return False

    # Basic email regex pattern
    pattern = r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'
    return bool(re.match(pattern, str(email).strip()))

# Apply validation
is_email_valid = df['Email'].apply(is_valid_email_basic)
print(f"Valid emails: {is_email_valid.sum()} out of {len(df)}")
```

```
Valid emails: 1000 out of 1000
```

In [12]:
```
# Viewing the subscription date
df['Subscription Date'].head(30)
```

Out[12]:
```
0      7/26/2021
1      6/24/2021
2       4/5/2020
3     11/29/2020
4      4/22/2022
5     11/17/2021
6      3/26/2022
7      8/14/2020
8     12/30/2020
9       9/3/2020
10     4/26/2021
11     1/19/2022
12     4/18/2022
13      3/6/2021
14     1/30/2022
15     5/30/2020
16     1/23/2020
17     7/27/2021
18     4/11/2022
19     9/11/2020
20      1/4/2020
21     11/6/2020
22      1/1/2022
23     2/28/2022
24      2/7/2020
25    10/24/2021
26      7/2/2020
27     2/17/2020
28      3/2/2022
29      9/4/2021
Name: Subscription Date, dtype: object
```

In [13]:
```
# Viewing unique entries in each column to check categorical values
for col in df.columns:
    print(col, end=", ")
    print(len(df[col].unique()))
```

```
Customer Id, 996
First Name, 616
Last Name, 690
Company, 990
City, 985
Country, 341
Phone 1, 996
Phone 2, 996
Email, 996
Subscription Date, 609
Website, 970
```

Data Quality Report

Dataset Overview

- **Number of Rows:** 1,000
- **Number of Columns:** 11
- **Memory Usage:** ~86.1 KB
- **Data Types:** All columns are `object`

---

1. Missing Values

There are **3 missing values** across 3 columns:

Column	Missing Count	Recommended Action
Customer Id	1	Investigate and drop row if value cannot be recovered
Country	1	Infer value based on corresponding **City**
Website	1	Fill with `"Not Available"` or leave blank depending on business rules

---

2. Duplicated Records

- **Total Duplicates:** 4 rows

Action: Remove duplicate rows to maintain data integrity.

---

3. Inconsistent Text Formatting

- Columns affected: `First Name`, `Last Name`, `Company`, `City`, `Country`
- Issue: Inconsistent letter casing (mix of uppercase, lowercase, title case)

Action: Convert all text columns to **Title Case** for consistency.

---

4. Phone Number Formatting

- Columns affected: `Phone 1`, `Phone 2`
- Issue: Inconsistent formats, mixed separators, and extensions

Action:

- Standardize phone numbers into a uniform format:
  - Extract and separate **phone number** and **extension** (if available)
  - Recommended format: `+CountryCode-Number xExtension`

---

5. Subscription Date Format

- Issue: Dates are stored as strings with inconsistent formats

Action:

- Convert `Subscription Date` to datetime
- Use a consistent format: **DD-MM-YYYY**

Summary of Cleaning Steps

- ✅ Handle missing values (infer, impute, or drop)
- ✅ Drop duplicate rows
- ✅ Standardize text columns to title case
- ✅ Normalize phone number formats
- ✅ Convert subscription dates to DD-MM-YYYY format