In [2]:

```
import pandas as pd
import numpy as np
import faker
from faker import Faker
import warnings
warnings.filterwarnings("ignore")
```

In [3]:

```
fake = Faker()
```

In [4]:

```
print("name: ", fake.name())
print("address: ", fake.address())
print("name_male: ", fake.name_male())
print("phone_number: ", fake.phone_number())
print("street_address: ", fake.street_address())
print("email: ", fake.email())
print("user_name: ", fake.user_name())
print("ipv4: ", fake.ipv4())
print("sentence: ", fake.sentence())
```

```
name:  Richard Garrison
address:  22580 Philip Fords Suite 659
New Christopherport, DC 96853
name_male:  Antonio Whitehead
phone_number:  886-342-0715
street_address:  650 Susan Glen Apt. 059
email:  stephaniemartin@gmail.com
user_name:  derek92
ipv4:  221.129.255.210
sentence:  Adult thought billion attorney common sound.
```

In [5]:

```
print(fake.job())
print(fake.date_of_birth())
print(fake.company())
print("company_email: ", fake.company_email())
print("color_name: ", fake.color_name())
print("color: ", fake.color()) # Hexa color)
print("password: ", fake.password())
```

```
Media buyer
1976-05-14
Burgess-Edwards
company_email:  robertrowe@fisher.com
color_name:  Violet
color:  #c86cd8
password:  lf4Dh)nM(6
```

In [6]:

```
for _ in range(10):
    print(fake.date_between(start_date = '-1y', end_date = 'today'))
```

```
2022-10-12
2023-01-28
2022-10-02
2023-03-22
2023-03-16
2022-12-25
2022-07-09
2022-05-22
2023-03-08
2022-06-20
```

In [8]:

```
for _ in range(10):
    print(fake.name_female())
```

```
Lisa Carter
Jane Obrien
Alexandra Macias
Amy Douglas
Dana Hood
Crystal Moore
Sarah Martin
Teresa Jones
Laurie Rivera
Amanda Harmon
```

In [9]:

```
from faker.providers import internet
```

In [10]:

```
fake.add_provider(internet)
print(fake.ipv4_private())
```

```
10.52.245.29
```

In [12]:

```
fake = faker.Faker(['it_IT','en_US','en_IN'])
for _ in range(20):
    print(fake.name())
```

```
Umang Ghosh
Salvatore Murri
Brittany Landry
Gianpaolo Grassi
Debra Gonzalez
Kashvi Tank
Silvia Fantoni
Lacey Johnson
Faiyaz Sant
Kyle Marshall
Jayan Wali
John Olson
Saira Sachar
Hazel Bala
Taimur Lall
Hansh Korpal
Zeeshan Dave
Sarah Williams
Renzo Corradi
Kismat Sura
```

In [13]:

```
names = [fake.unique.first_name() for i in range(100)]
assert len(set(names)) == len(names)
```

In [14]:

```
print(names)
```

```
['Adira', 'Christopher', 'Eva', 'Tejas', 'Morena', 'Abigail', 'Antonina',
'Carol', 'Prerak', 'Stefani', 'Jon', 'David', 'Lori', 'Rati', 'Jeffery',
'Alphons', 'Jill', 'Bhavin', 'Tammy', 'Parinaaz', 'Enzio', 'Gabriele', 'Hi
ran', 'Mishti', 'Piergiorgio', 'Dario', 'Wendy', 'Angela', 'Mahika', 'Dian
e', 'Michelotto', 'Lawrence', 'Jennifer', 'Pranay', 'Bernardo', 'Amber',
'Vritika', 'Michelle', 'Beppe', 'Hridaan', 'Lakshay', 'Ivana', 'Dana', 'Aa
rav', 'Ivan', 'Jose', 'Kartik', 'Eshani', 'Tarini', 'Brandon', 'Virgilio',
'Melody', 'Eugenia', 'Adam', 'Sabatino', 'Pierina', 'Oscar', 'Shlok', 'Ett
ore', 'Antonio', 'Stella', 'Nirvaan', 'Matthew', 'Eddie', 'Anita', 'Kara',
'Alan', 'Dale', 'Giancarlo', 'Nakul', 'Ottone', 'Daria', 'Tiya', 'Samar',
'Brittany', 'Cesare', 'Anthony', 'Hrishita', 'Dino', 'Biagio', 'Ann', 'Man
jari', 'Berenice', 'Marcus', 'Damini', 'Katrina', 'Robert', 'Victoria', 'A
lberto', 'Bonnie', 'Valerio', 'Zeeshan', 'Lavanya', 'Joseph', 'Rosario',
'Gelsomina', 'Sonia', 'Steven', 'Cristina', 'Fausto']
```

In [15]:

```
customer_name = fake.bothify(text = '????####', letters = 'ABCDE')
```

In [16]:

```
customer_name
```

Out[16]:

```
'DEAD4990'
```

In [17]:

```
for i in range(20):
    print(fake.bothify(text = '????####', letters = 'ABCDE'), end = " ")
```

```
ECCC6033 BEAB2555 DDEA3281 DEAE3019 EABA0254 BCBD2886 BBCE6693 CADD7547 AB
DC5055 DDAC0509 EAAE0652 ECBD6465 BABB8230 EECE2213 AEAB3325 DEDD5945 DAEC
9879 DABC3367 ABEC7932 BEED4422
```

In [18]:

```
from faker.providers import misc, date_time, address
import random
```

In [19]:

```
order_ids = [fake.bothify(text = '????####', letters = 'ABCDE') for _ in range(100)]
order_dates = pd.date_range(start = "2022-01-01", end = "2022-04-10", periods = 100)
customers = [fake.name() for _ in range(100)]
regions = [fake.random_element(elements = ('East','West','North','South')) for _ in range
categories = ['Grocery' for _ in range(100)]
subcategories = [fake.random_element(elements = ('Bakery','Canned Goods','Dairy','Produce
grocery_words = ['break','milk','cheese','eggs','butter','flour','suger','rice','pasta']
product = [fake.random_element(elements = grocery_words) for _ in range(100)]
states = [fake.state() for _ in range(100)]

sales = [round(random.uniform(500,10000),2) for _ in range(100)]
profit = [round(random.uniform(50,2000),2) for _ in range(100)]
quantity = [random.randint(1,20) for _ in range(100)]
```

In [20]:

```
data = {"Order_ID" : order_ids,
       'Order_Date': order_dates,
       'Customer':customers,
       'Region': regions,
       'Category':categories,'Sub_Category':subcategories,"Product":product,
       'State':states, 'Sales':sales,"Profit":profit,"Quantity":quantity}
```

In [21]:

```
data.keys()
```

Out[21]:

```
dict_keys(['Order_ID', 'Order_Date', 'Customer', 'Region', 'Category', 'Su
b_Category', 'Product', 'State', 'Sales', 'Profit', 'Quantity'])
```

In [22]:

```
df = pd.DataFrame(data)
```

In [23]:

```
df.head()
```

Out[23]:

	Order_ID	Order_Date	Customer	Region	Category	Sub_Category	Product	State
0	EEEC7458	2022-01-01	Gatik Shroff	East	Grocery	Dairy	break	New York
1	CCCE3847	2022-01-02	Vanessa Sonnino	East	Grocery	Produce	butter	Tamil Nadu
2	EEBD2250	2022-01-03	Benedetto Montalti	North	Grocery	Canned Goods	flour	Telangana
3	DABE3293	2022-01-04	Jacqueline Harris	South	Grocery	Canned Goods	cheese	Idaho
4	CDAA1805	2022-01-05	Vivaan Boase	South	Grocery	Dairy	milk	Indiana

In [ ]:

```

```