

```
In [1]: !pip install faker
```

```
Requirement already satisfied: faker in c:\users\rahul\anaconda3\lib\site-packages (26.0.0)  
Requirement already satisfied: python-dateutil>=2.4 in c:\users\rahul\anaconda3\lib\site-packages (from faker) (2.8.2)  
Requirement already satisfied: six>=1.5 in c:\users\rahul\anaconda3\lib\site-packages (from python-dateutil>=2.4->faker) (1.16.0)
```

```
In [2]: from faker import Faker  
fake = Faker()
```

```
In [3]: print("Name:", fake.name())  
  
# Generate a fake address  
print("Address:", fake.address())  
  
# Generate a fake email  
print("Email:", fake.email())  
  
# Generate a fake phone number  
print("Phone Number:", fake.phone_number())  
  
# Generate a fake date of birth  
print("Date of Birth:", fake.date_of_birth())
```

```
Name: Jennifer Scott  
Address: 72358 Carolyn Motorway  
Lake Jennifer, AS 19702  
Email: antonio85@example.org  
Phone Number: 2402260029  
Date of Birth: 1944-10-12
```

```
In [4]: fake_fr = Faker('fr_FR')  
  
print("French Name:", fake_fr.name())  
print("French Address:", fake_fr.address())
```

```
French Name: Sophie Pelletier  
French Address: 2, boulevard Noël Charles  
97259 Poulain-la-Forêt
```

```
In [5]: fake.seed_instance(42)  
  
print("Consistent Name:", fake.name())
```

```
Consistent Name: Allison Hill
```

```
In [6]: from faker import Faker  
from faker.providers import BaseProvider  
  
class MyProvider(BaseProvider):  
    def magic_spell(self):  
        return 'Expecto Patronum!'  
  
# Add custom provider  
fake = Faker()  
fake.add_provider(MyProvider)  
  
print("Magic Spell:", fake.magic_spell())
```

```
Magic Spell: Expecto Patronum!
```

```
In [7]: # Generate fake user profiles
user_profiles = [fake.profile() for _ in range(5)]
print(user_profiles)
```

```
[{'job': 'TEFL teacher', 'company': 'Fitzpatrick, Ramirez and Palmer', 'ssn': '706-12-1200', 'residence': '204 Peterson Centers\nNorth Anthony, DC 41254', 'current_location': (Decimal('38.3540835'), Decimal('49.810820')), 'blood_group': 'A-', 'website': ['http://parker-wilson.com/', 'http://ingram.com/', 'http://www.good-wright.com/', 'https://www.jackson.info/'], 'username': 'anthonyford', 'name': 'Shannon Rodriguez', 'sex': 'F', 'address': '269 Vazquez Knolls\nEast Aliciaberg, MT 16823', 'mail': 'zwalters@yahoo.com', 'birthdate': datetime.date(1913, 7, 6)}, {'job': 'Farm manager', 'company': 'Gaines-Jones', 'ssn': '358-26-8065', 'residence': '9362 Ryan Drive\nGinaburgh, OH 65723', 'current_location': (Decimal('84.7064175'), Decimal('103.053720')), 'blood_group': 'O-', 'website': ['https://taylor.info/', 'https://www.walters-marshall.com/'], 'username': 'sharonpierce', 'name': 'Lauren Brown', 'sex': 'F', 'address': '6822 Miller Common Suite 476\nAlexanderton, ND 45003', 'mail': 'adam47@hotmail.com', 'birthdate': datetime.date(1918, 7, 28)}, {'job': 'Accountant, chartered', 'company': 'Strong-Miller', 'ssn': '376-41-5328', 'residence': '6272 Angela Plaza Suite 989\nMckinneyside, ME 99129', 'current_location': (Decimal('44.121327'), Decimal('139.648123')), 'blood_group': 'B+', 'website': ['http://barton-rios.com/', 'https://www.becker.net/', 'http://www.allen.net/'], 'username': 'jose68', 'name': 'Sarah Diaz', 'sex': 'F', 'address': '02303 Carol Mountains Suite 385\nPort Philip, NH 78272', 'mail': 'johnsonlauren@gmail.com', 'birthdate': datetime.date(1921, 1, 27)}, {'job': 'Operational researcher', 'company': 'Mann, Young and Powers', 'ssn': '540-79-6454', 'residence': '419 Caleb Station\nPort Stephen, DC 07018', 'current_location': (Decimal('-39.712857'), Decimal('25.202229')), 'blood_group': 'B-', 'website': ['https://www.walsh.info/', 'http://www.cook.info/', 'https://martin.com/'], 'username': 'mistycook', 'name': 'Michelle Hines', 'sex': 'F', 'address': '369 Warren Parks Suite 823\nThomasborough, ND 39269', 'mail': 'edwardsjennifer@gmail.com', 'birthdate': datetime.date(1997, 11, 18)}, {'job': 'Hydrologist', 'company': 'Pugh and Sons', 'ssn': '390-32-0640', 'residence': '6168 Brenda Manor\nMeganborough, MH 56048', 'current_location': (Decimal('-68.7362935'), Decimal('71.066930')), 'blood_group': 'A+', 'website': ['http://www.bradshaw.com/', 'https://www.flores-murphy.com/'], 'username': 'laurathornton', 'name': 'Justin Summers', 'sex': 'M', 'address': '56601 James Crossing\nNorth Lorishire, IL 18736', 'mail': 'omar30@yahoo.com', 'birthdate': datetime.date(1947, 10, 26)}]
```

```
In [8]: # Generate a List of fake transactions
transactions = [{"amount": fake.random_number(digits=4), "date": fake.date()} for _ in range(10)]
print(transactions)
```

```
[{'amount': 7751, 'date': '1973-10-29'}, {'amount': 3703, 'date': '2009-06-02'}, {'amount': 6336, 'date': '1977-08-28'}, {'amount': 8428, 'date': '1980-06-02'}, {'amount': 4951, 'date': '2014-04-11'}, {'amount': 5721, 'date': '1992-10-28'}, {'amount': 4933, 'date': '1977-03-06'}, {'amount': 6649, 'date': '2024-04-03'}, {'amount': 4907, 'date': '2020-02-10'}, {'amount': 4186, 'date': '1996-06-08'}]
```

```
In [9]: # Replace real names with fake names in a dataset
real_names = ['Alice', 'Bob', 'Charlie']
fake_names = [fake.name() for _ in real_names]
print(fake_names)
```

```
['Rebecca Perkins', 'Nicholas Brown', 'Dale Ross']
```

```
In [10]: # Generate emails for a specific domain
company_domain = "example.com"
emails = [f"{fake.first_name().lower()}.{fake.last_name().lower()}@{company_domain}" for _ in range(5)]
print(emails)
```

```
['edwin.singleton@example.com', 'evan.ward@example.com', 'timothy.richardson@example.com', 'nancy.reynolds@example.com', 'michelle.bridges@example.com']
```

```
In [11]: # Generate fake orders with customer info
orders = [{"customer": fake.name(), "product": fake.word(), "price": fake.random_number(digits=4)} for _ in range(10)]
```

```
print(orders)
```

```
[{'customer': 'Kimberly Jones', 'product': 'site', 'price': 708}, {'customer': 'William Mclean', 'product': 'any', 'price': 349}, {'customer': 'Jeanne Lee', 'product': 'fish', 'price': 122}, {'customer': 'Matthew Smith', 'product': 'organization', 'price': 205}, {'customer': 'Hector Smith', 'product': 'study', 'price': 606}, {'customer': 'Pedro Cole', 'product': 'leave', 'price': 378}, {'customer': 'David Tran', 'product': 'conference', 'price': 254}, {'customer': 'Timothy Sloan', 'product': 'grow', 'price': 779}, {'customer': 'Scott Mathews', 'product': 'him', 'price': 402}, {'customer': 'Deanna Sullivan', 'product': 'wind', 'price': 193}]
```

In []: