1. Disease Screening in Different Age Groups:

Scenario: Disease Screening

A particular disease is more common in older people. Let's consider two age groups: under 50 and over 50. The prevalence of the disease is 2% in the under 50 age group and 8% in the over 50 age group. A screening test has different accuracy levels for these age groups: 95% accuracy for under 50s and 90% for over 50s. However, the test has a 5% false positive rate in both groups.

The task is to calculate the probability of having the disease given a positive test result in each age group.

In [5]:
```
def bayesian_update_disease(prior, accuracy, false_positive_rate):
    numerator = accuracy * prior
    denominator = numerator + false_positive_rate * (1 - prior)
    return numerator / denominator

priority_under50 = 0.02
accuracy_under50 = 0.95
false_positive_rate = 0.05

priority_over50 = 0.08
accuracy_over50 = 0.90

prob_under50 = bayesian_update_disease(priority_under50, accuracy_unde
prob_over50 = bayesian_update_disease(priority_over50, accuracy_over5(

print("Probability of disease given positive test (Under 50):", prob_u
print("Probability of disease given positive test (Over 50):", prob_ov
```

```
Probability of disease given positive test (Under 50): 0.27941176470
58823
Probability of disease given positive test (Over 50): 0.610169491525
4237
```

2. Weather Forecast

Scenario: Predicting Rain

Suppose a weather forecast model predicts rain with a 70% accuracy rate. However, it also incorrectly predicts rain 30% of the time when it's not going to rain. Let's say the actual chance of rain on any given day in a particular region is 20%.

We want to calculate the probability of it actually raining given that the forecast predicts rain.

In [4]:
```
def bayesian_update_weather(prior, likelihood, false_positive_rate):
    numerator = likelihood * prior
    denominator = numerator + false_positive_rate * (1 - prior)
    return numerator / denominator

priority_rain = 0.20
likelihood_rain = 0.70
false_positive_rate_rain = 0.30

prob_rain_given_forecast = bayesian_update_weather(priority_rain, like

print("Probability of rain given forecast predicts rain:", prob_rain_g
```

```
Probability of rain given forecast predicts rain: 0.3684210526315789
```

In [ ]:
```

```