Data generation:
Generated 100 synthetic restaurant examples with program functions fix seeded.

- Create all attributes and their values: Alternate, Bar, Fri/Sat, Hungry, Raining, Reservation, Type, Patrons, Price, WaitEstimate
- Each example is created randomly with 1 value for every attributes
- Use a hard code decision tree that ensure the correct wait or no wait for every combination.
- Make a balanced dataset (50 wait, 50 won't) by accepting generated examples only until each class reaches 50.
- When train create stratified train/test splits (20/80, 30/70, 40/60) that preserve class balance in each split.
- Train set has equal % of wait and no wait
- Test set also has equal % of wait and no wait

Label	Count
Wait	50
No Wait	50
Total	100

We will evaluate 2 models:

- Decision Tree
- Neural Network

Each will trained using 3 training set sizes:

- 20% training / 80% testing
- 30% training / 70% testing
- 40% training / 60% testing

Train split	Decision Tree Accuracy	Neural Network Accuracy
20% train (20 train, 80 test)	0.512	0.825
30% train (30 train, 70 test)	0.429	0.814
40% train (40 train, 60 test)	0.550	0.917

Observation:

- Generally, when the training set increase,s the accuracy of the test set also increases
- The Neural Network Model performs better across the split

- The Decision Tree seems to fluctuate across the split, showing signs of overfitting and hearing too much noise froma small dataset