Lab 3

Lab 3: Singing a song

```
import pandas as pd
xmas = pd.read_csv("https://www.dropbox.com/scl/fi/qxaslqqp5p08i1650rpc4/xmas.csv?rlkey=erdxi7jbh7p
xmas
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location
0	1	first	partridge	NaN	NaN	in a pear tree
1	2	second	dove	NaN	turtle	NaN
2	3	third	hen	NaN	french	NaN
3	4	fourth	bird	NaN	calling	NaN
4	5	fifth	ring	NaN	golden	NaN
5	6	sixth	goose	a-laying	NaN	NaN
6	7	seventh	swan	a-swimming	NaN	NaN
7	8	eighth	maid	a-milking	NaN	NaN
8	9	ninth	lady	dancing	NaN	NaN
9	10	tenth	lord	a-leaping	NaN	NaN
10	11	eleventh	piper	piping	NaN	NaN
11	12	twelfth	drummer	drumming	NaN	NaN

Function 1: pluralize_gift()

```
def pluralize_gift(gift):
  """
  Returns plural of a noun

  Parameters

  ----------
  gift: str
    A noun

  Return
  ------
  str
    Plural version
  """

  if gift == "goose":
    gift = "geese"
  elif gift == "lady":
    gift = "ladies"
  else:
    gift = gift + "s"

  return gift
```

```
# Should work
pluralize_gift("goose")

#change the line
xmas["Plural.Gift"] = xmas["Gift.Item"].apply(pluralize_gift)
xmas["Gift.Item"].apply(pluralize_gift)
```

```
0     partridges
1          doves
2           hens
3          birds
4          rings
5          geese
6          swans
7          maids
8         ladies
9          lords
10        pipers
11      drummers
Name: Gift.Item, dtype: object
```

Function 2: make_phrase()

```
def make_phrase(num, num_word, item, verb, adjective, location):
    """
    Returns a phrase of the song

    Parameters
    ----------
    <your detailed documentation here>

    Return
    ------
    str
        Phrase of the song
    """

    # Replace NaN with blank strings
    if pd.isnull(verb):
        verb = ""
    if pd.isnull(adjective):
        adjective = ""
    if pd.isnull(location):
        location = ""

    # Pluralize the item if the day number is larger than 1
    if num > 1:
        item = pluralize_gift(item)
    # Create the phrase
    if num == 1:
        article = 'an' if item[0] in 'aeiou' else 'a'
        phrase = f"{article} {item} {location}".strip()
    else:
        phrase = f"{num_word} {adjective} {item} {verb} {location}".strip()

    return phrase
```

```
Day = pd.DataFrame(xmas)

# Create a dictionary for mapping
number_words_mapping = {
    'first': 'one',
    'second': 'two',
    'third': 'three',
    'fourth': 'four',
    'fifth': 'five',
    'sixth': 'six',
    'seventh': 'seven',
    'eighth': 'eight',
    'ninth': 'nine',
    'tenth': 'ten',
    'eleventh': 'eleven',
    'twelfth': 'twelve'
}

# Apply the mapping to the DataFrame
Day["Day.in.Words"] = Day["Day.in.Words"].map(number_words_mapping)

Day
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Plural.Gift	Full.Phrase
0	1	one	partridge	NaN	NaN	in a pear tree	partridges	a partridge in a pear tree
1	2	two	dove	NaN	turtle	NaN	doves	second turtle doves
2	3	three	hen	NaN	french	NaN	hens	third french hens
3	4	four	bird	NaN	calling	NaN	birds	fourth calling birds
4	5	five	ring	NaN	golden	NaN	rings	fifth golden rings
5	6	six	goose	a-laying	NaN	NaN	geese	sixth geese a-laying
6	7	seven	swan	a-swimming	NaN	NaN	swans	seventh swans a-swimming
7	8	eight	maid	a-milking	NaN	NaN	maids	eighth maids a-milking
8	9	nine	lady	dancing	NaN	NaN	ladies	ninth ladies dancing
9	10	ten	lord	a-leaping	NaN	NaN	lords	tenth lords a-leaping
10	11	eleven	piper	piping	NaN	NaN	pipers	eleventh pipers piping
11	12	twelve	drummer	drumming	NaN	NaN	drummers	twelfth drummers drumming

```
print(make_phrase(1, "one", "patridge", "", "", "in a pear tree"))
xmas['Full.Phrase'] = xmas.apply(lambda row: make_phrase(row["Day"], row["Day.in.Words"], row["Gift
xmas["Plural.Gift"] = xmas["Gift.Item"].apply(pluralize_gift)
xmas
```

```
a patridge in a pear tree
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Plural.Gift	Full.Phrase
0	1	one	partridge	NaN	NaN	in a pear tree	None	a partridge in a pear tree
1	2	two	dove	NaN	turtle	NaN	None	two turtle None
2	3	three	hen	NaN	french	NaN	None	three french None
3	4	four	bird	NaN	calling	NaN	None	four calling None
4	5	five	ring	NaN	golden	NaN	None	five golden None
5	6	six	goose	a-laying	NaN	NaN	None	six None a-laying
6	7	seven	swan	a-swimming	NaN	NaN	None	seven None a-swimming
7	8	eight	maid	a-milking	NaN	NaN	None	eight None a-milking
8	9	nine	lady	dancing	NaN	NaN	None	nine None dancing
9	10	ten	lord	a-leaping	NaN	NaN	None	ten None a-leaping
10	11	eleven	piper	piping	NaN	NaN	None	eleven None piping
11	12	twelve	drummer	drumming	NaN	NaN	None	twelve None drumming

Function 3: sing_day()

```
def sing_day(dataset, num, phrase_col):
    """
    Sings the song for a specific day

    Parameters
    ----------
    dataset: pd.DataFrame
        The dataset containing song phrases
    num: int
        The day number
    phrase_col: str
        The column name containing phrases
    Return
    ------
    str
        The song for the specified day
    """

    # Step 1: Setup the intro line
    num_words = {1: 'first', 2: 'second', 3: 'third', 4: 'fourth', 5: 'fifth',
                 6: 'sixth', 7: 'seventh', 8: 'eighth', 9: 'ninth', 10: 'tenth',
                 11: 'eleventh', 12: 'twelfth'}
    num_word = num_words[num]
    intro = f"On the {num_word} day of Christmas, my true love sent to me:"

    # Step 2: Sing the gift phrases in reverse order
    gifts = '\n'.join(dataset[phrase_col][:num][::-1])
    if num > 1:
        gifts = gifts.replace("and a", "and", 1)  # Remove 'and' before 'a partridge in a pear tree

    # Step 3: Put it all together and return
    song = f"{intro}\n{gifts}"

    return song
```

```
print(sing_day(xmas, 3, "Full.Phrase"))
```

```
On the third day of Christmas, my true love sent to me:
three french None
two turtle None
a partridge in a pear tree
```

#Test Entire Song

```
print(sing_day(xmas, 12, "Full.Phrase"))
```

```
On the twelfth day of Christmas, my true love sent to me:
twelve  None drumming
eleven  None piping
ten  None a-leaping
nine  None dancing
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree
```

```
# Function 1: pluralize_gift()
def pluralize_gift(gift):

# Function 2: make_phrase()
    def make_phrase(num, num_word, item, verb, adjective, location):

# Function 3: sing_day()
      def sing_day(dataset, num, phrase_col):

# Create a sequence of numbers from 0 to 12 representing the days
        days_sequence = pd.Series(range(0, 12))

# Use .map() to generate phrases for each day and concatenate them
song_phrases = days_sequence.map(lambda day: sing_day(xmas, num=day, phrase_col="Full.Phrase")+"\n"
# Print the entire song
print("\n".join(song_phrases))
```

```
On the first day of Christmas, my true love sent to me:
a partridge in a pear tree

On the second day of Christmas, my true love sent to me:
two turtle None
a partridge in a pear tree

On the third day of Christmas, my true love sent to me:
three french None
two turtle None
a partridge in a pear tree

On the fourth day of Christmas, my true love sent to me:
four calling None
three french None
two turtle None
a partridge in a pear tree

On the fifth day of Christmas, my true love sent to me:
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the sixth day of Christmas, my true love sent to me:
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the seventh day of Christmas, my true love sent to me:
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the eighth day of Christmas, my true love sent to me:
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree
```

```
On the ninth day of Christmas, my true love sent to me:
nine  None dancing
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the tenth day of Christmas, my true love sent to me:
ten  None a-leaping
nine  None dancing
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the eleventh day of Christmas, my true love sent to me:
eleven  None piping
ten  None a-leaping
nine  None dancing
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree

On the twelfth day of Christmas, my true love sent to me:
twelve  None drumming
eleven  None piping
ten  None a-leaping
nine  None dancing
eight  None a-milking
seven  None a-swimming
six  None a-laying
five golden None
four calling None
three french None
two turtle None
a partridge in a pear tree
```

```
xmas2 = pd.read_csv("https://www.dropbox.com/scl/fi/p9x9k8xwuzs9rhp582vfy/xmas_2.csv?rlkey=kvc3j3lm
xmas2
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location
0	1	first	email	NaN	NaN	from Cal Poly
1	2	second	point	NaN	meal	NaN
2	3	third	pen	NaN	lost	NaN
3	4	fourth	review	NaN	course	NaN
4	5	fifth	exam	NaN	practice	NaN
5	6	sixth	grader	grading	NaN	NaN
6	7	seventh	senior	stressing	NaN	NaN
7	8	eighth	mom	a-calling	NaN	NaN
8	9	ninth	party	bumping	NaN	NaN
9	10	tenth	load	of laundry	NaN	NaN
10	11	eleventh	friend	goodbye-ing	NaN	NaN
11	12	twelfth	hour	sleeping	NaN	NaN

```
xmas2["Day.in.Words"] = list(map(Day_in_Words_map.get, xmas2['Day']))

xmas2['Full.Phrase'] = xmas2.apply(lambda row: make_phrase(row['Day'], row['Day.in.Words'], row['Gi
```

```
Day = pd.DataFrame(xmas2)

# Create a dictionary for mapping
number_words_mapping = {
    'first': 'one',
    'second': 'two',
    'third': 'three',
    'fourth': 'four',
    'fifth': 'five',
    'sixth': 'six',
    'seventh': 'seven',
    'eighth': 'eight',
    'ninth': 'nine',
    'tenth': 'ten',
    'eleventh': 'eleven',
    'twelfth': 'twelve'
}

# Apply the mapping to the DataFrame
Day["Day.in.Words"] = Day["Day.in.Words"].map(number_words_mapping)

Day
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Full.Phrase
0	1	one	email	NaN	NaN	from Cal Poly	an email from Cal Poly
1	2	two	point	NaN	meal	NaN	second meal None
2	3	three	pen	NaN	lost	NaN	third lost None

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Full.Phrase
3	4	four	review	NaN	course	NaN	fourth course None
4	5	five	exam	NaN	practice	NaN	fifth practice None
5	6	six	grader	grading	NaN	NaN	sixth None grading
6	7	seven	senior	stressing	NaN	NaN	seventh None stressing
7	8	eight	mom	a-calling	NaN	NaN	eighth None a-calling
8	9	nine	party	bumping	NaN	NaN	ninth None bumping
9	10	ten	load	of laundry	NaN	NaN	tenth None of laundry
10	11	eleven	friend	goodbye-ing	NaN	NaN	eleventh None goodbye-ing
11	12	twelve	hour	sleeping	NaN	NaN	twelfth None sleeping

```
# Function 1: pluralize_gift()
def pluralize_gift(gift):
    # Implementation here

# Function 2: make_phrase()
  def make_phrase(num, num_word, item, verb, adjective, location):
    # Implementation here

# Function 3: sing_day()
    def sing_day(dataset, num, phrase_col):
    # Implementation here



# Map day numbers to their corresponding words
      xmas2["Day.in.Words"] = xmas2['Day'].map(Day_in_Words_map)

# Create the Full.Phrase column using make_phrase function
xmas2['Full.Phrase'] = xmas2.apply(lambda row: make_phrase(row['Day'], row['Day.in.Words'], row['Gi

# Print the modified DataFrame
xmas2
```

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Full.Phrase
0	1	one	email	NaN	NaN	from Cal Poly	an email from Cal Poly
1	2	two	point	NaN	meal	NaN	two meal None
2	3	three	pen	NaN	lost	NaN	three lost None
3	4	four	review	NaN	course	NaN	four course None
4	5	five	exam	NaN	practice	NaN	five practice None
5	6	six	grader	grading	NaN	NaN	six None grading
6	7	seven	senior	stressing	NaN	NaN	seven None stressing
7	8	eight	mom	a-calling	NaN	NaN	eight None a-calling
8	9	nine	party	bumping	NaN	NaN	nine None bumping
9	10	ten	load	of laundry	NaN	NaN	ten None of laundry

	Day	Day.in.Words	Gift.Item	Verb	Adjective	Location	Full.Phrase
10	11	eleven	friend	goodbye-ing	NaN	NaN	eleven None goodbye-ing
11	12	twelve	hour	sleeping	NaN	NaN	twelve None sleeping

```
print(sing_day(xmas2, 12, "Full.Phrase"))
```

```
On the twelfth day of Christmas, my true love sent to me:
twelve  None sleeping
eleven  None goodbye-ing
ten  None of laundry
nine  None bumping
eight  None a-calling
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly
```

```
# Function 1: pluralize_gift()
def pluralize_gift(gift):

# Function 2: make_phrase()
    def make_phrase(num, num_word, item, verb, adjective, location):

# Function 3: sing_day()
      def sing_day(dataset, num, phrase_col):

# Create a sequence of numbers from 0 to 12 representing the days
        days_sequence = pd.Series(range(0, 12))

# Use .map() to generate phrases for each day and concatenate them
song_phrases = days_sequence.map(lambda day: sing_day(xmas2, num=day, phrase_col="Full.Phrase")+ "\

# Print the entire song
print("\n".join(song_phrases))
```

```
On the first day of Christmas, my true love sent to me:
an email from Cal Poly

On the second day of Christmas, my true love sent to me:
two meal None
an email from Cal Poly

On the third day of Christmas, my true love sent to me:
three lost None
two meal None
an email from Cal Poly

On the fourth day of Christmas, my true love sent to me:
```

```
four course None
three lost None
two meal None
an email from Cal Poly

On the fifth day of Christmas, my true love sent to me:
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the sixth day of Christmas, my true love sent to me:
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the seventh day of Christmas, my true love sent to me:
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the eighth day of Christmas, my true love sent to me:
eight  None a-calling
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the ninth day of Christmas, my true love sent to me:
nine  None bumping
eight  None a-calling
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the tenth day of Christmas, my true love sent to me:
ten  None of laundry
nine  None bumping
eight  None a-calling
```

```
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the eleventh day of Christmas, my true love sent to me:
eleven  None goodbye-ing
ten  None of laundry
nine  None bumping
eight  None a-calling
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly

On the twelfth day of Christmas, my true love sent to me:
twelve  None sleeping
eleven  None goodbye-ing
ten  None of laundry
nine  None bumping
eight  None a-calling
seven  None stressing
six  None grading
five practice None
four course None
three lost None
two meal None
an email from Cal Poly
```