💖 Digital Literacy Spring 2023 Final exam: 3 Questions

Read the following before you begin:

- You are allowed to use open sources including course lessons, Google, ChatGPT, and anything but other classmates' answers :-)
- You are allowed to add more code cells when needed.
- At the end, you are asked to uplade the coding page in pdf (**DL_yourname.pdf**). (Plus an image file you generated in Question 3)
- When coding is done, save this page on your Github DL repository. (as **DL_final.ipynb**)

💧 Question 1: "Mad libs game" (말이 되지 않는 문장 만들기)

- Step 1)

1. 사용자가 명사 하나, 동사 하나, 형용사 하나를 각각 별도로 입력하게 합니다. (질문에 'Write one noun word:' 와 같이 가이드를 해 줄 것. 힌트: input 함수를 세 번 사용함)
2. 입력된 세 개의 단어는 문장 틀 "The 형용사 명사 동사 in the forest." 속으로 자동으로 배치되어 출력을 하면 말이 되지 않을 수 있지만 문법적인 문장이 나오게 되는 게임입니다.

Example 1) If I input adjective as 'pretty', noun as 'book', verb as 'smiled', your output should be "The pretty book smiled in the forest.

Example 2) If I input adjective as 'cold', noun as 'candy', verb as 'flies', then your output should be "The cold candy flies in the forest.

```
adjective = input("Write one adjective word: ")
noun = input("Write one noun word: ")
verb = input("Write one verb word: ")

sentence = "The {} {} {} in the forest.".format(adjective, noun, verb)
print(sentence)
```

```
    Write one adjective word: active
    Write one noun word: strawberry
    Write one verb word: speak
    The active strawberry speak in the forest.
```

- Step 2) 위에서 만들어진 문장이 오디오로 출력되도록 하시요. gTTS 사용

gtts는 아래코드를 활용해도 됩니다.

```
# Necessary libraries installed
!pip install gtts
from gtts import gTTS
from IPython.display import Audio


# tts(text) is defined
def tts(mytext):
  tts = gTTS(mytext)
```

```
    tts.save('sample.mp3')
    return Audio('sample.mp3', autoplay=True)
```

```
# Necessary libraries installed
!pip install gtts
from gtts import gTTS
from IPython.display import Audio
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting gtts
  Downloading gTTS-2.3.2-py3-none-any.whl (28 kB)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.10/dist-packages (from gtts) (2.27.1)
Requirement already satisfied: click<8.2,>=7.1 in /usr/local/lib/python3.10/dist-packages (from gtts) (8.1.3)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.10/dist-packages (from request
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->g
Installing collected packages: gtts
Successfully installed gtts-2.3.2
```

```
# tts(text) is defined
def tts(mytext):
  tts = gTTS(mytext)
  tts.save('sample.mp3')
  return Audio('sample.mp3', autoplay=True)
```

```
text = "The active strawberry speak in the forest."
tts(text)
```

0:03 / 0:03

💧 Question 2: 동사의 과거형 발음해 주는 문장 만들기

- Step 1) 아래 단어 리스트에서 random 함수를 이용해서 임의로 단어를 하나 선택하게 하고, 단어가 선택되면

> The past form of ***kick*** is **kicked**.

라고 문장이 출력되도록 코드를 작성하시오. 단어리스트는 아래를 사용하시오.

wlist = ["kick","play","print","seek","need"]

```
import random

wlist = ["kick", "play", "print", "seek", "need"]

verb = random.choice(wlist)
past_form = verb + "ed"

sentence = f"The past form of {verb} is {past_form}."
print(sentence)
```

```
The past form of kick is kicked.
```

💧 Question 3

Using the following text, complete the following tasks:

Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known for his relentless work ethic, always gathering food and strengthening his ant hill, even when the sun was at its peak.

One day, a grasshopper named Gerry hopped by, laughing at Archie. "Why do you work so hard, Archie?" Gerry asked. "Come, bask in the sun with me."

Archie, wiping his brow, replied, 'I am investing my time now, Gerry, so that I do not need to have to worry later.'

Gerry shrugged and hopped away, leaving Archie to his work. The summer passed, and the harsh winter arrived. Gerry, who had spent his summer hopping and basking, found himself cold and hungry. He hopped over to Archie's hill, shivering.

Archie, who was comfortably nestled inside his warm and well-stocked hill, saw Gerry. He invited him in, sharing his food and warmth. Gerry, grateful yet regretful, understood the lesson Archie had tried to teach him in the summer.

From that day forward, Gerry learned to invest his time wisely, understanding that hard work today can ensure comfort and security tomorrow. And Archie, with his wisdom and foresight, became a symbol of diligence and wise investment in the forest.

- Step 1) 위의 텍스트를 'mytext'라는 변수이름으로 저장하시오. 'mytext'의 데이터 타입은 string이어야 합니다.

```
# Step 1

mytext = '''Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known for his relentless

One day, a grasshopper named Gerry hopped by, laughing at Archie. "Why do you work so hard, Archie?" Gerry asked. "Come,

Archie, wiping his brow, replied, 'I am investing my time now, Gerry, so that I do not need to have to worry later.'

Gerry shrugged and hopped away, leaving Archie to his work. The summer passed, and the harsh winter arrived. Gerry, who

Archie, who was comfortably nestled inside his warm and well-stocked hill, saw Gerry. He invited him in, sharing his foc

From that day forward, Gerry learned to invest his time wisely, understanding that hard work today can ensure comfort ar
```

- Step 2) 불필요한 스페이스 제거, 개별 단어를 리스트로 만들어 'mywords'라는 이름으로 저장한 후, len(mywords)를 이용하여 마지막으로 'The story contains N words."가 출력되도록 하시오. (N은 총 단어수)

```
text = "Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known for his relentless work

# Remove unnecessary spaces
text = text.replace(", ", " ")

# Split the text into words and store them in a list
mywords = text.split()

# Calculate the total number of words
total_words = len(mywords)

# Print the result
print("The story contains {} words.".format(total_words))
```

```
    The story contains 205 words.
```

- Step 3) mytext를 문장 단위로 나누고 'mysentence'로 저장하고, len(mysentence)를 이용하여 마지막으로 'The story contains N sentences."가 출력되도록 하시오. (N은 총 문장수)

```
text = "Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known for his relentless work

# Split the text into sentences and store them in a list
mysentence = text.split(". ")

# Calculate the total number of sentences
total_sentences = len(mysentence)

# Print the result
print("The story contains {} sentences.".format(total_sentences))
```

```
The story contains 13 sentences.
```

- Step 4) mytext를 가지고 워드클라우드를 만드시오.

```
# Step 3

#@markdown Wordcloud: First try with the text above. Once you find it working, get your text (from chatGPT) ready to pas
# Install necessary packages
!pip install wordcloud matplotlib

# Import necessary libraries
from wordcloud import WordCloud
import matplotlib.pyplot as plt

# Your text data
text = input("Paste your text here: ")

# Create a WordCloud object
wordcloud = WordCloud(width=800, height=800, background_color='white', stopwords=None, min_font_size=10).generate(text)

# Plot the WordCloud image
plt.figure(figsize=(8, 8), facecolor=None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad=0)

# Save the plot as an image file
plt.savefig("wordcloud.png", format="png")

# Show the plot
plt.show()
```

Wordcloud: First try with the text above. Once you find it working, get your text (from chatGPT) ready to paste when a pop-up window appears

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: wordcloud in /usr/local/lib/python3.10/dist-packages (1.8.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.7.1)
Requirement already satisfied: numpy>=1.6.1 in /usr/local/lib/python3.10/dist-packages (from wordcloud) (1.22.4)
Requirement already satisfied: pillow in /usr/local/lib/python3.10/dist-packages (from wordcloud) (8.4.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.0
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.3
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (23.1
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.0
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->ma
Paste your text here: Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known fo
```

Gerry
work
hill
time
hopped
food
hard
named
ant
sun
upon
day
lived
dense
always
diligent
asked
invited
security
invest
winter
today
ensure
comfortably
shrugged
even
foresight
ethic
grasshopper
tomorrow
Come
tried
investment
replied
lesson
teach
diligence
sharing
bask
saw
One
leaving
hungry
found
basking
later
strengthening
wisdom
laughing
stocked
symbol
warmth
wise
forward
shivering
cold
known
peak
now

- Step 5) mytext에서 문장 번호를 선택하면 해당 문장이 오디오로 재생되도록 코드를 작성하시오. (힌트, gtts, Audio를 불러오고, 코드에 #@param을 이용하여 리스트로 문장번호를 작성한 후, text = 선택된 문장이 되도록 한 후, Audio()함수를 이용하여 음성을 합성합니다.)
- 아래 제공한 코드는 #를 풀고 사용하실 수 있습니다.
- gtts는 아래코드를 활용해도 됩니다.

```
# Necessary libraries installed
!pip install gtts
from gtts import gTTS
from IPython.display import Audio


# tts(text) is defined
def tts(mytext):
tts = gTTS(mytext)
```

```
        tts.save('sample.mp3')
        return Audio('sample.mp3', autoplay=True)
```

```
# Necessary libraries installed
!pip install gtts
from gtts import gTTS
from IPython.display import Audio
import random

# Define the function to generate audio from text
def tts(mytext):
    tts = gTTS(mytext)
    tts.save('sample.mp3')
    return Audio('sample.mp3', autoplay=True)

# Define your text
mytext = "Once upon a time, in a dense forest, lived a diligent ant named Archie. Archie was known for his relentless wo

# Split the text into sentences and store them in a list
mysentence = mytext.split(". ")

# Generate a random sentence number
sentence_num = random.randint(1, len(mysentence))

# Select the sentence based on the generated number
selected_sentence = mysentence[sentence_num - 1]

# Generate audio and play it
tts(selected_sentence)
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: gtts in /usr/local/lib/python3.10/dist-packages (2.3.2)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.10/dist-packages (from gtts) (2.27.1)
Requirement already satisfied: click<8.2,>=7.1 in /usr/local/lib/python3.10/dist-packages (from gtts) (8.1.3)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.10/dist-packages (from request
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.27->g
```

0:06 / 0:06

- Step 6: Question 3번에 있는 텍스트 내용을 상징하는 이미지를 온라인에서 접근 가능한 AI 도구를 사용하여 생성하고, 만들어진 이미지를 본인 Github에 저장한 후, 링크를 이용하여 아래 코드를 돌리면 해당 이미지가 보일 수 있도록 하시오.
- 추가로 맨 마지막에 이미지 파일을 업로드하게 되어 있습니다. (바탕화면에도 저장해 두기 바랍니다.)

```
from IPython.display import Image

Image(url='https://github.com/seonbeenkim/image../blob/main/DALL%C2%B7E%202023-06-14%2013.02.37%20-%20working%20deligen
```

😍 The End:

- 한 학기 동안 고생 많았고 여러분 협조 덕분에 잘 마무리된 것 같습니다. 2년 후에는 여러분 학생들과 재밌는 수업하고 있는 즐거운 상상을 해봅니다 :-)

<기말제출 안내>

- 작성한 코드를 **Github이 아니라 본인 Google Drive에** 'DL_final.ipynb'로 저장하세요. **시험이 모두 완료된 1시 이후 (본인 시간될 때)** 구글드라이브에서 해당 파일을 더블클릭하여 열면 코랩으로 열립니다. 이를 다시 Github DL repository에 저장해 두시기 바랍니다 (보관용) 채점시 확인용으로 참고할 필요가 있을 수 있습니다. File > Save a copy in Drive (나중에는 Save a copy in Github)
- 주의사항: pdf로 만들어 제출하는데 혹시라도 확인이 필요한 경우 제가 github을 방문하여 코드를 확인하는 경우가 생길 수 있으니, 꼭 (Google) Drive에 사본을 저장해 두세요.
- 완성된 코드는 중간고사 때와 마찬가지로 **pdf로 만들어 이미지파일과 함께 두 개의 파일을** 여기를 눌러 제출하시면 됩니다.
- Colab munu > File > print > save as pdf

✓ 0초 오후 1:09에 완료됨 ● ×