Function to create unigram and bigram count dictionaries.

Specific tasks:

- read text
- remove newlines
- tokenize
- create unigrams list
- create bigrams list
- -- bigrams as tuples, bigrams as strings
- create dicts for both unigrams and bigrams
- -- key: unigram/bigram
- -- value: count
- return these dicts

```
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize
from nltk.util import ngrams
from collections import Counter


def create_ngram_dicts(file_path):
    with open(file_path, 'r') as file:
        text = file.read()

        # Remove newlines
        text = text.replace('\n', ' ')

        # Tokenize
        tokens = word_tokenize(text)

        # tokens are already unigrams
        unigrams = tokens

        # Create bigrams, as list of paired tokens
        bigram_tokens = list(ngrams(tokens, 2))

        # Count occurrences of unigrams and bigram_strings
        unigram_counts = Counter(unigrams)
        bigram_counts = Counter(bigram_tokens)

        return unigram_counts, bigram_counts
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

- Loop thru the 3 langauges, using create_ngram_dicts to get unigram and bigram counts.
- Print these counts and store it all in a single large dictionary which contains the 6 smaller dictionaries.
- Pickle ngram_meta_dict

```
import pickle

languages = ['English','French','Italian']
ngram_meta_dict = {}

# loop thru the 3 languages
for language in languages:
  training_file = f'/LangId.train.{language}.txt'

  # use create_ngram_dicts to get unigram and bigram counts
  unigram_counts, bigram_counts = create_ngram_dicts(training_file)

  # print the counts
  # print(f"{language} Unigram Counts:", unigram_counts)
  print(f"{language} Bigram Counts:", bigram_counts)

  # store the counts in ngram_meta_dict
  if language not in ngram_meta_dict:
    ngram_meta_dict[language] = {}
  ngram_meta_dict[language]['unigram'] = unigram_counts
  ngram_meta_dict[language]['bigram'] = bigram_counts

# pickle ngram_meta_dict
with open('ngram_meta_dict_2.pkl','wb') as file:
  pickle.dump(ngram_meta_dict, file)

print("pickling complete!")
```

```
English Bigram Counts: Counter({('of', 'the'): 903, ('in', 'the'): 418, ('.', 'The'): 341, ('to', 'the'): 330, ('
French Bigram Counts: Counter({('l', "'"): 1786, ('d', "'"): 1158, ("'", '-'): 1040, ('de', 'la'): 723, ('de', 'l
Italian Bigram Counts: Counter({('l', "'"): 715, ('dell', "'"): 432, ('Presidente', ','): 227, (',', 'che'): 216,
pickling complete!
```