1. What is the name of the object used to tokenize sentences? `1 / 1 point`

   ● Tokenizer

   ○ CharacterTokenizer

   ○ TextTokenizer

   ○ WordTokenizer

   ✓ **Correct**
   That's right!

2. What is the name of the method used to tokenize a list of sentences? `1 / 1 point`

   ○ tokenize_on_text(sentences)

   ○ fit_to_text(sentences)

   ○ tokenize(sentences)

   ● fit_on_texts(sentences)

   ✓ **Correct**
   Correct!

3. Once you have the corpus tokenized, what's the method used to encode a list of sentences to use those tokens? `1 / 1 point`

   ● texts_to_sequences(sentences)

   ○ texts_to_tokens(sentences)

   ○ text_to_sequences(sentences)

   ○ text_to_tokens(sentences)

   ✓ **Correct**
   Correct!

4. When initializing the tokenizer, how do you specify a token to use for unknown words? `1 / 1 point`

   ○ unknown_token=<Token>

   ○ unknown_word=<Token>

   ○ out_of_vocab=<Token>

   ● oov_token=<Token>

   ✓ **Correct**
   That's right!

5. If you don't use a token for out of vocabulary words, what happens at encoding?   **1 / 1 point**

    ◉ The word isn't encoded, and is skipped in the sequence

    ○ The word isn't encoded, and is replaced by a zero in the sequence

    ○ The word isn't encoded, and the sequencing ends

    ○ The word is replaced by the most common token

> ✓ **Correct**
> Correct!

6. If you have a number of sequences of different lengths, how do you ensure that they are understood when fed   **1 / 1 point**
    into a neural network?

    ○ Process them on the input layer of the Neural Network using the pad_sequences property

    ○ Specify the input layer of the Neural Network to expect different sizes with dynamic_length

    ◉ Use the pad_sequences function from the tensorflow.keras.preprocessing.sequence namespace

    ○ Make sure that they are all the same length using the pad_sequences method of the tokenizer

> ✓ **Correct**
> That's right!

7. If you have a number of sequences of different length, and call pad_sequences on them, what's the default result?   **1 / 1 point**

    ○ Nothing, they'll remain unchanged

    ◉ They'll get padded to the length of the longest sequence by adding zeros to the beginning of shorter ones

    ○ They'll get padded to the length of the longest sequence by adding zeros to the end of shorter ones

    ○ They'll get cropped to the length of the shortest sequence

> ✓ **Correct**
> Correct!

8. When padding sequences, if you want the padding to be at the end of the sequence, how do you do it?   **1 / 1 point**

    ○ Call the padding method of the pad_sequences object, passing it 'post'

    ◉ Pass padding='post' to pad_sequences when initializing it

    ○ Pass padding='after' to pad_sequences when initializing it

    ○ Call the padding method of the pad_sequences object, passing it 'after'

> ✓ **Correct**
> That's right!