

API Documentation

`get_sentences()`

`get_sentence_type()`

`get_annotated_sentences()`

built with 

igannotator.backend

[View Source](#)

```
def get_sentences(  
    txt: str,  
    sentence_split_type: str = 'rule_based',  
    atomize: bool = True,  
    verbose: bool = False  
) -> List[str]:
```

[View Source](#)

Split text document into sentences :param txt: Text document :param sentence_split_type: ml or rule_based - use Spacy or regular expression to split into sentences :param atomize: whether try to split enumerations into atomic sentences :param verbose: whether create additional output while processing :return: list of sentences

```
def get_sentence_type(sentences: List[str]) -> pandas.core.frame.DataFrame:
```

[View Source](#)

Assign sentence type :param sentences: List of processed and clean sentences :return: DataFrame with sentences and their corresponding type (c or r)

```
def get_annotated_sentences(  
    sentences: pandas.core.frame.DataFrame  
) -> Tuple[List[int], List[Tuple[igannotator.annotator.lexical_tree.LexicalTreeNode,  
    List[igannotator.rulesexecutor.rules.IGTag]]], List[str]]:
```

[View Source](#)

:param sentences: DataFrame of sentences with sentence type (r/c) - output of get_sentence_type :return: Tuple with tagged tokens and used layers