Activités du Lundi 11 février au Vendredi 15 février :

Ce qui était à faire :

- Un nouveau csv pour les timex + leurs attributs (dans dataframe.csv)
- Un nouveau csv pour les signaux + leurs attributs (dans dataframe.csv)
- Récupérer pour chaque event/timex son contexte (4 mots pré event-timex / 4 mots post event-timex)
    - Les ajouter aux csv existants
- Récupérer pour chaque event/timex son contexte (4 pos pré event-timex / 4 pos post event-timex)
    - Les ajouter aux csv existants
- Récupérer les id de phrases, les id de mots, les id d'events, les id des timex et les id des signaux
    - Les ajouter aux csv existants
- Trouver les synonymes des events pour les trouver dans verbocean
- Un nouveau csv pour les relations
- Prendre en compte le POS tagging (première lettre) pour le lemmatiseur de NLTK (lemmatizer.lemmatize(event, pos='v')
- Effectuer les mêmes traitements sur les fichiers de AQUAINT
- Vérifier dans quels documents nous pouvons trouver des events avec plusieurs makeinstance
- Adapter les dictionnaires en dataframes

Fait :

Au départ, pour obtenir les attributs des events / makeinstance / timex / signaux j'avais utilisé des dictionnaires de dictionnaires (comme écrit dans le résumé du 1er au 8 février).

J'ai eu des difficultés pour écrire toutes les informations dans différents fichiers csv. Anaïs m'a proposé de passer sur des **dataframes** en utilisant la librairie **Pandas** pour simplifier le traitement.

J'ai adapté les dictionnaires que j'avais obtenus suite à l'extraction des attributs des events / makeinstance / timex / signaux en dataframes.

L'idée est de faire un grand dataframe qui ressemblerait à ça pour l'énoncé suivant :

« *Jean sort le mardi. Marc travaille le jeudi.* »

indexPandas	docId	tokenId	SentId	eventId	timexId	ClassEvent	TypeTimex	Eiid…
0	ABC	1	1	NaN	NaN	NaN	NaN	…
1	ABC	2	1	1	NaN	OCC	NaN	…
2	ABC	3	1	NaN	NaN	NaN	NaN	…
3	ABC	4	1	NaN	1	NaN	DATE	…
4	ABC	5	2	NaN	NaN	NaN	NaN	…
5	ABC	6	2	2	NaN	I_ACTION	NaN	…
6	ABC	7	2	NaN	NaN	NaN	NaN	…
7	ABC	8	2	NaN	2	NaN	DATE	…

Nous avons donc 4 dataframes (un pour la récupération des events, un pour les timex, un pour les signaux, un pour les makeinstances + leurs attributs), qui ont ensuite été concaténés à l'horizontale puis introduits dans un fichier csv.

concatenation = pd.concat([df1, df2, df3, df4], axis=1, sort=False)

A faire pour la semaine du 18 février :

- Vérifier dans quels documents nous pouvons trouver des events avec plusieurs makeinstance
- Récupérer pour chaque event/timex son contexte (4 pos d'avant / 4 pos d'après)
- Les ajouter dans le dataframe
- Ajouter les 4 mots d'avant et 4 mots d'après dans le dataframe
- Prendre en compte le POS tagging (première lettre) pour le lemmatiseur de NLTK (lemmatizer.lemmatize(event, pos='v')
- Trouver les synonymes des events pour les trouver dans verbocean
- Intégrer les identifiants de mots / event / timex / signal / phrase dans le dataframe