Mode d’emploi fonctions des modules Round, Combination, HandCombination

Une HandCombination est un objet constitué d’une liste qui contient : Une combinaison, les cartes les plus hautes qui complètent la combinaison

Ex : [Combinaison paire de 2, Roi de cœur, Dame de pique, 9 de pique]

On a bien les 5 cartes les plus avantageuses pour sa main.

Une Combination est un objet qui a pour type "pair", "full\_house" ou autre combinaison …

Le high va etre un objet Card qui décrit la hauteur de la combinaison.

Par ex si type = "straight" et high = Card("spade", "5") on a une quinte hauteur 5

La valeur second doit etre précisée seulement pour les two\_pairs et les full\_house. Pour les two pair ça represente la paire la plus basse, pour les full house ça représente la paire.

La liste cards ne doit etre précisée que lors du cas d’un full\_house, elle permet de comparer 2 full houses qui auraient une carte haute similaire.

Il y a 10 fonctions qui vérifient chacune la présence d’une combinaison dans l’addition d’un board et d’une main

Is\_royal\_flush(board,hand)

Is\_straight\_flush(board,hand)

Is\_four\_of\_a\_kind(board,hand)

…

Ces fonctions renvoient soit False si la combinaison n’est pas présente dans l’addition du board et de la hand, si c’est le cas elles renvoient un tuple , (True, HandCombination)

La fonction : get\_best\_combination(board, hand) renvoie la meilleure HandCombination possible avec les valeurs entrées.

Les méthodes \_\_lt\_\_ \_\_mt\_\_ et \_\_eq\_\_ ont été redéfinies sur les Card, Combination, HandCombination ce qui permet d’utiliser les opérateurs > < == sur ces objets afin de comparer leurs valeurs.

Si vous manquez de précisions passez discord je réécrirai sur ce papier :3