



Le Black Jack

I- Présentation

Il s'agit d'un jeu de carte célèbre pratiqué dans les casinos du monde entier. Le Black Jack se joue avec un jeu de 52 cartes.

Un joueur et le croupier s'affrontent. Ils disposent chacun de la même somme d'argent en début de partie :

- Une carte, visible de tous, est distribuée à chacun.
- Le joueur indique alors sa mise.
- L'objectif est de se rapprocher le plus possible de 21, sans dépasser ce score.
- Le joueur demande une carte, puis une autre, etc.
- S'il dépasse 21, il perd sa mise qui va au croupier. Sinon, il s'arrête dès qu'il est satisfait de son score.
- Le croupier essaie alors de réaliser un score supérieur ou égal à celui du joueur. Il doit s'arrêter lorsqu'il a de 17 à 21 et tirer une carte à 16 ou moins.
- S'il y arrive, il empoche la mise du joueur. Sinon, Si le joueur atteint le Blackjack (soit une carte valant 10 + un As) sa mise est payée 3 pour 2, s'il gagne contre le Croupier, mais sans atteindre 21 points, il remporte 1 fois sa mise.

Le jeu se poursuit jusqu'à ce que l'un d'eux n'ait plus d'argent.

Le **casino** a l'avantage au **Blackjack**. Le principe est que les joueurs reçoivent des cartes avant le croupier et ils perdront en premier, même si le croupier crève aussi sa main dans la même ronde.

II- L'interface graphique

On vous fournit une interface graphique à travers la classe GUIblackjack que vous pouvez importer depuis le module fourni. Si le répertoire GUI_blackjack est dans le même répertoire que votre fichier :

```
from GUI_balckjack.guiBlackJack import GUIblackjack
```

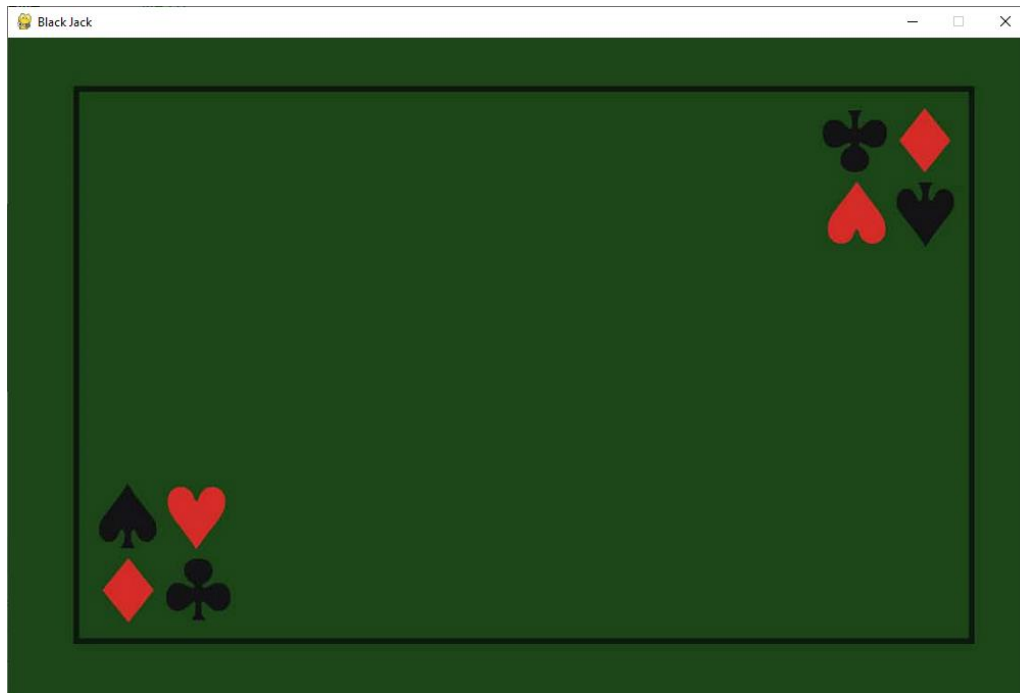
Constructeur de la classe :

- Attribut public : **aucun**.

- Le constructeur ne prend aucun paramètre. Il permet de créer un objet graphique vierge.

Par exemple :

GUI = GUIblackjack()



- Les méthodes :
 - **refresh(j1, j2, t = "", cacheCroupier = False)** : Cette méthode rafraichie l'affichage des deux listes de cartes passée en argument.
 - **j1** et **j2** sont des objets itérables, dont les éléments sont de la forme (n, c). n est un entier de 1 à 13 représentant la valeur faciale de la carte (11, 12, et 13 étant respectivement le valet, la dame et le roi). C'est un caractère unicode parmi '♥', '♦', '♣', '♠' représentant la couleur de la carte. D'autre part j1 et j2 doivent disposer d'une méthode `__len__` pour répondre à la fonction `len()`. Il peut donc s'agir de listes, de tuples ou de tout autres objets répondants à ces critères.
 - **cacheCroupier** est un booléen. Si cet argument est True, toutes les cartes du croupier seront affichées retournées sauf la première.
 - **t** est un texte à afficher à destination du joueur.

Par exemple :

`GUI.refresh([(5, '♥'), (10, '♦'), (13, '♣')], [(13, '♠'), (1, '♥')], "Voulez-vous recommencer ?")`



- **messCentre(mess)** : Affiche le message mess en gros, en plein milieu de l'écran. Cette méthode est principalement destinée à afficher la victoire ou la défaite du joueur.

Par exemple :

`GUI.messCentre("GAGNE !!!")`



- **affJetons(nj0, nj1, njp)** : Cette méthode permet d'afficher 3 jetons. Un pour le croupier, un pour le joueur et un au centre pour le pari.

La valeur de chaque pile de jetons : respectivement nj0, nj1, njp est inscrite dessus. Si un nombre de jetons est à zéro : la pile n'apparaît pas.

Par exemple :

GUI.refresh([(5, '♥'), (10, '♦'), (13, '♣')], [(13, '♠'), (1, '♥')], "Avec des jetons ...")

GUI.affJetons(56, 115, 5)



- **waitClick()** : Cette méthode attend l'action d'un joueur. Elle gère trois types d'actions :
 - **demande fermeture de la fenêtre** : fermeture propre de la fenêtre pygame et fin du programme python.
 - **click sur la fenêtre** :
 - click sur les jetons du croupier : renvoi 'j0'
 - click sur les jetons du joueur : renvoi 'j1'
 - click sur les jetons au centre : renvoi 'jp'
 - click sur les cartes du croupier : renvoi 'c0'
 - click sur les cartes du joueur : renvoi 'c1'
 - **appui sur des touches spéciales** :
 - fleche RIGHT : retourne '_R';
 - fleche LEFT : retourne '_L';
 - fleche DOWN : retourne '_D';
 - fleche UP : retourne '_U';
 - touche BACKSPACE : retourne '_B';

- touche RETURN : retourne '_E';
- touche ESCAPE : retourne '_S';
- **appui sur une autre touche du clavier** : retourne le caractère unicode correspondant.

Attention, une fois exécutée, on ne peut sortir de cette méthode que par l'une de ces trois actions.