

Calculateur de scores pour le tarot

Essai de réalisation d'un calculateur de score pour le tarot. Ecris en python 3. Toujours en développement (reste à encoder la partie à 5 joueurs, opérationnel pour 3 et 4 joueurs à ce jour).

Plus de docs et de détails par la suite..

Pour bénéficier de la dernière version à jour, merci de vérifier sur le git : [tarot-v3.py sur le git](#)

```
# Gestion du nombre de joueurs
nb_joueurs = -1
while True:
    try:
        print('Entre 3 et 5 joueurs, svp.')
        nb_joueurs = int(input("Saisir le nombre de joueurs : "))
        if 2 < nb_joueurs < 6:
            print("Nombre de joueurs : ", nb_joueurs)
            break
    except ValueError:
        print("Oops! Réponse incorrecte, ce n'est pas un nombre... Réessayer...")
# =====

# Creation d'un dictionnaire de joueurs
dico_joueurs = dict()

for i in range(0, nb_joueurs):
    joueur = input("Nom du joueur {} : ".format(i + 1))
    print(joueur)

    # Initialisation des scores
    # Tout le monde à zéro
    dico_joueurs[joueur] = [0]

for nom, point in dico_joueurs.items():
    print("Le joueur {} débute avec {} point.".format(nom, point[0]))
# =====

# Creation d'un dictionnaire de contrat
dico_contrat = dict()
dico_contrat['petite'] = 25
dico_contrat['garde'] = 25 * 2
dico_contrat['garde sans '] = 25 * 3
dico_contrat['garde contre'] = 25 * 5

# Gestion des oudlers
```

```
# Création d'un dictionnaire d'oudlers
dico_oudlers = dict()
dico_oudlers[0] = 56
dico_oudlers[1] = 51
dico_oudlers[2] = 41
dico_oudlers[3] = 36

# Gestion des primes
# Création d'un dictionnaire d'oudlers
dico_primes = dict()
dico_primes['petit au bout'] = 10
dico_primes['simple poignée'] = 20
dico_primes['double poignée'] = 30
dico_primes['triple poignée'] = 40
dico_primes['chelem annoncé réalisé'] = 400
dico_primes['chelem annoncé non réalisé'] = -200
dico_primes['chelem non annoncé mais réalisé'] = 200

#
=====
=====

# Les fonctions
#
=====
=====

def verificationContrat(nombre_de_points, nombre_d_oudlers):
    # Contrat rempli ou non ? Calcul du bonus / malus
    if nombre_de_points >= nombre_d_oudlers:
        print("Le contrat est rempli.")
    else:
        print("Le contrat n'est pas rempli.")

    gain = nombre_de_points - nombre_d_oudlers

    print("{} points remportés dans ce tour.".format(gain))
    return gain

# =====

def updateScore(gain, dictionnaire_joueurs, preneur, nb_joueurs):
    # Mise à jour du score
    # Ajout des scores

    for nom in dictionnaire_joueurs:
        score = dictionnaire_joueurs[nom][-1]

        if nom == preneur:
            score = score + (gain + dico_contrat[contrat_tour]) *
```

```
(nb_joueurs - 1)
    else:
        score = score - (gain + dico_contrat[contrat_tour])
        dictionnaire_joueurs[nom].append(score)

print(dictionnaire_joueurs)

# Initialisation des paramètres du tour
def parametrageTour(dictionnaire_joueurs,
                    dictionnaire_contrats,
                    dictionnaire_oudlers,
                    dictionnaire_primes,
                    nombre_de_points_realises
                    ):
    # Gestion du preneur
    preneur = ""
    while True:
        try:
            preneur = input("Preneur : ")
            if preneur in dico_joueurs:
                print("Le preneur est {}".format(preneur))
                break
            else:
                print("Ce joueur n'existe pas. Réessayer.")
        except ValueError:
            print("Erreur...")

    # =====

    # Gestion du contrat
    contrat_tour = ""
    while True:
        try:
            contrat_tour = input("Contrat : ")
            contrat_tour = contrat_tour.lower()
            if contrat_tour in dico_contrat:
                print("Le contrat est {}".format(contrat_tour))
                break
            else:
                print("Ce contrat n'existe pas. Réessayer.")
        except ValueError:
            print("Erreur...")

    # =====

    # Gestion du nombre d'oudlers
    nb_oudlers = ""
    while True:
        try:
            print('Nombre d\'oudler(s) : 0, 1, 2 ou 3')
            nb_oudlers = int(input("Saisir le nombre d'oudler(s) : "))
            if nb_oudlers in dico_oudlers:
```

```

        print(
            "Nombre d'oudler est {}. Il faut réaliser {}
points.".format(nb_oudlers, dico_oudlers[nb_oudlers]))
        break
    except ValueError:
        print("Oops! Réponse incorrecte, ce n'est pas un nombre...
Réessayer...")
# =====

# Gestion du nombre de points
while True:
    try:
        nb_points = int(input("Saisir le nombre de points réalisés : "))
        if 0 < nb_points < 92:
            print("Nombre de points réalisés : ", nb_points)
            break
    except ValueError:
        print("Oops! Réponse incorrecte, ce n'est pas un nombre...
Réessayer...")
# =====

# Gestion des primes
while True:
    try:
        primes = input(" Y a-t-il des primes (oui/o ou non/n ) ? : ")
        if primes.lower() in ['oui', 'o']:
            print("des primes cool")

        # Gestion des primes
        fin_des_primes = 'oui'
        beneficiaire = ''
        liste_des_primes = []

        while fin_des_primes.lower() not in ['non', 'n']:

            while True:
                try:
                    beneficiaire = input("Bénéficiaire de la prime :
")

                    if beneficiaire in dico_joueurs:
                        print("Le beneficiaire est
{}".format(beneficiaire))

                        break
                    else:
                        print("Ce joueur n'existe pas. Réessayer.")
                        print('Rappel. Voici la liste des joueurs :
')

                        i = 0
                        for nom in dico_joueurs:
                            print(" --- Joueur {} : {}".format(i +
1, nom))

```

```

        i += 1
    except ValueError:
        print("Erreur...")

    while True:
        try:
            prime = input("Quelle prime attribuer à {} ? :
".format(beneficiaire))

            if prime in dictionnaire_primes:
                print("{} accordée à {}".format(prime,
beneficiaire))
                break
            else:
                print("Cette prime n'existe pas.
Réessayer.")
                print('Rappel. Voici la liste des primes :
')
                for nom, valeur in
dictionnaire_primes.items():
                    print("Prime : {}, valeur :
{}".format(nom, valeur))
        except ValueError:
            print("Erreur...")

        liste_des_primes.append({beneficiaire, prime})
        fin_des_primes = input("Y a-t-il des autres primes à
saisir ? ")

        else:
            print('Les primes ont été enregistrées.')
            break

        elif primes.lower() in ['non', 'n']:
            print('Pas de primes à prendre en compte.')
            break
    except ValueError:
        print("Oops! Réponse incorrecte. Saisir : (oui/o ou non/n )
... Réessayer...")
    # =====

    return nb_points, nb_oudlers, contrat_tour, preneur, liste_des_primes

#
=====
=====
# Lancement du jeu
#
=====
=====

```

```
fin_du_jeu = ''
while fin_du_jeu.lower() != 'quitter':
    nb_points, nb_oudlers, contrat_tour, preneur, liste_des_primes =
parametrageTour(dico_joueurs,
dico_contrat,
dico_oudlers,
dico_primes,
nb_joueurs)
    gain = verificationContrat(nb_points, dico_oudlers[nb_oudlers])
    updateScore(gain, dico_joueurs, preneur, nb_joueurs)
    fin_du_jeu = input('Cesser la partie ? Saisir "Quitter" : ')

else:
    print('on stoppe la partie.')
```

From: <https://cbiot.fr/dokuwiki/> - **Cyrille BIOT**

Permanent link: <https://cbiot.fr/dokuwiki/python:tarot?rev=1578238375>

Last update: **2020/01/05 15:32**

