

```
import random
```

```
# 1 jeu a un joueur
```

```
def genere_secret():  
    rep = []  
    for i in range(4):  
        rep.append(random.randint(0, 5))  
    return rep
```

```
def lire_prop():  
    prop = "  
    p = "  
    while prop == "":  
        prop = input('Votre Proposition ? (au format xxxx)')  
        p = conversion(prop)  
        if valide_mm(p) == False:  
            prop = "  
    return p
```

```
def conversion(prop):  
    liste = []  
    for i in range(len(prop)):  
        liste.append(int(prop[i]))  
    return liste
```

```
def valide_mm(prop):  
    if len(prop) != 4:  
        print("je n'ai pas compris.")  
        print("Il faut 4 chiffres dans l'intervalle[0,5]")  
        return False  
    for i in range(len(prop)):  
        if prop[i] > 5:  
            print("je n'ai pas compris.")  
            print("Il faut 4 chiffres dans l'intervalle[0,5]")  
            return False  
    return True
```

```
def bp_mp(prop, code):  
    bp = bien_place(prop, code)  
    mp = valeur_commune(prop, code)  
    return(bp, (mp-bp))
```

```
def bien_place(prop, code):  
    bp = 0  
    for i in range(len(code)):
```

```

    if prop[i] == code[i]:
        bp += 1
    return bp

```

```

def distribution(prop):
    liste = [0, 0, 0, 0, 0, 0]
    for i in range(len(prop)):
        liste[prop[i]] += 1
    return liste

```

```

def valeur_commune(prop, code):
    p = distribution(prop)
    c = distribution(code)
    mp = 0
    for i in range(len(c)):
        if p[i] == c[i] and c[i] != 0:
            mp += p[i]
        if c[i] < p[i] and c[i] != 0:
            mp += c[i]
    return mp

```

```

def affiche_reponse(bp, mp):
    print('Vous avez {} bien placé et {} mal placés.'.format(bp, mp))

```

```

def main1_joueur():
    nb_coups = 0
    secret = genere_secret()
    prop = ""
    while prop != secret:
        nb_coups += 1
        prop = lire_prop()
        efrf = bp_mp(prop, secret)
        affiche_reponse(efrf[0], efrf[1])
    if prop == secret:
        print('Bravo ! vous avez trouvé en {} coups.'.format(nb_coups))

```

2 joueur

```

def main2_joueur():
    historique = []
    nb_coups = 0
    secret_1 = genere_secret()
    secret_2 = genere_secret()
    prop_1 = ""
    prop_2 = []
    print('Jeu du Master-mind')
    print('je viens de générer 2 code secret à 4 chiffres')
    print('à vous ! les chiffres possible vont de [0 à 5]')
    while prop_1 != secret_1 and prop_2 != secret_2:

```

```

prop_2 = []
nb_coups += 1
prop_2 = genere_secret()
historique += prop_2
efrf = bp_mp(prop_2, secret_2)
print('mastermind propose: {}'.format(prop_2))
affiche_reponse(efrf[0], efrf[1])
if prop_2 == secret_2:
    print('il a trouvé en {} coups.'.format(nb_coups))

prop_1 = lire_prop()
efr = bp_mp(prop_1, secret_1)
affiche_reponse(efr[0], efr[1])
if prop_1 == secret_1:
    print('Bravo ! vous avez trouvé en {} coups.'.format(nb_coups))

```

3 fonction principale

```

def main():
    choix = ""
    print('=====MASTER-MIND=====')
    choix = input('Choisir le mode de jeux: 1(un seul joueur) ou 2 (2 joueur): ')
    while choix != '1' and choix != '2':
        choix = input('mettre 1 ou 2: ')
    if choix == '1':
        print('=====MODE AVEC 1 JOUEUR=====')
        main1_joueur()
    else:
        print('=====MODE AVEC 2 JOUEUR=====')
        main2_joueur()

```