

```

1 from PyQt5.uic import loadUi
2 from PyQt5.QtWidgets import QApplication
3 #*****
4 from random import randint
5 from numpy import array
6 T=array([int()]*8)
7 def REMP_AFFICHE(T):
8     for i in range(7):
9         T[i]=randint(1,99)
10        print(T[i], end=" | ")
11    print()
12 #*****
13 def GainPerd(N):
14     s=Somme(N)
15     while len(str(s)) !=1:
16         s=Somme(str(s))
17     return Existence(s*s,T)
18 #*****
19 def Somme(x):
20     som=0
21     for i in range(len(x)):
22         som+=int(x[i])
23     return som
24 #*****
25 def Existence(nb,T):
26     i=-1
27     while i < 8 and T[i] != nb:
28         i+=1
29     return i < 8
30 #*****
31 def Play():
32     N=fen.a.text()
33     if not ( N.isdecimal() and ( len(N) ==8 ) and (N[0] in {"2","3","4","5","9"}) ):
34         fen.res.setText("Désolé ! vérifier le numéro de téléphone du client !!")
35     elif GainPerd(N):
36         fen.res.setText("Le client ayant le numéro de téléphone "+N+" \n a gagné un chariot
gratuit")
37     else:
38         fen.res.setText("Le client ayant le numéro de téléphone "+N+" \n doit payer les achats du
chariot")
39
40 #*****
41 def efface():
42     fen.a.clear()
43     fen.res.clear()
44 #*****
45 REMP_AFFICHE(T)
46 app=QApplication([])
47 fen=loadUi("InterCHARIOT.ui")
48 fen.show()
49 fen.b1.clicked.connect(Play)
50 fen.b2.clicked.connect(efface)
51 app.exec_()

```

```

1 from PyQt5.uic import loadUi
2 from PyQt5.QtWidgets import QApplication
3 #*****
4 from random import randint
5 def Generer():
6     ch=""
7     for i in range(26):
8         n=randint(65,90)
9         while ch.find(chr(n)) != -1 or n == i:
10             n=randint(65,90)
11         ch+=chr(n)
12     return ch
13 #*****
14 def Crypter(m,c):
15     ch=""
16     for i in range(len(m)):
17         if "A" <= m[i] <= "Z":
18             ch+=c[ord(m[i])-65]
19         else:
20             ch+=m[i]
21     return ch
22
23 #*****
24 def Majuscule_Espace(ch):
25     i=0
26     while i < len(ch) and ("A" <= ch[i] <= "Z" or ch[i]==" "):
27         i+=1
28     return i==len(ch)
29 #*****
30 def Play():
31     m=fen.a.text()
32     if not ( 1 <= len(m) <= 100 and Majuscule_Espace(m) ):
33         fen.res.setText("Désolé ! vérifier le message à crypter !!")
34     else:
35         cle=Generer()
36         MC=Crypter(m,cle)
37         fen.cle.setText("la clé générée est: " + cle)
38         fen.res.setText("Le message crypté sera: " + MC)
39 #*****
40 def efface():
41     fen.a.clear()
42     fen.cle.clear()
43     fen.res.clear()
44 #*****
45 app=QApplication([])
46 fen=loadUi("InterCRYPTE_AC.ui")
47 fen.show()
48 fen.b1.clicked.connect(Play)
49 fen.b2.clicked.connect(efface)
50 app.exec_()

```

```

1  from PyQt5.uic import loadUi
2  from PyQt5.QtWidgets import QApplication
3  #*****
4  def Crypter(ch):
5      ch_crypte=""
6      nb=1
7      for i in range(len(ch)-1):
8          if ch[i]==ch[i+1]:
9              nb+=1
10             else:
11                 ch_crypte+=str(nb)+ch[i]
12                 nb=1
13         ch_crypte+=str(nb)+ch[len(ch)-1]
14     return ch_crypte
15
16 #*****
17 def Alphanetique(ch):
18     i=0
19     while i < len(ch) and ("A" <= ch[i].upper() <= "Z"):
20         i+=1
21     return i==len(ch)
22 #*****
23 def Play():
24     CH=fen.a.text()
25     if not ( 1 <= len(CH) <= 50 and Alphanetique(CH) ):
26         fen.res.setText("Désolé ! vérifier la chaîne à crypter !!")
27     else:
28         CHC=Crypter(CH)
29         fen.res.setText("La chaîne cryptée sera: " + CHC)
30 #*****
31 def efface():
32     fen.a.clear()
33     fen.res.clear()
34 #*****
35 app=QApplication([])
36 fen=loadUi("InterCRYPTTE_OS.ui")
37 fen.show()
38 fen.b1.clicked.connect(Play)
39 fen.b2.clicked.connect(efface)
40 app.exec_()

```



```
1 from PyQt5.uic import loadUi
2 from PyQt5.QtWidgets import QApplication
3 #####
4 def occurrence(c,ch):
5     nb=0
6     for i in range(len(ch)):
7         if ch[i]==c:
8             nb+=1
9     return nb
10 #####
11 def Crypter(ch):
12     m=""
13     for i in range(len(ch)):
14         n=occurrence(ch[i],ch)
15         if n % 2==0:
16             k= n // 2
17         else:
18             k=2*n
19         m+=chr ( (ord(ch[i]) - ord ("A") + k ) % 26 + ord ("A") )
20     return m
21 #####
22 def Majuscule(ch):
23     i=0
24     while i < len(ch) and ("A" <= ch[i] <= "Z"):
25         i+=1
26     return i==len(ch)
27 #####
28 def Play():
29     MS=fen.a.text()
30     if not ( 1 <= len(MS) and Majuscule(MS) ):
31         fen.res.setText("Désolé ! vérifiez le mot à crypter !!")
32     else:
33         fen.res.setText("Le mot crypté sera: " + Crypter(MS))
34 #####
35 def efface():
36     fen.a.clear()
37     fen.res.clear()
38 #####
39 app=QApplication([])
40 fen=loadUi("InterCRYPTE_AM.ui")
41 fen.show()
42 fen.b1.clicked.connect(Play)
43 fen.b2.clicked.connect(efface)
44 app.exec_()
```