```
RESULTAT DES TESTS

============================= test session starts ==============================
platform darwin -- Python 3.6.2, pytest-5.2.2, py-1.8.0, pluggy-0.13.0
rootdir: /Users/emiliendurif/Dropbox/cpge/ipt_mpsi_ds/DS08/correction_auto/scrip
t/DS_08_besson
plugins: timeout-1.3.4
timeout: 60.0s
timeout method: signal
timeout func_only: False
collected 22 items

test_TP.py .........FFF.F.FFFFFFF                                        [100%]

=================================== FAILURES ===================================
_____________________________________ test_Q8_req _____________________________________

    def test_Q8_req ():
>       assert requete(sol_Q8_req) == requete(Q8_req)

test_TP.py:125:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

req = 'SELECT CLI_NOM from T_CLIENT group by having count(CLI_NOM)>1'

    def requete (req):
        conn = sqlite3.connect('../../bdd/hotel_'+convert(alpha)+'.db')
        c=conn.cursor()
>       c.execute(req)
E       sqlite3.OperationalError: near "having": syntax error

test_TP.py:25: OperationalError
_____________________________________ test_Q9_req _____________________________________

    def test_Q9_req ():
>       assert requete(sol_Q9_req) == requete(Q9_req)

test_TP.py:128:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

req = 'SELECT CLI_NOM,count(CLI_NOM) from T_CLIENT group by having count(CLI_NOM
)>1'

    def requete (req):
        conn = sqlite3.connect('../../bdd/hotel_'+convert(alpha)+'.db')
        c=conn.cursor()
>       c.execute(req)
E       sqlite3.OperationalError: near "having": syntax error

test_TP.py:25: OperationalError
_____________________________________ test_Q10_res _____________________________________

    def test_Q10_res ():
>       assert sol_Q10_res.upper() ==  Q10_res.upper()
E       AssertionError: assert '(51.0, 85.0)' == 'MOYENNE POUR... REMISE: 85.0'
E         - (51.0, 85.0)
E         + MOYENNE POURCENT: 51.0 ET MOYENNE REMISE: 85.0

test_TP.py:131: AssertionError
_____________________________________ test_Q11_res _____________________________________

    def test_Q11_res ():
>       assert sol_Q11_res.upper() ==  Q11_res.upper()
E       AssertionError: assert '(51.0, 85)' == 'MAX POURECNT...AX REMISE: 85'
E         - (51.0, 85)
E         + MAX POURECNTAGE: 51 ET MAX REMISE: 85

test_TP.py:137: AssertionError
```

```
_____________________________________ test_Q12_req _____________________________________

    def test_Q12_req ():
>       assert requete(sol_Q12_req) == requete(Q12_req)
E       assert [(1,), (2,), ...), (26,), ...] == [(1,), (3,), ...,), (2,), ...]
E         At index 1 diff: (2,) != (3,)
E         Right contains 894 more items, first extra item: (712,)
E         Use -v to get the full diff

test_TP.py:143: AssertionError
_____________________________________ test_Q13_req _____________________________________

    def test_Q13_req ():
>       assert requete(sol_Q13_req) == requete(Q13_req)

test_TP.py:146:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

req = 'SELECT T_facture.CLI_ID from T_FACTURE join T_LIGNE_FACTURE on T_LIGNE_FA
CTURE;FAC_ID=T_FACTURE.FAC_ID WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POURCENT>
0'

    def requete (req):
        conn = sqlite3.connect('../../bdd/hotel_'+convert(alpha)+'.db')
        c=conn.cursor()
>       c.execute(req)
E       sqlite3.OperationalError: no such column: T_LIGNE_FACTURE

test_TP.py:25: OperationalError
_____________________________________ test_Q14_req _____________________________________

    def test_Q14_req ():
>       assert requete(sol_Q14_req) == requete(Q14_req)

test_TP.py:149:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

req = 'SELECT T_FACTURE.CLI_ID FROM T_FACTURE EXCEPT(SELECT T_facture.CLI_ID fro
m T_FACTURE join T_LIGNE_FACTURE on T_LIGNE_FACTURE;FAC_ID=T_FACTURE.FAC_ID WHER
E LIF_REMISE_MONTANT>0 or LIF_REMISE_POURCENT>0) '

    def requete (req):
        conn = sqlite3.connect('../../bdd/hotel_'+convert(alpha)+'.db')
        c=conn.cursor()
>       c.execute(req)
E       sqlite3.OperationalError: near "(": syntax error

test_TP.py:25: OperationalError
_____________________________________ test_Q15_res _____________________________________

    def test_Q15_res ():
>       assert sol_Q15_res.upper() ==  Q15_res.upper()
E       assert "('SILLET', 'JACQUES', 935)" == 'NOM,PRENOM,MONTANT'
E         - ('SILLET', 'JACQUES', 935)
E         + NOM,PRENOM,MONTANT

test_TP.py:152: AssertionError
_____________________________________ test_Q15_req _____________________________________

    def test_Q15_req ():
>       assert requete(sol_Q15_req) == requete(Q15_req)
E       AssertionError: assert [('SILLET', 'Jacques', 935)] == []
E         Left contains one more item: ('SILLET', 'Jacques', 935)
E         Use -v to get the full diff

test_TP.py:155: AssertionError
_____________________________________ test_Q15_res2 _____________________________________
```

```
    def test_Q15_res2 ():
>       assert sol_Q15_res2.upper() ==  Q15_res.upper()
E       assert "('GARREAU', 'PAUL', 22)" == 'NOM,PRENOM,MONTANT'
E         - ('GARREAU', 'PAUL', 22)
E         + NOM,PRENOM,MONTANT

test_TP.py:158: AssertionError
_____________________________________ test_Q15_req2 _____________________________________

    def test_Q15_req2 ():
>       assert requete(sol_Q15_req2) == requete(Q15_req)
E       AssertionError: assert [('GARREAU', 'Paul', 22)] == []
E         Left contains one more item: ('GARREAU', 'Paul', 22)
E         Use -v to get the full diff

test_TP.py:161: AssertionError
======================== 11 failed, 11 passed in 0.38s =========================

MODIFICATIONS EFFECTUEES SUR LE FICHIER RENDU
=============================================

--- prog.py.orig        2020-06-22 08:23:31.000000000 +0200
+++ prog.py     2020-06-24 14:55:47.000000000 +0200
@@ -4,8 +4,7 @@
 alpha="36"

 ## Question 1
-Q1_req = "SELECT CLI_NOM, CLI_PRENOM, TIT_CODE
-            FROM T_CLIENT"
+Q1_req = "SELECT CLI_NOM, CLI_PRENOM, TIT_CODE FROM T_CLIENT"

 ## Question 2
 Q2_req = "SELECT COUNT(*) FROM T_CLIENT"
@@ -52,8 +51,7 @@
 Q13_req = "SELECT T_facture.CLI_ID from T_FACTURE join T_LIGNE_FACTURE on T_LIG
NE_FACTURE;FAC_ID=T_FACTURE.FAC_ID WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POUR
CENT>0"

 ## Question 14
-Q14_req = "SELECT T_FACTURE.CLI_ID FROM T_FACTURE
-EXCEPT(SELECT T_facture.CLI_ID from T_FACTURE join T_LIGNE_FACTURE on T_LIGNE_F
ACTURE;FAC_ID=T_FACTURE.FAC_ID WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POURCENT
>0) "
+Q14_req = "SELECT T_FACTURE.CLI_ID FROM T_FACTURE EXCEPT(SELECT T_facture.CLI_I
D from T_FACTURE join T_LIGNE_FACTURE on T_LIGNE_FACTURE;FAC_ID=T_FACTURE.FAC_ID
 WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POURCENT>0) "


 ## Question 15
```

```
NOM = "BESSON"
Prenom = "Mathis"
Classe = "MPSI2"
alpha="36"

## Question 1
Q1_req = "SELECT CLI_NOM, CLI_PRENOM, TIT_CODE
            FROM T_CLIENT"

## Question 2
Q2_req = "SELECT COUNT(*) FROM T_CLIENT"
Q2_res = "91"

## Question 3
Q3_req = "SELECT CLI_NOM, CLI_PRENOM FROM T_CLIENT WHERE TIT_CODE='Mme.'"

## Question 4
Q4_req = "SELECT CLI_NOM, CLI_PRENOM, TIT_CODE FROM T_CLIENT where TIT_CODE='Mme
.' or TIT_CODE='Melle.'"


## Question 5
Q5_req = "SELECT count(*) FROM T_CLIENT where TIT_CODE='Mme.' or TIT_CODE='Melle
.'"
Q5_res = "15"

## Question 6
Q6_req = "SELECT CLI_NOM as Noms, CLI_PRENOM as Prénoms FROM T_CLIENT where TIT_
CODE='Mme.' or TIT_CODE='Melle.' ORDER BY Noms"


## Question 7
Q7_req = "SELECT T_CLIENT.CLI_NOM as Noms, T_TELEPHONE.TEL_NUMERO from T_CLIENT
join T_TELEPHONE on T_TELEPHONE.CLI_ID=T_CLIENT.CLI_ID WHERE T_TELEPHONE.TYP_COD
E='TEL'"

## Question 8
Q8_req = "SELECT CLI_NOM from T_CLIENT group by having count(CLI_NOM)>1"

## Question 9
Q9_req = "SELECT CLI_NOM,count(CLI_NOM) from T_CLIENT group by having count(CLI_
NOM)>1"
Q9_res = "BENATTAR:2 et MARTIN:2"

## Question 10
Q10_req = "SELECT avg(LIF_remise_pourcent) as pourcent_moy, avg(LIF_remise_monta
nt) as remise_moy from T_LIGNE_FACTURE"
Q10_res = "moyenne pourcent: 51.0 et moyenne remise: 85.0"

## Question 11
Q11_req = "SELECT max(LIF_remise_pourcent),max(LIF_remise_montant) FROM T_LIGNE_
FACTURE"
Q11_res = "max pourecntage: 51 et max remise: 85"

## Question 12
Q12_req = "SELECT FAC_ID as fact_id1 from T_LIGNE_FACTURE WHERE LIF_REMISE_MONTA
NT>0 or LIF_REMISE_POURCENT>0"


## Question 13
Q13_req = "SELECT T_facture.CLI_ID from T_FACTURE join T_LIGNE_FACTURE on T_LIGN
E_FACTURE;FAC_ID=T_FACTURE.FAC_ID WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POURC
ENT>0"

## Question 14
Q14_req = "SELECT T_FACTURE.CLI_ID FROM T_FACTURE
EXCEPT(SELECT T_facture.CLI_ID from T_FACTURE join T_LIGNE_FACTURE on T_LIGNE_FA
CTURE;FAC_ID=T_FACTURE.FAC_ID WHERE LIF_REMISE_MONTANT>0 or LIF_REMISE_POURCENT>
```

```
0) "


## Question 15
Q15_req = ""
Q15_res ="nom,prenom,montant"
```