

## 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_test/DS_08_abid
plugins: timeout-1.3.4
timeout: 60.0s
timeout method: signal
timeout func_only: False
collected 20 items
```

```
test_TP.py F..FF...FF.FFFFF.FFF [100%]
```

```
===== FAILURES =====
_____ test_Q1_req _____
```

```
def test_Q1_req ():
>     assert requete(sol_Q1_req) == requete(Q1_req)
E       AssertionError: assert [('DUPONT', '...', 'M.'), ...] == [('M.', 'DUPO..
.'Alain'), ...]
E         At index 0 diff: ('DUPONT', 'Alain', 'M.') != ('M.', 'DUPONT', 'Alain'
)
E         Use -v to get the full diff
```

```
test_TP.py:92: AssertionError
```

```
_____ test_Q3_req _____
```

```
def test_Q3_req ():
>     assert requete(sol_Q3_req) == requete(Q3_req)
E       AssertionError: assert [('BOYER', 'M... 'Mme.'), ...] == [('BOYER', 'M..
.ueline'), ...]
E         At index 0 diff: ('BOYER', 'Martine', 'Mme.') != ('BOYER', 'Martine')
E         Use -v to get the full diff
```

```
test_TP.py:101: AssertionError
```

```
_____ test_Q4_req _____
```

```
def test_Q4_req ():
>     assert requete(sol_Q4_req) == requete(Q4_req)
E       AssertionError: assert [('DUHAMEL', ... 'Mme.'), ...] == [('Melle.', '..
.Carmen'), ...]
E         At index 0 diff: ('DUHAMEL', 'Evelyne', 'Melle.') != ('Melle.', 'DUHAM
EL', 'Evelyne')
E         Use -v to get the full diff
```

```
test_TP.py:104: AssertionError
```

```
_____ test_Q7_req _____
```

```
def test_Q7_req ():
>     assert requete(sol_Q7_req) == requete(Q7_req)
E       AssertionError: assert [('DUPONT', '...-52-50'), ...] == [('DUPONT', '..
.-52-50'), ...]
E         At index 0 diff: ('DUPONT', '01-45-42-56-63') != ('DUPONT', 'Alain', '
01-45-42-56-63')
E         Use -v to get the full diff
```

```
test_TP.py:116: AssertionError
```

```
_____ test_Q8_req _____
```

```
def test_Q8_req ():
>     assert requete(sol_Q8_req) == requete(Q8_req)
E       AssertionError: assert [('BENATTAR',), ('MARTIN',)] == [('MARTIN', '...'
, 'Bernard')]
E         At index 0 diff: ('BENATTAR',) != ('MARTIN', 'Marc')
E         Right contains 2 more items, first extra item: ('BENATTAR', 'Pierre')
E         Use -v to get the full diff
```

test\_TP.py:119: AssertionError

test\_Q10\_res

```
def test_Q10_res ():
> assert sol_Q10_res.upper() == Q10_res.upper()
E AssertionError: assert '(81.0, 47.0)' == 'POURCENTAGE ...ONTANT : 81.0'
E - (81.0, 47.0)
E + POURCENTAGE : 47.0 ET MONTANT : 81.0
```

test\_TP.py:125: AssertionError

test\_Q10\_req

```
def test_Q10_req ():
> assert requete(sol_Q10_req) == requete(Q10_req)
E assert [(81.0, 47.0)] == [(47.0, 81.0)]
E At index 0 diff: (81.0, 47.0) != (47.0, 81.0)
E Use -v to get the full diff
```

test\_TP.py:128: AssertionError

test\_Q11\_res

```
def test_Q11_res ():
> assert sol_Q11_res.upper() == Q11_res.upper()
E AssertionError: assert '(81, 47.0)' == 'POURCENTAGE ... MONTANT : 81'
E - (81, 47.0)
E + POURCENTAGE : 47 ET MONTANT : 81
```

test\_TP.py:131: AssertionError

test\_Q11\_req

```
def test_Q11_req ():
> assert requete(sol_Q11_req) == requete(Q11_req)
E assert [(81, 47.0)] == [(47, 81)]
E At index 0 diff: (81, 47.0) != (47, 81)
E Use -v to get the full diff
```

test\_TP.py:134: AssertionError

test\_Q12\_req

```
def test_Q12_req ():
> assert requete(sol_Q12_req) == requete(Q12_req)
E assert [(1,), (3,), ...), (2,), ...] == [(1,), (2,), ...), (26,), ...]
E At index 1 diff: (3,) != (2,)
E Left contains 893 more items, first extra item: (712,)
E Use -v to get the full diff
```

test\_TP.py:137: AssertionError

test\_Q14\_req

```
def test_Q14_req ():
> assert requete(sol_Q14_req) == requete(Q14_req)
E assert [] == [(1,), (2,), ...), (6,), ...]
E Right contains 93 more items, first extra item: (1,)
E Use -v to get the full diff
```

test\_TP.py:143: AssertionError

test\_Q15\_res

```
def test_Q15_res ():
> assert sol_Q15_res.upper() == Q15_res.upper()
E assert "('SILLET', 'JACQUES', 891)" == 'GÉRARD,MATHIEU,1037'
E - ('SILLET', 'JACQUES', 891)
E + GÉRARD,MATHIEU,1037
```

test\_TP.py:146: AssertionError

test\_Q15\_req

```
def test_Q15_req ():
```

```
>         assert requete(sol_Q15_req) == requete(Q15_req)
E         AssertionError: assert [('SILLET', 'Jacques', 891)] == [('MATHIEU', 'Gérard', 1037)]
E             At index 0 diff: ('SILLET', 'Jacques', 891) != ('MATHIEU', 'Gérard', 1037)
E             Use -v to get the full diff
```

```
test_TP.py:149: AssertionError
```

```
===== 13 failed, 7 passed in 0.32s =====
```

```
NOM = "ABID"
Prenom = "Hamza"
Classe = "MPSI2"
alpha="32"
```

```
## Question 1
```

```
Q1_req = "SELECT TIT_CODE, CLI_NOM, CLI_PRENOM FROM T_CLIENT;"
```

```
## Question 2
```

```
Q2_req = "SELECT COUNT(*) FROM T_CLIENT;"
```

```
Q2_res = "93"
```

```
## Question 3
```

```
Q3_req = "SELECT CLI_NOM, CLI_PRENOM FROM T_CLIENT WHERE TIT_CODE='Mme.';"
```

```
## Question 4
```

```
Q4_req = "SELECT TIT_CODE, CLI_NOM, CLI_PRENOM FROM T_CLIENT WHERE TIT_CODE IN (
'Mme.', 'Melle.');"

```

```
## Question 5
```

```
Q5_req = "SELECT COUNT(*) FROM T_CLIENT WHERE TIT_CODE IN ('Mme.', 'Melle.');"

```

```
Q5_res = "16"
```

```
## Question 6
```

```
Q6_req = "SELECT CLI_NOM AS Noms, CLI_PRENOM AS Prenoms FROM T_CLIENT WHERE TIT_
CODE IN ('Mme.', 'Melle.') ORDER BY Noms, Prenoms ASC;"

```

```
## Question 7
```

```
Q7_req = "SELECT cl.CLI_NOM, cl.CLI_PRENOM, tel.TEL_NUMERO FROM T_CLIENT as cl I
NNER JOIN T_TELEPHONE as tel ON cl.CLI_ID = tel.CLI_ID;"

```

```
## Question 8
```

```
Q8_req = "SELECT CLI_NOM, CLI_PRENOM FROM T_CLIENT WHERE CLI_NOM IN (SELECT CLI_
NOM FROM T_CLIENT GROUP BY CLI_Nom HAVING COUNT(CLI_NOM) > 1);"

```

```
## Question 9
```

```
Q9_req = "SELECT CLI_NOM, COUNT(*) as nb_occur FROM T_CLIENT GROUP BY CLI_Nom HA
VING nb_occur > 1"

```

```
Q9_res = "(BENATTAR, 2) et (MARTIN, 2)"
```

```
## Question 10
```

```
Q10_req = "SELECT AVG(LIF_REMISE_POURCENT), AVG(LIF_REMISE_MONTANT) FROM T_LIGNE
_FACTURE;"

```

```
Q10_res = "Pourcentage : 47.0 et Montant : 81.0"
```

```
## Question 11
```

```
Q11_req = "SELECT MAX(LIF_REMISE_POURCENT), MAX(LIF_REMISE_MONTANT) FROM T_LIGNE
_FACTURE;"

```

```
Q11_res = "Pourcentage : 47 et Montant : 81"
```

```
## Question 12
```

```
Q12_req = "SELECT DISTINCT FAC_ID as fac_id1 FROM T_LIGNE_FACTURE WHERE (LIF_REM
ISE_MONTANT IS NOT NULL) OR (LIF_REMISE_POURCENT IS NOT NULL);"

```

```
## Question 13
```

```
Q13_req = "SELECT DISTINCT CLI_ID FROM T_FACTURE AS fac WHERE fac.FAC_ID IN (SEL
ECT DISTINCT FAC_ID as fac_id1 FROM T_LIGNE_FACTURE WHERE (LIF_REMISE_MONTANT IS
NOT NULL) OR (LIF_REMISE_POURCENT IS NOT NULL));"

```

```
## Question 14
```

```
Q14_req = "SELECT DISTINCT CLI_ID FROM T_FACTURE AS fac WHERE fac.FAC_ID IN (SEL
ECT DISTINCT FAC_ID FROM T_LIGNE_FACTURE WHERE (LIF_REMISE_MONTANT IS NULL) AND
(LIF_REMISE_POURCENT IS NULL));"

```

```
## Question 15
```

```
Q15_req = "SELECT cli.CLI_NOM as nom, cli.CLI_PRENOM as prenom, SUM(LIF_REMISE_M
ONTANT)+SUM(LIF_REMISE_POURCENT) as valeur_max FROM T_FACTURE as fac INNER JOIN
T_CLIENT as cli ON fac.CLI_ID = cli.CLI_ID INNER JOIN (SELECT CLI_ID, MAX(nb) as
```

```
max_nb FROM (SELECT CLI_ID, COUNT(*) as nb FROM T_FACTURE AS fac WHERE fac.FAC_ID IN (SELECT DISTINCT FAC_ID as fac_id1 FROM T_LIGNE_FACTURE WHERE (LIF_REMISE_MONTANT IS NOT NULL) OR (LIF_REMISE_POURCENT IS NOT NULL)) GROUP BY CLI_ID)) as max_tab ON fac.CLI_ID = max_tab.CLI_ID INNER JOIN T_LIGNE_FACTURE as ligfac ON ligfac.FAC_ID = fac.FAC_ID;"
Q15_res ="Gérard,Mathieu,1037"
```

# Il y avait plusieurs personnes qui ont bénéficié du plus grand nombre de remis es or une seule était demandée.

# J'ai donc mis celle que me proposait directement sqlite