Faculté des Sciences d'Agadir – Ibn Zohr

Licence d'excellence
Filière : Analytique de Données et Intelligence
Artificielle

TP 3

COMPT RENDU

Réalisé Par :
Elhillali Naoual

- Réalisation de question 2 :


```
SQL> SELECT nometudiant ,prenometudiant,
  2   TO_CHAR(datenaissance, 'Day, Month DD, YYYY') AS date_de_naissance
  3   FROM
  4   etudiant;

NOMETUDIANT          PRENOMETUDIANT
-------------------- --------------------
DATE_DE_NAISSANCE
--------------------------------------------------------------------
elhillali            nawal
Saturday , January   15, 2000

alaoui               imane
Wednesday, January   15, 2003

benjaloun            nora
Tuesday  , November  25, 2003


NOMETUDIANT          PRENOMETUDIANT
-------------------- --------------------
DATE_DE_NAISSANCE
--------------------------------------------------------------------
benani               ahmed
Tuesday  , November  30, 2004

kahtani              mustapha
Monday   , March     29, 2004

bahaoui              said
```

- Réalisation de question 3 et 4 et 5:


```
SQL> SELECT
  2       AVG(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_moyen
  3   FROM
  4       etudiant;

 AGE_MOYEN
----------
      20.5

SQL> SELECT
  2       AVG(MONTHS_BETWEEN(SYSDATE, datenaissance) / 12) AS age_moyen
  3   FROM
  4       etudiant;

 AGE_MOYEN
----------
 21.032406

SQL> SELECT
  2       COUNT(*) AS nombre_etudiants,AVG(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_moyen
  3   FROM
  4       etudiant;

NOMBRE_ETUDIANTS  AGE_MOYEN
---------------- ----------
              6       20.5

SQL> select
  2   count(*) as nombre_etudiants,
  3   min(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_minimum,
```

- Réalisation de question 6 :


```
2      COUNT(*) AS nombre_etudiants,AVG(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_moyen
3  FROM
4      etudiant;

NOMBRE_ETUDIANTS  AGE_MOYEN
---------------- ----------
              6       20.5

SQL>  select
  2   count(*) as nombre_etudiants,
  3   min(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_minimum,
  4   max(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_maximum,
  5   AVG(EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM datenaissance)) AS age_moyen
  6   from
  7   etudiant;

NOMBRE_ETUDIANTS AGE_MINIMUM AGE_MAXIMUM  AGE_MOYEN
---------------- ----------- ----------- ----------
              6          19          23       20.5

SQL>
```

- Réalisation de question 6 :


```
SQL> SELECT
  2      EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3      COUNT(*) AS nombre_etudiant,
  4   AVG(note) AS moyenne_notes,
  5      MAX(note) AS note_maximum,
  6      MIN(note) AS note_minimum
  7  FROM
  8      etudiant
  9  JOIN
 10  resultat on etudiant.codeetudiant=resultat.codeetudiant
 11  GROUP BY
 12      EXTRACT(YEAR FROM datenaissance);

ANNEE_NAISSANCE NOMBRE_ETUDIANT MOYENNE_NOTES NOTE_MAXIMUM NOTE_MINIMUM
--------------- --------------- ------------- ------------ ------------
           2000               1          18.9         18.9         18.9
           2003               1         15.75        15.75        15.75
           2004               2         15.75         16.8         14.7

SQL> SELECT
  2      EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3      COUNT(*) AS nombre_etudiants,
  4      AVG(note) AS moyenne_notes,
  5      MAX(note) AS note_maximum,
  6      MIN(note) AS note_minimum
  7  FROM
  8      etudiant
  9
SQL> JOIN
```

- Réalisation de question 7 :

```
SQL> SELECT
  2      EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3      COUNT(*) AS nombre_etudiants,
  4      AVG(note) AS moyenne_notes,
  5      MAX(note) AS note_maximum,
  6      MIN(note) AS note_minimum
  7  FROM
  8      etudiant
  9  JOIN
 10  resultat on etudiant.codeetudiant=resultat.codeetudiant
 11  WHERE
 12      EXTRACT(YEAR FROM datenaissance) IN (1, 2)
 13  GROUP BY
 14      EXTRACT(YEAR FROM datenaissance);

no rows selected

SQL> SELECT
  2      EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3      ville,
  4      COUNT(*) AS nombre_etudiants,
  5      AVG(note) AS moyenne_notes,
  6      MAX(note) AS note_maximum,
  7      MIN(note) AS note_minimum
  8  FROM
  9      etudiant
 10  JOIN
 11  resultat on etudiant.codeetudiant=resultat.codeetudiant
 12  WHERE
 13      ville IN ('Agadir', 'casa')
```

- Réalisation de question 8:


```
 14  GROUP BY
 15      EXTRACT(YEAR FROM datenaissance), ville;

ANNEE_NAISSANCE VILLE      NOMBRE_ETUDIANTS MOYENNE_NOTES NOTE_MAXIMUM
--------------- ---------- ---------------- ------------- ------------
NOTE_MINIMUM
------------
           2000 Agadir                    1          18.9         18.9
        18.9

           2004 casa                      1          16.8         16.8
        16.8

SQL> SELECT
  2      EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3      ville,
  4      COUNT(*) AS nombre_etudiants,
  5      AVG(note) AS moyenne_notes,
  6      MAX(note) AS note_maximum,
  7      MIN(note) AS note_minimum
  8  FROM
  9  etudiant
 10  JOIN
 11  resultat on etudiant.codeetudiant=resultat.codeetudiant
 12  WHERE
 13      EXTRACT(YEAR FROM datenaissance) IN (1, 2)
 14      AND (ville = 'Marrakech' OR ville = 'Taroudant')
 15  GROUP BY
 16      EXTRACT(YEAR FROM datenaissance), ville;
```

- Réalisation de question 9 :

```
15  GROUP BY
16      EXTRACT(YEAR FROM datenaissance), ville;

no rows selected

SQL> SELECT
 2      EXTRACT(YEAR FROM e.datenaissance) AS annee_naissance,
 3      COUNT(*) AS nombre_etudiants,
 4      AVG(r.note) AS moyenne_notes,
 5      MAX(r.note) AS note_maximum,
 6      MIN(r.note) AS note_minimum
 7  FROM
 8      etudiant e
 9  JOIN
10      resultat r ON e.codeetudiant=r.codeetudiant
11  GROUP BY
12      EXTRACT(YEAR FROM e.datenaissance)
13  HAVING
14      AVG(r.note) BETWEEN 12 AND 16;

ANNEE_NAISSANCE NOMBRE_ETUDIANTS MOYENNE_NOTES NOTE_MAXIMUM NOTE_MINIMUM
--------------- ---------------- ------------- ------------ ------------
           2003                1         15.75        15.75        15.75
           2004                2         15.75         16.8         14.7

SQL>
```

Réalisation de question 10 :


```
13  ORDER BY
14      moyenne_notes DESC;
    etudiants e
    *
ERROR at line 8:
ORA-00942: table or view does not exist

SQL> SELECT
 2      EXTRACT(YEAR FROM e.datenaissance) AS annee_naissance,
 3      COUNT(*) AS nombre_etudiants,
 4      AVG(r.note) AS moyenne_notes,
 5      MAX(r.note) AS note_maximum,
 6      MIN(r.note) AS note_minimum
 7  FROM
 8      etudiant e
 9  JOIN
10  resultat r ON e.codeetudiant=r.codeetudiant
11  GROUP BY
12  EXTRACT(YEAR FROM e.datenaissance)
13  ORDER BY
14  moyenne_notes DESC;

ANNEE_NAISSANCE NOMBRE_ETUDIANTS MOYENNE_NOTES NOTE_MAXIMUM NOTE_MINIMUM
--------------- ---------------- ------------- ------------ ------------
           2000                1          18.9         18.9         18.9
           2004                2         15.75         16.8         14.7
           2003                1         15.75        15.75        15.75

SQL>
```

Réalisation de question 11 :

```
Command Prompt - sqlplus  / as sysdba                                        —    □    ✕
SQL> SELECT
  2  EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3  COUNT(*) AS nombre_etudiants,
  4      AVG(note) AS moyenne_notes,
  5      MAX(note) AS note_maximale,
  6      MIN(note) AS note_minimale
  7  from etudiant
  8  join
  9  resultat on etudiant.codeetudiant=resultat.codeetudiant
 10   GROUP BY EXTRACT(YEAR FROM datenaissance)
 11   ORDER BY moyenne_notes DESC;

ANNEE_NAISSANCE NOMBRE_ETUDIANTS MOYENNE_NOTES NOTE_MAXIMALE NOTE_MINIMALE
--------------- ---------------- ------------- ------------- -------------
           2000                1          18.9          18.9          18.9
           2004                2         15.75          16.8          14.7
           2003                1         15.75         15.75         15.75

SQL> select
  2  EXTRACT(YEAR FROM datenaissance) AS annee_naissance,
  3  COUNT(*) AS nombre_etudiants,
  4  AVG(note) AS moyenne_notes,
  5  MAX(note) AS note_maximale,
  6   MIN(note) AS note_minimale
  7   from etudiant
  8   join
  9  resultat on etudiant.codeetudiant=resultat.codeetudiant
 10  GROUP BY EXTRACT(YEAR FROM datenaissance)
 11  ORDER BY  nombre_etudiants, moyenne_notes DESC;
```

Réalisation de question 12:

```
Command Prompt - sqlplus  / as sysdba                                        —    □    ✕
ANNEE_NAISSANCE NOMBRE_ETUDIANTS MOYENNE_NOTES NOTE_MAXIMALE NOTE_MINIMALE
--------------- ---------------- ------------- ------------- -------------
           2000                1          18.9          18.9          18.9
           2003                1         15.75         15.75         15.75
           2004                2         15.75          16.8          14.7

SQL> SELECT
  2      CONCAT(UPPER(SUBSTRING(nomenseignant, 1, 1)), LOWER(SUBSTRING(nomenseignant, 2))) AS nom_formate
  3  FROM Enseignant
  4  WHERE nomenseignant LIKE '%i';
WHERE nomenseignant LIKE '%i'
      *
ERROR at line 4:
ORA-00904: "NOMENSEIGNANT": invalid identifier


SQL> SELECT
  2   CONCAT(UPPER(SUBSTRING(nomeenseignant, 1, 1)), LOWER(SUBSTRING(nomeenseignant, 2))) AS nom_formate
  3  FROM Enseignant
  4  WHERE nomeenseignant LIKE '%i';
 CONCAT(UPPER(SUBSTRING(nomeenseignant, 1, 1)), LOWER(SUBSTRING(nomeenseignant, 2))) AS nom_formate
                                        *
ERROR at line 2:
ORA-00904: "SUBSTRING": invalid identifier


SQL> SELECT
  2      INITCAP(nomeenseignant) AS nom_formate
  3  FROM Enseignant
```

Réalisation de question 13 :

```
Command Prompt - sqlplus  / as sysdba
SQL> SELECT
  2    INITCAP(nomenseignant) AS nom_formate
  3   FROM Enseignant
  4   WHERE nomenseignant LIKE '%i';

no rows selected

SQL> SELECT
  2   INITCAP(nomenseignant) AS nom_formate
  3   FROM Enseignant
  4   WHERE nomenseignant LIKE '%e';

NOM_FORMATE
--------------------
Lefevre

SQL> SELECT
  2   nomenseignant
  3   FROM Enseignant
  4   JOIN Charge ON enseignant.codeenseignant=charge.codeenseignant
  5   GROUP BY nomenseignant
  6   HAVING COUNT(charge.codecours) > 2;
JOIN Charge ON enseignant.codeenseignant=charge.codeenseignant
            *
ERROR at line 4:
ORA-00904: "ENSEIGNANT"."CODEENSEIGNANT": invalid identifier


SQL> SELECT
  2   nomenseignant
```

Réalisation de question 14 :


```
Command Prompt - sqlplus  / as sysdba
SQL> SELECT
  2   nomenseignant
  3    FROM Enseignant
  4   JOIN Charge ON enseignant.codeensignant=charge.codeenseignant
  5    GROUP BY nomenseignant
  6    HAVING COUNT(charge.codecours) > 2;

no rows selected

SQL> select
  2   nomenseignant
  3   from enseignant
  4   where specialite is not null;

NOMEENSEIGNANT
--------------------
Martin
Dupont
Leclerc
Lefevre
Roux

SQL> select
  2   nomenseignant,specialite
  3   from enseignant
  4   where specialite is not null;

NOMEENSEIGNANT       SPECIALITE
-------------------- --------------------
Martin               Mathématiques
```

Réalisation de question 15 :

```
SQL> select
  2  nomeenseignant
  3  from enseignant
  4  where specialite is not null;

NOMEENSEIGNANT
--------------------
Martin
Dupont
Leclerc
Lefevre
Roux

SQL> select
  2  nomeenseignant,specialite
  3  from enseignant
  4  where specialite is not null;

NOMEENSEIGNANT          SPECIALITE
-------------------- --------------------
Martin                  Mathématiques
Dupont                  Informatique
Leclerc                 Physique
Lefevre                 Chimie
Roux                    Anglais

SQL> _
```

Réalisation de question 16 :

```
SQL> SELECT nomeenseignant ,specialite
  2  from enseignant  where specialite IN (
  3  SELECT specialite FROM enseignant
  4  GROUP BY specialite
  5  HAVING COUNT(*) > 1
  6  );

no rows selected
```

Réalisation de question 17 :

```
SQL> SELECT
  2      e1.nomeenseignant AS enseignant1,
  3    e2.nomeenseignant AS enseignant2,
  4    c.intitule AS cours
  5    FROM
  6    enseignant e1
  7    JOIN
  8    charge ch1 ON e1.codeensignant = ch1.codeenseignant
  9    JOIN
 10   cours c ON ch1.codecours = c.codecours
 11    JOIN
 12   charge ch2 ON c.codecours = ch2.codecours
 13    JOIN
 14   enseignant e2 ON ch2.codeenseignant = e2.codeensignant
 15    WHERE
 16   e1.codeensignant < e2.codeensignant;

no rows selected
```

Réalisation de question 18 - 19:

```
SQL> CREATE TABLE ETUDIANTS AS
  2  SELECT * FROM ETUDIANT WHERE 1 = 0;

Table created.

SQL> INSERT INTO ETUDIANTS (codeetudiant, nometudiant, prenometudiant, datenaissance, ville)
  2  SELECT DISTINCT codeetudiant, nometudiant, prenometudiant, datenaissance, ville
  3  FROM ETUDIANT;

6 rows created.

SQL> SELECT * FROM ETUDIANT
  2  UNION
  3  SELECT * FROM ETUDIANTS;

CODEETUDIA NOMETUDIANT          PRENOMETUDIANT       DATENAISS VILLE
---------- -------------------- -------------------- --------- ----------
1          elhillali            nawal                15-JAN-00 Agadir
2          alaoui               imane                15-JAN-03 Agadir
3          benjaloun            nora                 25-NOV-03 tiznit
4          benani               ahmed                30-NOV-04 casa
5          kahtani              mustapha             29-MAR-04 safi
6          bahaoui              said                 19-JUN-01 taroudant

6 rows selected.
```

Réalisation de question 20:

```
SQL> SELECT * FROM ETUDIANT
  2  WHERE EXTRACT(YEAR FROM datenaissance) = 2004
  3  UNION
  4  SELECT * FROM ETUDIANTS
  5  WHERE EXTRACT(YEAR FROM datenaissance) = 2004;

CODEETUDIA NOMETUDIANT          PRENOMETUDIANT       DATENAISS VILLE
---------- -------------------- -------------------- --------- ----------
4          benani               ahmed                30-NOV-04 casa
5          kahtani              mustapha             29-MAR-04 safi

SQL> SELECT * FROM ETUDIANT
  2  WHERE EXTRACT(YEAR FROM datenaissance) = 2000
  3  UNION
  4  SELECT * FROM ETUDIANTS
  5  WHERE EXTRACT(YEAR FROM datenaissance) = 2003;

CODEETUDIA NOMETUDIANT          PRENOMETUDIANT       DATENAISS VILLE
---------- -------------------- -------------------- --------- ----------
1          elhillali            nawal                15-JAN-00 Agadir
2          alaoui               imane                15-JAN-03 Agadir
3          benjaloun            nora                 25-NOV-03 tiznit
```

Réalisation de question 21 :

```
SQL> SELECT
  2       r.codeetudiant,
  3  e.nometudiant AS Nom,
  4   AVG(r.note) AS Moyenne,
  5      MIN(r.note) AS Minimum,
  6      MAX(r.note) AS Maximum
  7  FROM
  8      resultat r
  9  JOIN
 10      ETUDIANT e ON r.codeetudiant = e.codeetudiant
 11  GROUP BY
 12      r.codeetudiant, e.nometudiant;

CODEETUDIA NOM                    MOYENNE    MINIMUM    MAXIMUM
---------- -------------------- ---------- ---------- ----------
1          elhillali                 18.9       18.9       18.9
3          benjaloun                15.75      15.75      15.75
4          benani                    16.8       16.8       16.8
5          kahtani                   14.7       14.7       14.7
```

Réalisation de question 22:

```
SQL> SELECT
  2       e.codeetudiant,
  3       e.nometudiant AS Nom,
  4       AVG(r.note) AS Moyenne,
  5       MIN(r.note) AS Minimum,
  6       MAX(r.note) AS Maximum
  7  FROM
  8      ETUDIANT e
  9  JOIN
 10      resultat r ON e.codeetudiant = r.codeetudiant
 11  WHERE
 12      EXTRACT(YEAR FROM e.datenaissance) = 2003
 13  GROUP BY
 14      e.codeetudiant, e.nometudiant;

CODEETUDIA NOM                    MOYENNE    MINIMUM    MAXIMUM
---------- -------------------- ---------- ---------- ----------
3          benjaloun                15.75      15.75      15.75
```

Réalisation de question 23 :

```
SQL> SELECT
  2       e.codeetudiant,
  3       e.nometudiant AS Nom,
  4       AVG(r.note) AS Moyenne,
  5       MIN(r.note) AS Minimum,
  6       MAX(r.note) AS Maximum
  7  FROM
  8       ETUDIANT e
  9  JOIN
 10       resultat r ON e.codeetudiant = r.codeetudiant
 11  GROUP BY
 12       e.codeetudiant, e.nometudiant
 13  HAVING
 14       AVG(r.note) > 11;

CODEETUDIA NOM                        MOYENNE    MINIMUM    MAXIMUM
---------- -------------------- ---------- ---------- ----------
1          elhillali                 18.9       18.9       18.9
3          benjaloun                15.75      15.75      15.75
4          benani                    16.8       16.8       16.8
5          kahtani                   14.7       14.7       14.7
```

Réalisation de question 24:

```
SQL> SELECT
  2       e.codeetudiant,
  3       e.nometudiant AS Nom,
  4       AVG(r.note) AS Moyenne,
  5       MIN(r.note) AS Minimum,
  6       MAX(r.note) AS Maximum
  7  FROM
  8       ETUDIANT e
  9  JOIN
 10       resultat r ON e.codeetudiant = r.codeetudiant
 11  WHERE
 12       EXTRACT(YEAR FROM e.datenaissance) = 2003
 13  GROUP BY
 14       e.codeetudiant, e.nometudiant
 15  HAVING
 16       AVG(r.note) > 12;

CODEETUDIA NOM                        MOYENNE    MINIMUM    MAXIMUM
---------- -------------------- ---------- ---------- ----------
3          benjaloun                15.75      15.75      15.75
```

Réalisation de question 25 :

```
SQL> SELECT
  2      e.codeetudiant AS "Numéro",
  3      e.nometudiant AS "Nom",
  4      AVG(r.note) AS "Moyenne"
  5  FROM
  6      ETUDIANT e
  7  JOIN
  8      resultat r ON e.codeetudiant = r.codeetudiant
  9  GROUP BY
 10      e.codeetudiant, e.nometudiant;

Numéro      Nom                      Moyenne
---------   --------------------     ----------
1           elhillali                    18.9
3           benjaloun                   15.75
4           benani                       16.8
5           kahtani                      14.7
```

Réalisation de question 26 :

```
SQL> SELECT
  2      E.nomeenseignant AS NomEnseignant,
  3      E.specialite
  4  FROM
  5      enseignant E
  6  WHERE
  7      NOT EXISTS (
  8          SELECT
  9              C.codecours
 10          FROM
 11              cours C
 12          WHERE
 13              C.codecours <> 'Reseaux'
 14              AND NOT EXISTS (
 15                  SELECT
 16                      NULL
 17                  FROM
 18                      charge Ch
 19                  WHERE
 20                      Ch.codeenseignant = E.codeensignant
 21                      AND Ch.codecours = C.codecours
 22              )
 23      );

no rows selected
```

Réalisation de question 27 :

```
SQL> SELECT DISTINCT
  2      E.*
  3  FROM
  4      enseignant E
  5  WHERE
  6      EXISTS (
  7          SELECT 1
  8          FROM charge C
  9          WHERE C.codeenseignant = E.codeensignant
 10      );

CODEENSIGN NOMEENSEIGNANT       PRENOMENSEIGNANT       SPECIALITE
---------- -------------------- -------------------- --------------------
E001       Martin               Jean                 Mathématiques
E003       Leclerc              Pierre               Physique
E004       Lefevre              Marie                Chimie
E005       Roux                 Luc                  Anglais
```

Réalisation de question 28 :

```
SQL> CREATE VIEW VueEtudiants1ereAnnee AS
  2  SELECT
  3      e.codeetudiant AS "Numéro",
  4      nometudiant AS "Nom",
  5      AVG(note) AS "Moyenne"
  6  FROM
  7      ETUDIANT e
  8  JOIN
  9      resultat r ON e.codeetudiant = r.codeetudiant
 10  WHERE
 11      EXTRACT(YEAR FROM e.datenaissance) = 2003
 12  GROUP BY
 13      e.codeetudiant, e.nometudiant;

View created.
```

Réalisation de question 29 :

```
SQL> SELECT * FROM VueEtudiants1ereAnnee;

Numéro     Nom                     Moyenne
---------- -------------------- ----------
3          benjaloun                 15.75

SQL>
```