

# 1 Correction du TD 8

## Première demi-heure : Récupérer les données

### Exercice 1

```
SELECT COUNT(*) FROM (  
    SELECT DISTINCT last_update FROM td8_velib  
) ;
```

```
SELECT MIN(last_update), MAX(last_update) FROM td8_velib ;
```

## Seconde demi-heure : GROUP BY

```
SELECT number, COUNT(*) AS nb  
FROM td8_velib  
WHERE available_bikes==0 AND last_update >= '2013-09-10 11:30:19'  
GROUP BY number  
ORDER BY nb DESC
```

```
SELECT nb, COUNT(*) AS nb_station  
FROM (  
    -- requête de l'exercice précédent  
    SELECT number, COUNT(*) AS nb  
    FROM td8_velib  
    WHERE available_bikes==0 AND last_update >= '2013-09-10 11:30:19'  
    GROUP BY number  
)  
GROUP BY nb
```

## Troisième demi-heure : JOIN

```
SELECT A.number, A.heure, A.minute, 1.0 * A.nb_velo / B.nb_velo_tot AS distribution_temporelle  
FROM (  
    SELECT number, heure, minute, SUM(available_bikes) AS nb_velo  
    FROM td8_velib  
    WHERE last_update >= '2013-09-10 11:30:19'  
    GROUP BY heure, minute, number  
) AS A  
JOIN (  
    SELECT number, heure, minute, SUM(available_bikes) AS nb_velo_tot  
    FROM td8_velib  
    WHERE last_update >= '2013-09-10 11:30:19'  
    GROUP BY number  
) AS B  
ON A.number == B.number  
--WHERE A.number in (8001, 8003, 15024, 15031) -- pour n'afficher que quelques stations  
ORDER BY A.number, A.heure, A.minute
```

## Quatrième demi-heure : zones de travail et zones de résidences

```
SELECT number, SUM(distribution_temporelle) AS velo_jour
FROM (
  -- requête de l'exercice 4
  SELECT A.number, A.heure, A.minute, 1.0 * A.nb_velo / B.nb_velo_tot AS distribution_temporelle
  FROM (
    SELECT number, heure, minute, SUM(available_bikes) AS nb_velo
    FROM td8_velib
    WHERE last_update >= '2013-09-10 11:30:19'
    GROUP BY heure, minute, number
  ) AS A
  JOIN (
    SELECT number, heure, minute, SUM(available_bikes) AS nb_velo_tot
    FROM td8_velib
    WHERE last_update >= '2013-09-10 11:30:19'
    GROUP BY number
  ) AS B
  ON A.number == B.number
)
WHERE heure >= 10 AND heure <= 16
GROUP BY number
```

## Pour aller plus loin ou pour ceux qui ont fini plus tôt

On trouve les arrondissements où les stations de vélib sont les plus remplies en journée au centre de Paris.

```
SELECT C.number, name, lat, lng, velo_jour FROM
(
  -- requête de la partie précédente
  SELECT number, SUM(distribution_temporelle) AS velo_jour
  FROM (
    -- requête de l'exercice 4
    SELECT A.number, A.heure, A.minute, 1.0 * A.nb_velo / B.nb_velo_tot AS distribution_temporelle
    FROM (
      SELECT number, heure, minute, SUM(available_bikes) AS nb_velo
      FROM td8_velib
      WHERE last_update >= '2013-09-10 11:30:19'
      GROUP BY heure, minute, number
    ) AS A
    JOIN (
      SELECT number, heure, minute, SUM(available_bikes) AS nb_velo_tot
      FROM td8_velib
      WHERE last_update >= '2013-09-10 11:30:19'
      GROUP BY number
    ) AS B
    ON A.number == B.number
  )
  WHERE heure >= 10 AND heure <= 16
  GROUP BY number
) AS C
INNER JOIN stations
ON C.number == stations.number
```

fin correction TD ?? □