

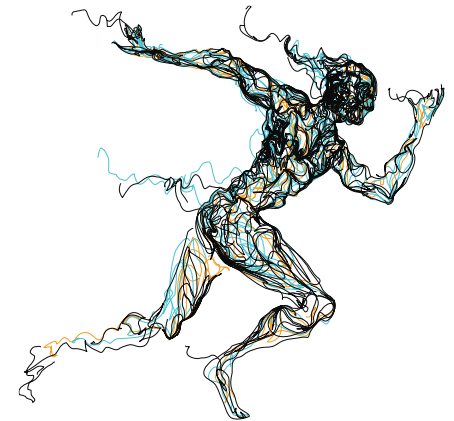
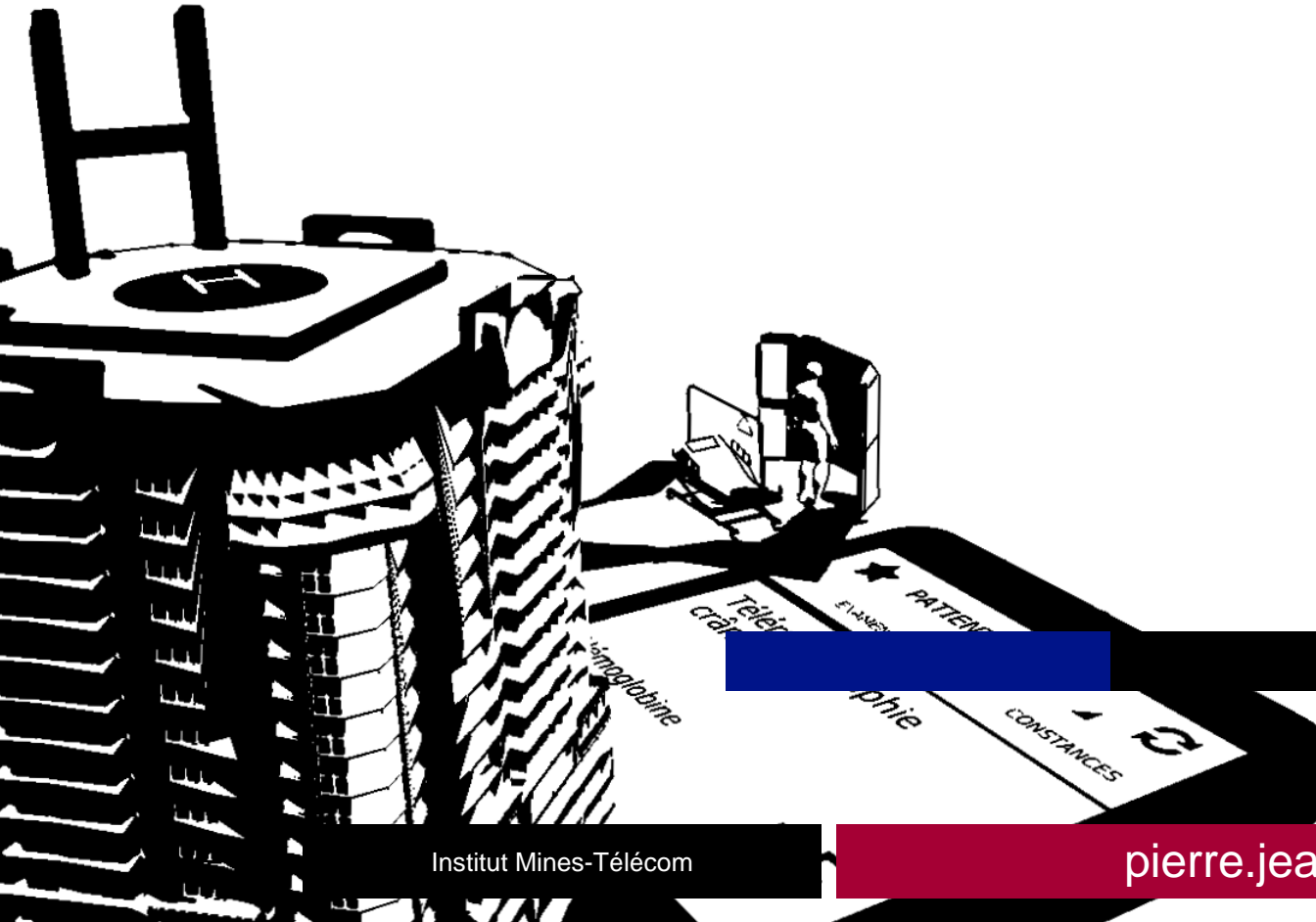


**IMT Mines Alès**  
École Mines-Télécom

# Master m-Rehab

## Traitement de données

### Pierre JEAN



## Les intervenants

Pierre Jean

Ingénieur de recherche IMT Mines Alès

IMT Mines d'Alès – EuroMov Digital Health in Motion



## Objectifs de la formation

Vocabulaire et concepts de bases de l'informatique

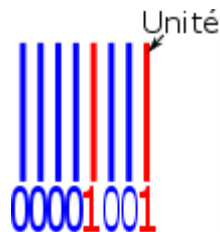
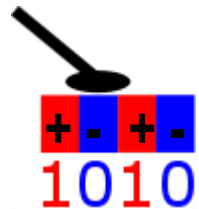
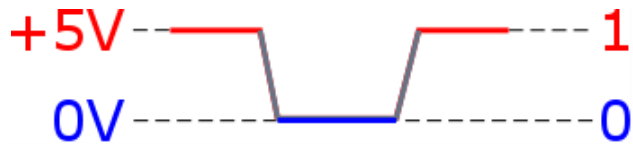
Nouvelles technique d'analyse numérique des données



## INFORMAtion de manière autoMATIQUE

### Bit électrique ou magnétique

8 bits = 1 byte



Binaire Base 2	Base 16	Binaire Base 2	Base 16
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

# Encodage des caractères

## Table des caractères ASCII

Hex	Char	Hex	Char	Hex	Char
20		40	@	60	`
21	!	41	A	61	a
22	..	42	B	62	b
23	#	43	C	63	c
24	\$	44	D	64	d
25	%	45	E	65	e
26	&	46	F	66	f
27	'	47	G	67	g
28	(	48	H	68	h
29	)	49	I	69	i
2A	*	4A	J	6A	j
2B	+	4B	K	6B	k
2C	,	4C	L	6C	l
2D	-	4D	M	6D	m
2E	.	4E	N	6E	n
2F	/	4F	O	6F	o
30	0	50	P	70	p
31	1	51	Q	71	q
32	2	52	R	72	r
33	3	53	S	73	s
34	4	54	T	74	t
35	5	55	U	75	u
36	6	56	V	76	v
37	7	57	W	77	w
38	8	58	X	78	x
39	9	59	Y	79	y
3A	:	5A	Z	7A	z
3B	;	5B	[	7B	{
3C	<	5C	\	7C	
3D	=	5D	]	7D	}
3E	>	5E	^	7E	~
3F	?	5F	-	7F	

## Table ISO8859-15

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	non utilisé															
1x	non utilisé															
2x		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8x	non utilisé															
9x	non utilisé															
Ax		ı	ç	£	€	¥	Š	š	©	ª	«	¬		®	-	
Bx	°	±	²	³	Ž	µ	¶	·	¸	¹	º	»	œ	ÿ	ı	
Cx	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	İ	Í	Î	Ï
Dx	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
Ex	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	ó	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

## UTF-8

si premier byte = 0 alors  
table ASCII  
sinon autre symbole

UTF-8	Serialized Bytes					
Unicode Range	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>
U-00000000 - U-0000007F	0nnnnnnn					
U-00000080 - U-000007FF	110nnnnn	10nnnnnn				
U-00000800 - U-0000FFFF	1110nnnn	10nnnnnn	10nnnnnn			
U-00010000 - U-001FFFFF	11110nnn	10nnnnnn	10nnnnnn	10nnnnnn		
U-00200000 - U-03FFFFFF	111110nn	10nnnnnn	10nnnnnn	10nnnnnn	10nnnnnn	
U-04000000 - U-7FFFFFFF	1111110n	10nnnnnn	10nnnnnn	10nnnnnn	10nnnnnn	10nnnnnn

Langage machine ou Assembleur 4 bytes pour une instruction



Extrait du programme en langage C équivalent

```
main()
{
  int i = 4 + 3;
}
```

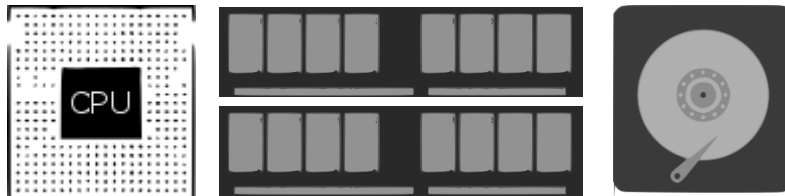
# Compilation

```
main()  
{  
  int i = 4 + 3;  
}
```

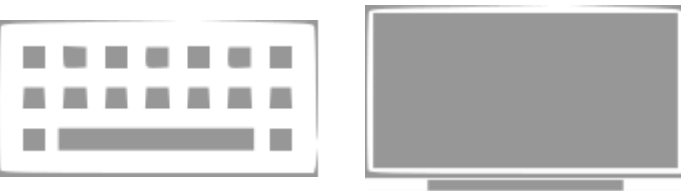
Programme  
en langage C



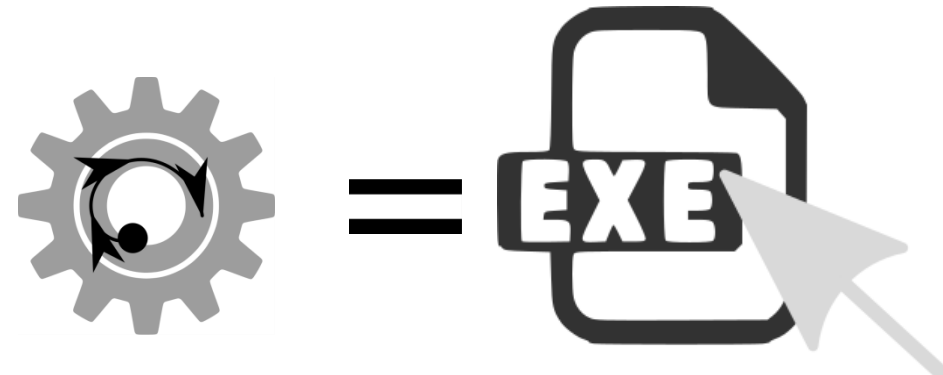
Bibliothèques



Processeur, mémoire, disque dur



Périphériques

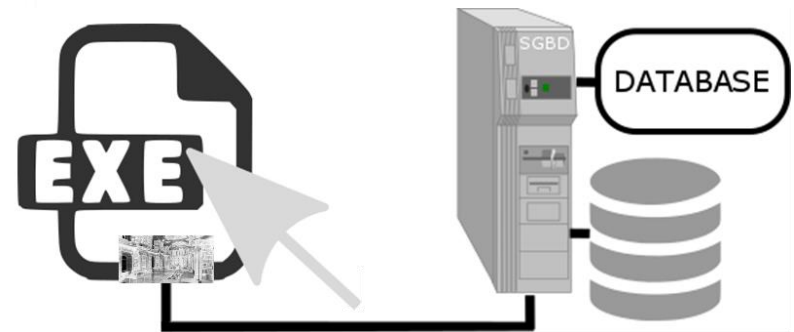


Le compilateur fabrique un  
exécutable (ou bibliothèque) pour  
un type de processeur et un  
système d'exploitation

# Logiciel SGBD ou Serveur de base de données



**Client SGBD connecté à un serveur SGBD en local ou en réseau**



**Logiciel avec bibliothèque SGBD connecté à un serveur SGBD en local ou en réseau**



**Logiciel sur un serveur Web avec bibliothèque SGBD connecté à un serveur SGBD en local ou en réseau**

## Accès au serveur de base de données



```
mysql --host=db.imt-ales.fr --database=projet1 --user=user1 --port=3306 --password
```



```
DB<-dbConnect(MySQL(), user="root", host="db.imt-ales.fr"  
, password="", dbname="projet1", port=3306)
```





# Application sur les bases de données relationnelles

- **Serveur de base de données = La base de données (erreur de langage)**
- **SGBD = Logiciel de base de données client et serveur**
- **Base de données = Ensemble de tables (classeur Excel)**
- **Table de base de données = Une table (1 feuille Excel)**
- **Colonne de base de données = Champ de la table**
- **Ligne de base de données = Enregistrement de la table**

Server	GPU max	RAM max	Core Max	HD Max	CPU	GHz	IP	Power	Model	Service ta	URL/Port	Ushe	Finance	Details
mshab	4	320	24	1800	2xPFC 7401	2,8	159.31.200.39	1410W	PowerEdge R742-DCSR22	VPRI/ssh	Calc	LG2P	4 x Tesla T4 16GB	
mshab2	4	320	16	1800	2xPFC 7501	2,9	159.31.200.42	(1-1)	1600W	PowerEdge R742-2F1213	VPRI/ssh	Calc	Projet Mshab	4 x Tesla T4 16GB
mshab3	2	200	12	1800	2xPFC 2617	3,5	159.31.200.22	(1-1)	1800W	PowerEdge T610	BTTCM2	ssh	Calc	2 x 074 1080 T1

```

c:\Developpement\wamp64\bin\mysql\mysql5.7.11\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 392
Server version: 5.7.11 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| eur-sante |
| eurosumup |
| event |
+-----+
mysql>rgunnd1
    
```

- Excel est un pseudo outil de base de données utile mais limité
- Utilisateurs de la base de données
- Schéma de la base de données
- Table et clef unique
- Liaisons

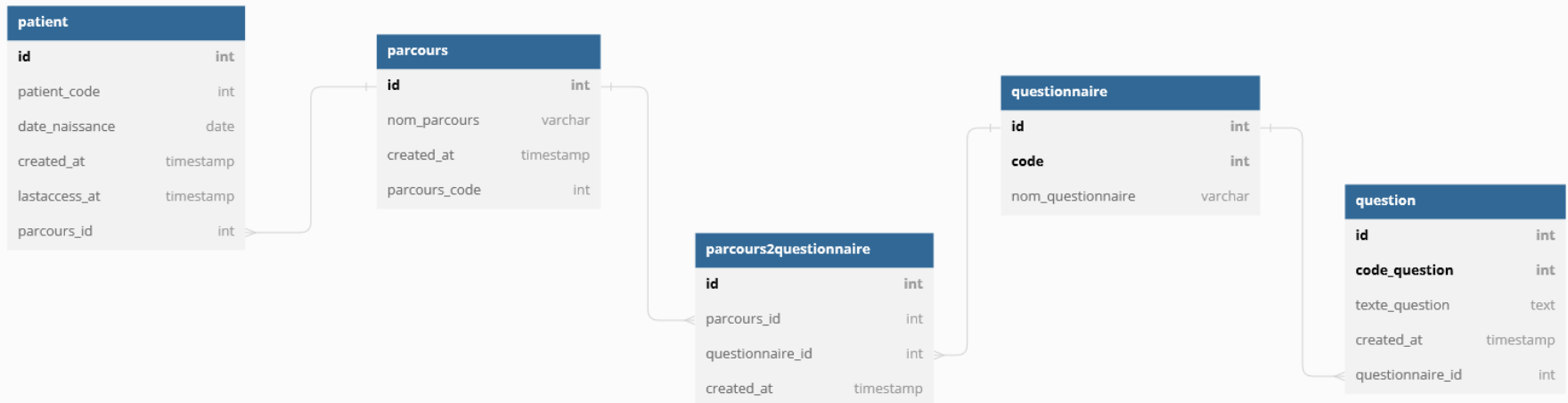
id	nom	prenom	email	adresse	tel	valide
1	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
2	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
3	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
4	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
5	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
6	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
7	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
8	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1
9	NCC1	Patrice	patrice.ncc1@mines-alès.fr	NCC1	00 00 00 00 00	1

A	B	C	D	E	F	G
1	Jobs	Serveur	Date de début			
2	Pierre	mrehab	01/01/2021			
3	Yu Du	mrehab	23-mars-21			
4	David ING	mrehab	01/04/2021			
5	Urtado - Mls	mrehab	19-avr-21			
6	*	ia0	08-mars-21			
7	Ali	ia0	01/01/2021			
8	Ali	ia0	01/01/2021			
9	Karim SemTe	ia0	08/04/2021			
10	Patrice : NCC	ia2	01/04/2021			
11	Patrice : NCC	ia2	01/04/2021			
12	Patrice : NCC	ia2	01/04/2021			
13	Patrice : NCC	ia2	01/04/2021			
14	Sebastien : N	ia1	01/04/2021			
15	Sebastien : N	ia1	01/04/2021			
16	Sebastien : N	ia1	01/04/2021			
17	Sebastien : N	ia1	01/04/2021			

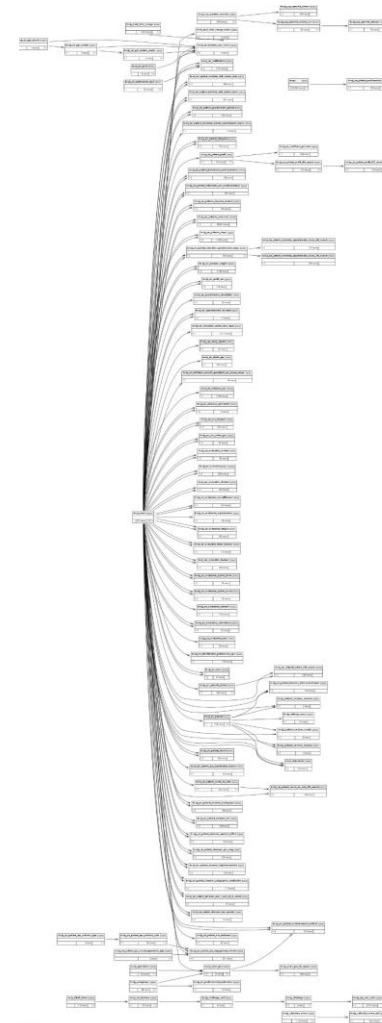
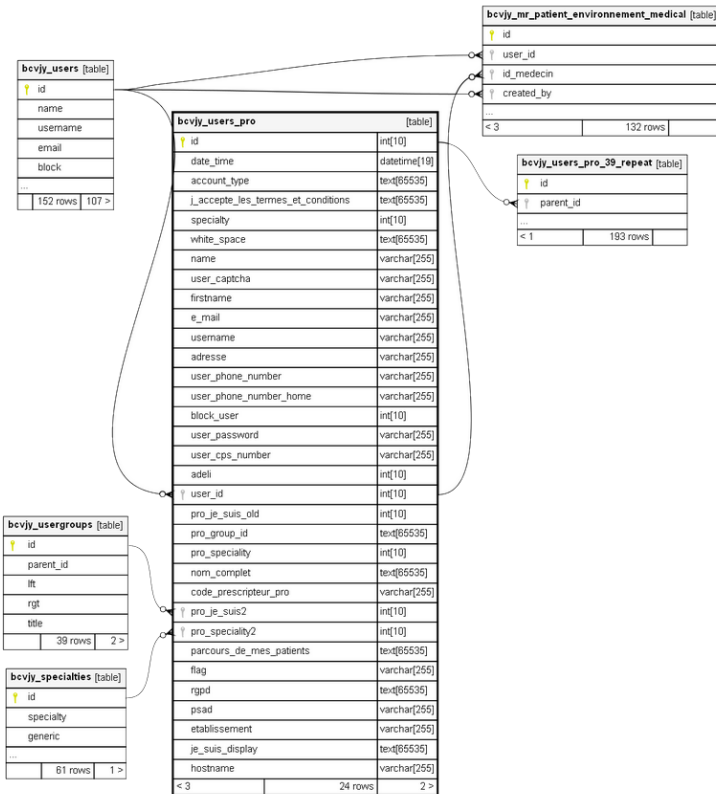
  

A	B	C	D
1	Server	GPU max	RAM max
2	ia0	4	320
3	mrehab	4	320
4	opencuda	2	200
5	ia1	4	320
6	ia2	4	320
7	kid2017	0	64
8	kidZ	0	64
9	ctn1	0	8
10	kidY	0	48
11	svn	0	32
12	liquid	0	4
13	kidx	0	64
14	cube	0	8
15	dev.lgi2p	0	24

# Schéma de base de données classique



# Schéma de base de données réels



## Description du besoin

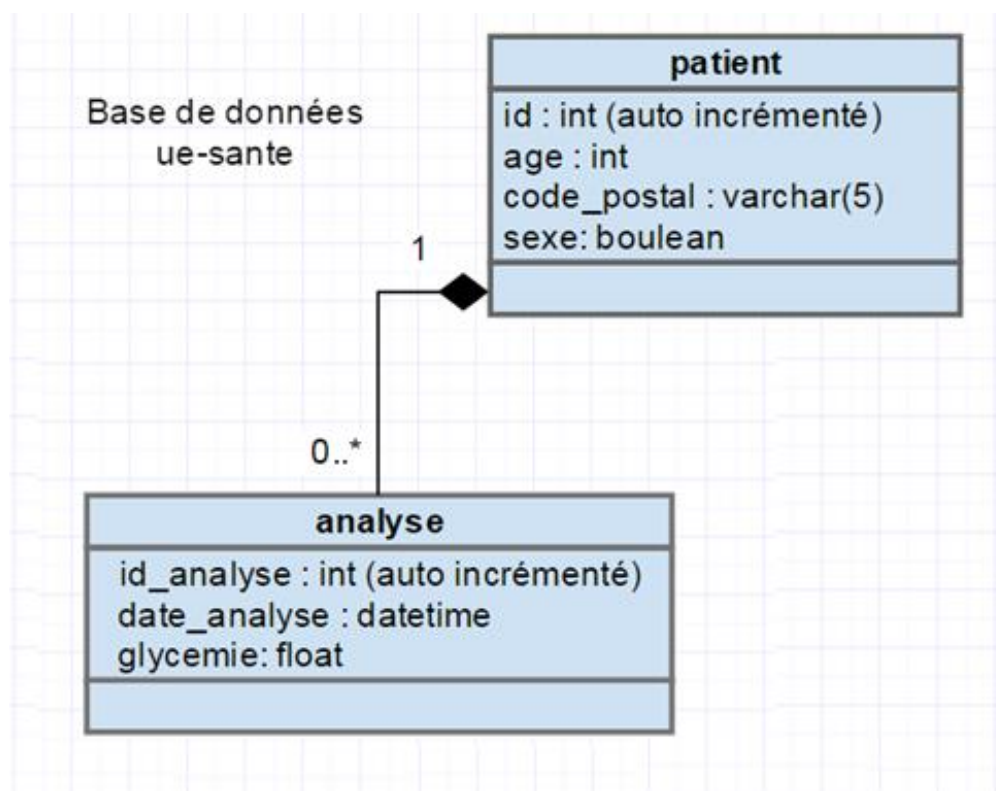
Un patient (âge, code postal, sexe) passe des analyses de glycémie ( date du test , glycémie en mmol/L )

id_analyse	Id_patient	date_analyse	glycemie
1	2	01/11/2016	8,4
2	2	02/11/2016	8,5

id	age	code_postal	sexe
1	55	31000	0
2	65	34000	1

## Relation 0-n entre le patient et ses analyses

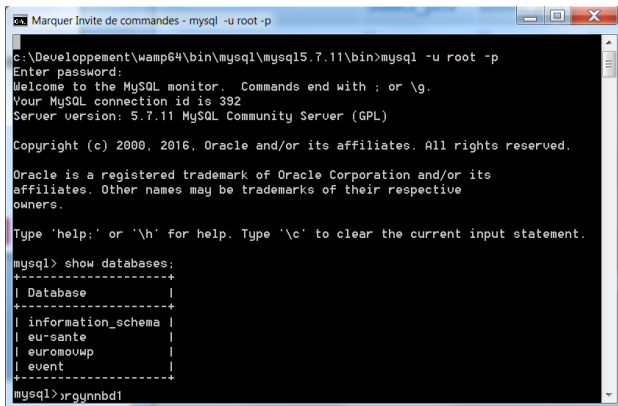
### Nature des données prévues



## Authentification des utilisateurs ( table mysql.user )

Host	User	authentication_string	Select_priv	Insert_priv	Update_priv
localhost	root		Y	Y	Y
localhost	mysql.sys	*3A819E0B8C4DFB30C6154338FEB70B93491D6D15	N	N	N
localhost	euromovwp	*263F3F0121A3975BDF5D85A0148274A6F1C4E497	N	N	N
localhost	ffnkbd	*8DA3D5DC5F3F5B30C6154338FEB70B93491D6D15	N	N	N
localhost	gynkbd45	*8DA3D5DC5F3F5B30C6154338FEB70B93491D6D15	N	N	N
localhost	event	*33A819E0B8C4DF1A42766EB782E0C53858E99677	N	N	N

## Gestion des droits sur les BDs, les tables et les champs Accès client-serveur ou Accès phpMyAdmin



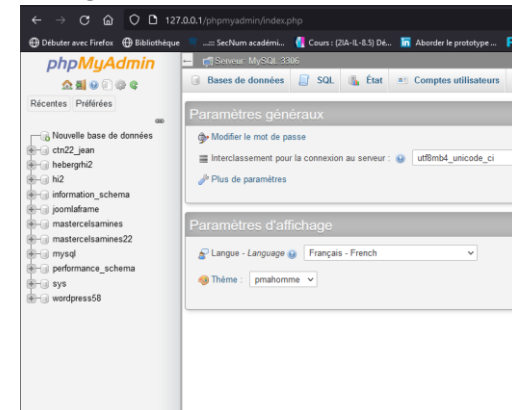
```
Marquer Invite de commandes - mysql -u root -p
c:\Developpement\wamp64\bin\mysql\mysql5.7.11\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 392
Server version: 5.7.11 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

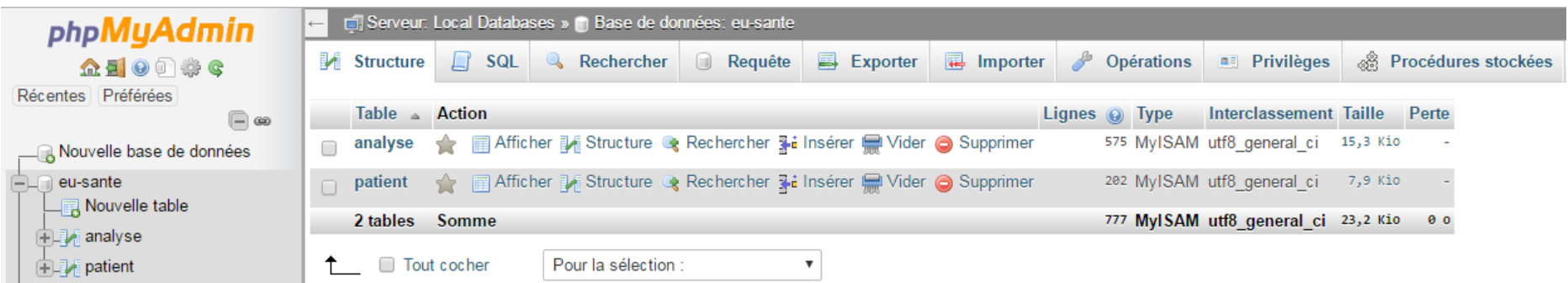
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| eu-sante |
| euromovwp |
| event |
+-----+
mysql>rgynkbd1
```





# Création de la base de données

**Mysql est un Système de Gestion de Base de Données (SGBD) relationnel manipulé en SQL pour créer des tables ou manipuler des données**



The screenshot shows the phpMyAdmin interface for a MySQL server. The database 'eu-sante' is selected, and two tables are listed: 'analyse' and 'patient'. The 'analyse' table has 575 rows, and the 'patient' table has 202 rows. The interface includes a navigation menu on the left and a toolbar at the top with options like Structure, SQL, Rechercher, Requête, Exporter, Importer, Opérations, Privilèges, and Procédures stockées.

Table	Action	Lignes	Type	Interclassement	Taille	Perte
analyse	Afficher Structure Rechercher Insérer Vider Supprimer	575	MyISAM	utf8_general_ci	15,3 Kio	-
patient	Afficher Structure Rechercher Insérer Vider Supprimer	202	MyISAM	utf8_general_ci	7,9 Kio	-
2 tables	Somme	777	MyISAM	utf8_general_ci	23,2 Kio	0 0

**La clef primaire identifie un enregistrement de manière unique**

**La clef étrangère fait la liaison entre deux tables**

**Nota: des SGBD No-Sql (Not Only SQL) existent pour stocker des données sous formes arborescences**



## Exemples de requêtes simple Search and Query Language

```
SELECT id, age FROM patient WHERE age > 50 and age < 65;
```

```
SELECT * FROM analyse order by age asc;
```

```
SELECT age FROM patient WHERE left( code_postal , 2 ) = '34'
```

```
update patient set age = 65 WHERE id = 1;
```

```
delete FROM patient WHERE age > 102;
```

```
insert into patient( age, code_postal, sexe)  
values ( 55 , '92000', 0 );
```

## Exemples de jointures en SQL

```
SELECT * FROM patient , analyse where  
patient.id = analyse.id_patient
```

→ produit cartésien : patient • analyse

```
SELECT id, age FROM patient left join analyse  
on patient.id = analyse.id_patient
```

id	age	code_postal	sexe	id_analyse	id_patient	date_analyse	glycemie
1	55	31000	0	NULL	NULL	NULL	NULL
2	65	34000	1	1	2	01/11/2016	8,4
2	65	34000	1	2	2	02/11/2016	8,5

## Exemples de groupe en SQL

```
SELECT count(*) as nombre_analyse , patient.id
FROM patient,analyse
WHERE analyse.id_patient = patient.id
and patient.age > 55 and patient.age < 65
GROUP BY analyse.id_patient
```



nombre_analyse	id_patient
4	1
2	2
5	4

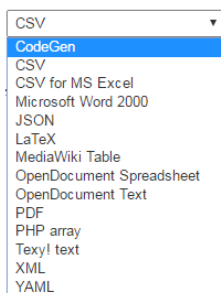
```
SELECT      age, code_postal, sexe, glycemie ,
if( glycemie >11 AND age >65 , 'O' , 'N' ) as hyperglycemie
FROM patient , analyse
WHERE patient.id = analyse.id_patient
```

## Exportation des lignes

Méthode d'exportation :

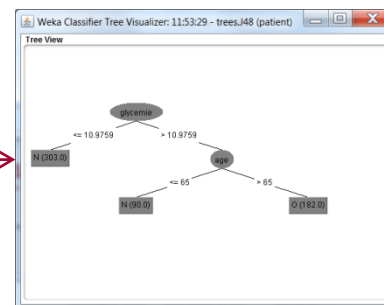
- Rapide - n'afficher qu'un minimum
- Personnalisée - afficher toutes les

Format :



```
age,code_postal,sexe,glycemie,hyperglycemie
94,91000,1,2.11294,N
56,47000,1,6.77298,N
56,3000,1,9.67017,N
87,85000,0,2.77634,N
62,3000,0,5.7189,N
56,73000,0,3.67211,N
04,53000,1,18.3,N
69,84000,0,14.6508,0
68,42000,0,13.6662,0
61,93000,0,7.80011,N
55,41000,1,5.20636,N
65,87000,0,8.71976,N
68,16000,1,13.8655,0
74,86000,1,16.7902,0
65,87000,0,14.2247,N
69,70000,0,19.1577,0
71,8000,0,15.9741,0
94,77000,0,11.0892,0
48,34000,0,6.92816,N
74,92000,0,11.8321,0
99,70000,1,2.4555,N
75,27000,1,0.165962,N
69,70000,0,1.05383,N
```

**Format CSV**



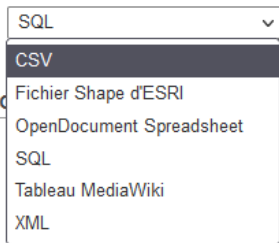
**Fouille de données**

Le SQL permet de recréer une base de données complète en rejouant un scénario

Des outils additionnels permettent des importations facile par exemple dans phpMyAdmin

Importation dans la base de données

Format :



**Format CSV**



id	nom	prenom	email	adresse	tel	valide
1	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
2	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
3	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
4	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
5	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
6	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
7	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
8	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1
9	NOEL	noel	noel@noel.com	1000	00 00 00 00 00	1

**SQL « brut »**



# Exemples d'Applications



# Projet Mogly



Démonstration :

12134

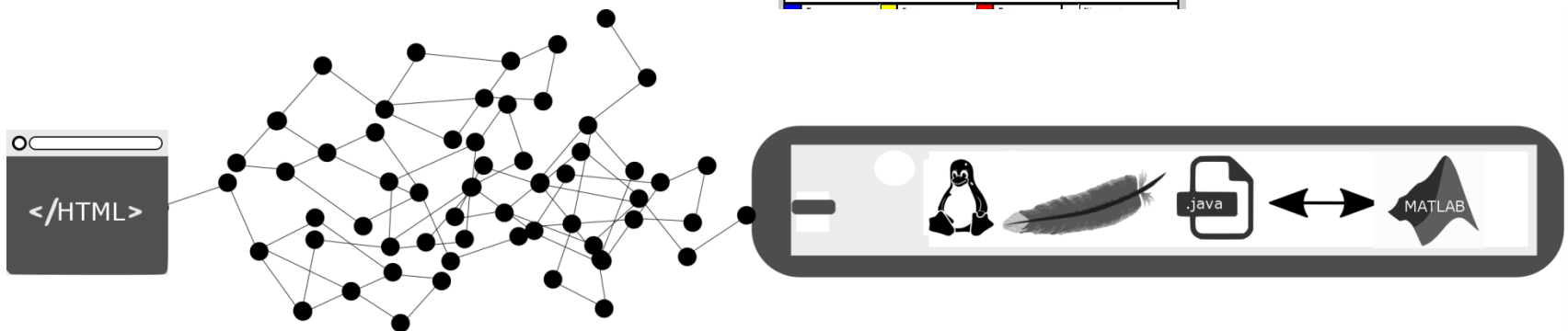
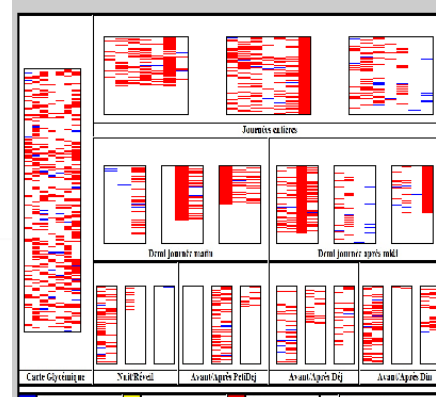
8 2 4

Commentaires

3 couleurs 5 couleurs

Voire fichier C:\fakepath\65589662.csv

Lancer le calcul >





# Détection de la malaria dans des analyses sanguines



Date                      Méthode                      Identifiant                      Version  
 Run\_2014-06-12\_13-20-19\_DIF\_5913-1.xml

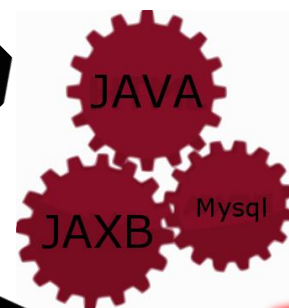
```
<xml />
</Opt>
<Items>19</Items>
<Mode>190</Mode>
<Mean>203.842102</Mean>
<StdDev>12.859337</StdDev>
</Opt>
```

```
<question>
  Qui était le premier
  empereur romain ?
</question>
<réponse>
  Auguste
</réponse>
<!-- Note : tu auras besoin
  de plus de questions. -->
</questionnaire>
```

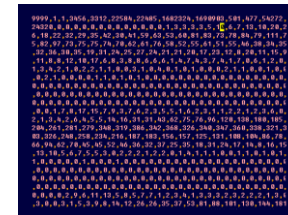
<xsd />

```
<xs:element name="Opt">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Items"/>
      <xs:element ref="Mode"/>
      <xs:element ref="Mean"/>
      <xs:element ref="StdDev"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="Items" type="xs:integer"/>
<xs:element name="Mode" type="xs:integer"/>
<xs:element name="Mean" type="xs:decimal"/>
<xs:element name="StdDev" type="xs:decimal"/>
```



Données de classifications



Rejet





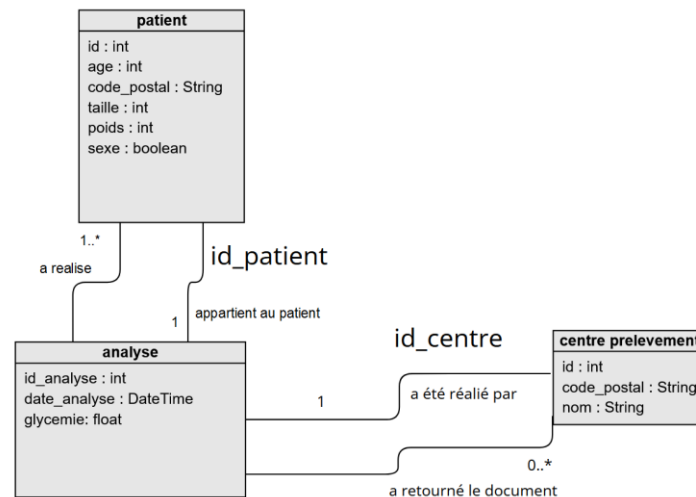
# TP



# Etapes 1 - Analyse du besoin et parfois de l'existant

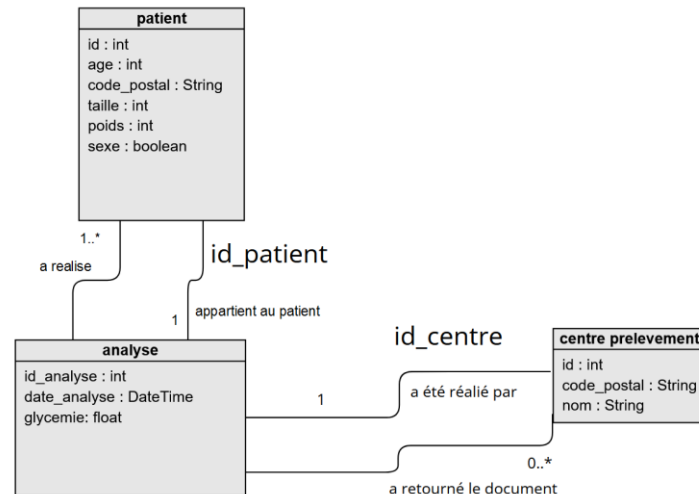
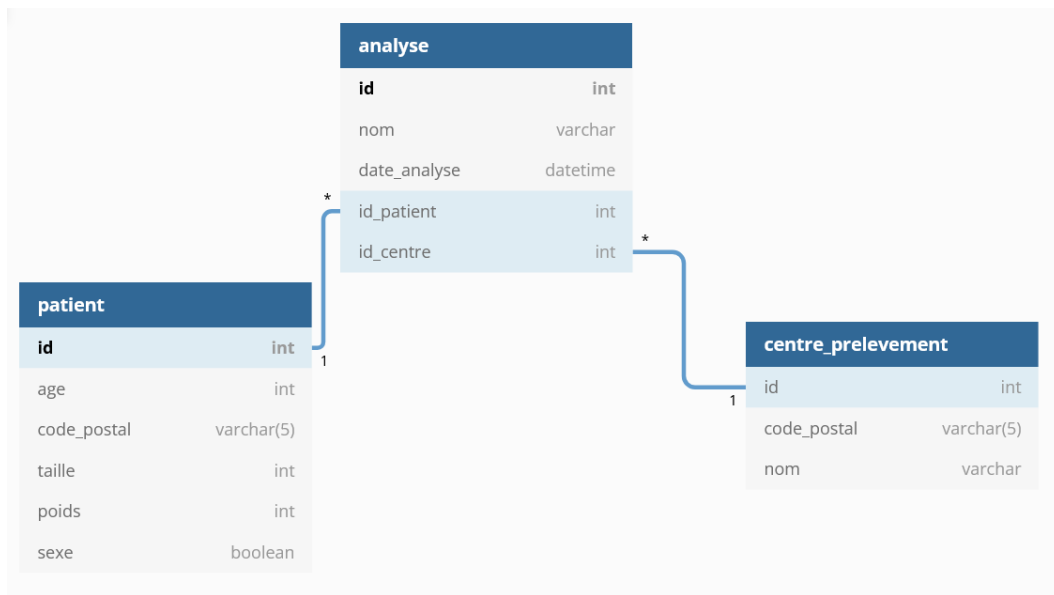
## ■ Votre sujet:

- ◆ Un patient (âge, code postal, sexe) passe des analyses de glycémie ( date du test , glycémie en mmol/L ) réalisé par un centre de prélèvement



## Etape 2 - La modélisation

### ■ Diagramme de classes/entités



Pouvez trouver les trois erreurs dans le diagramme de gauche ?

## Etape 3 – Création des tables

### ■ Création des tables

```
CREATE TABLE IF NOT EXISTS patient
(
  id int(11) NOT NULL AUTO_INCREMENT,
  age float NOT NULL,
  code_postal varchar(5) NOT NULL,
  taille int(11) NOT NULL,
  poids int(11) NOT NULL,
  sexe tinyint(1) NOT NULL,
  PRIMARY KEY ( id )
);
```

Nom de table: patient Ajouter 1 colonne(s) Exécuter

Nom	Type	Taille/Valeurs*	Valeur par défaut	Interclassement	Attributs	Null	Index	A	J
id	INT		Aucun(e)			<input type="checkbox"/>	PRIMARY	PRIMARY	<input checked="" type="checkbox"/>
age	FLOAT		Aucun(e)			<input type="checkbox"/>	---		<input type="checkbox"/>
code_postal	VARCHAR	5	Aucun(e)			<input type="checkbox"/>	---		<input type="checkbox"/>
taille	INT		Aucun(e)			<input type="checkbox"/>	---		<input type="checkbox"/>
poids	INT		Aucun(e)			<input type="checkbox"/>	---		<input type="checkbox"/>
sexe	BOOLEAN		Aucun(e)			<input type="checkbox"/>	---		<input type="checkbox"/>

Structure

Commentaires de table : Interclassement : Moteur de stockage :

## NULL OR NOT NULL pour un champ ?

## Etape 3 – Création des tables bis

### ■ Création des tables

```
CREATE TABLE IF NOT EXISTS centre_prelevement
```

```
(  
  id int(11) NOT NULL AUTO_INCREMENT,  
  nom VARCHAR(255) NOT NULL,  
  code_postal varchar(5) NOT NULL,  
  PRIMARY KEY ( id )  
);
```

```
CREATE TABLE IF NOT EXISTS analyse
```

```
(  
  id_analyse int(11) NOT NULL AUTO_INCREMENT,  
  date_analyse DATETIME,  
  glycemie float NOT NULL,  
  id_centre int(11) NOT NULL,  
  id_patient int(11) NOT NULL,  
  PRIMARY KEY ( id )
```

**Une erreur se cache dans ces requêtes SQL**

## Etape 4 – Injections des données

### ■ Création des tables

```
INSERT INTO patient
(id, age, code_postal, taille, poids, sexe)
VALUES
( NULL,49, '34090' , 179 , 68 , 0 );
```

```
INSERT INTO patient
(id, age, code_postal, taille, poids, sexe)
VALUES
( NULL,25, '34000' , 191 , 168 , 1 );
```

id	age	code_postal	taille	poids	sexe
1	31	64600	170	78	0
2	67	32700	169	49	1
3	76	35100	134	44	1
4	78	75600	197	88	0
5	73	86800	137	89	0
6	73	58600	174	117	1
7	28	39700	192	63	1
8	78	81700	131	136	0
9	75	67700	155	185	1
10	72	30700	195	120	0
11	34	53100	166	67	1
12	74	84000	177	51	0

```
1 1,31,64600,170,78,0
2 2,67,32700,169,49,1
3 3,76,35100,134,44,1
4 4,78,75600,197,88,0
5 5,73,86800,137,89,0
6 6,73,58600,174,117,1
7 7,28,39700,192,63,1
```

**Juste avant  
l'importation j'ai  
effacer la ligne des  
noms de colonnes**

Importation dans la table « patient »

Fichier à importer :

Le fichier peut être compressé (gzip, bzip2, zip) ou non.  
Le nom du fichier compressé doit se terminer par `.{format}.{compression}`

Parcourir les fichiers :  patient-rva.csv (Taille : ...)

Il est également possible de glisser-déposer un fichier sur n'importe quel élément de cette page.

Jeu de caractères du fichier :

Importation partielle :

Permettre l'interruption de l'importation si la limite de temps configurée est atteinte.

Ignorer ce nombre de requêtes (pour SQL), à partir du début :

Autres options :

Activer la vérification des clés étrangères

Format :

NB : si le fichier contient plusieurs tables, elles seront combinées en une seule.

Options spécifiques au format :

Mettre à jour les données lorsque des clés dupliquées sont trouvées.

Colonnes séparées par :

Colonnes entourées par :

Colonnes échappées avec :

Lignes terminées par :

Importer ce nombre de lignes (facultatif) :

## Construction de la base de données et des tables

Un patient (âge, code postal, sexe) passe des analyses de glycémie ( date du test , glycémie en mmol/L ) réalisé par un centre de prélèvement

id	age	code_postal	taille	poids	sexe
1	55	31000	150	75	0
2	65	34000	170	88	1

id_analyse	id_centre	id_patient	date_analyse	glycemie
1	2	2	01/11/2016	8,4
2	1	2	02/11/2016	8,5

id	nom	code_postal
1	Centre ville	34000
2	Université	34080



- **Mise à jours des enregistrements**
  - ◆ UPDATE patient SET poids=55 WHERE patient.id=4;
  - ◆ UPDATE patient SET age=age+1
- **Effacer un ou plusieurs enregistrements**
  - ◆ DELETE FROM patient WHERE patient.id=4;
  - ◆ DELETE FROM patient WHERE age < 20;
- **Insertion d'un nouvel enregistrement**
  - ◆ INSERT into patient values (NULL,19,34070,179,82,1);
  - ◆ INSERT into patient (age,poids,sexe) values (19,34070,82,1);

# EXEMPLES

```
SELECT * FROM patient WHERE sexe = 1;
SELECT * FROM patient WHERE sexe = 0;
SELECT * FROM patient WHERE age < 18;
SELECT * FROM patient WHERE age >= 18;
SELECT * FROM patient WHERE taille < 150 AND age > 52;
SELECT * FROM patient WHERE ( taille > 180 AND age > 70 ) OR ( taille < 150 AND age < 40 );
SELECT age,taille,poids FROM patient WHERE ( taille > 180 AND age > 70 ) OR ( taille < 150 AND age < 40 );
SELECT CONCAT(age, 'ans') , CONCAT( taille , 'cm') FROM patient WHERE ( taille > 180 AND age > 70 ) OR ( taille < 150 AND age < 40 );
SELECT CONCAT(age, 'ans') As 'AGE du patient' , CONCAT( taille , 'cm') As 'Taille du patient' FROM patient WHERE ( taille > 180 AND age > 70 ) OR ( taille < 150 AND age < 40 );
SELECT MAX(age) FROM patient;
SELECT MIN(age) FROM patient;
SELECT MAX(age),MIN(age) FROM patient;
SELECT COUNT(*) FROM patient WHERE sexe = 1;
SELECT COUNT(*) FROM patient WHERE sexe = 0;
SELECT COUNT(*) FROM patient;
SELECT COUNT(*),sexe FROM patient GROUP BY sexe
SELECT age-18 as 'delta age majoritee' FROM patient WHERE age >= 18;
SELECT * , (age > 60 ) as 'candidat cohorte' FROM patient WHERE age > 18 AND code_postal != '34000';
SELECT patient.id , LEFT(code_postal,2) , (age > 60 ) as 'candidat cohorte' FROM patient WHERE age > 18 AND code_postal NOT IN ( '34000' , '30000' , '13000' );
SELECT glycemie, date_analyse FROM analyse WHERE analyse.id_patient = 2 order by date_analyse asc;
SELECT glycemie FROM analyse WHERE analyse.id_patient = 2 order by date_analyse asc;
SELECT glycemie, date_analyse FROM analyse WHERE analyse. id_patient = 2 order by date_analyse desc;
SELECT patient.id, glycemie, age FROM analyse, patient WHERE analyse.id_patient = patient.id and age >= 49;
SELECT patient.id, MAX(glycemie) FROM analyse, patient WHERE analyse.id_patient = patient.id and age >= 49 GROUP BY patient.id;
CREATE TABLE max_glycemie SELECT patient.id, MAX(glycemie) FROM analyse, patient WHERE analyse.id_patient = patient.id and age >= 49 GROUP BY patient.id ;
SELECT DATE_FORMAT(date_analyse,'%y-%m-%d') as jour_analyse FROM analyse;
SELECT DATE_FORMAT(date_analyse,'%y-%m-%d')as jour_analyse, count(*) FROM analyse GROUP BY jour_analyse;
SELECT DATE_FORMAT(date_analyse,'%y-%m-%d')as jour_analyse, count(*) as nombre FROM analyse GROUP BY jour_analyse;
SELECT DATE_FORMAT(date_analyse,'%y-%m-%d')as jour_analyse, count(*) as nombre FROM analyse GROUP BY jour_analyse order by nombre DESC;
SELECT * FROM analyse LIMIT 0,10;
SELECT * FROM patient WHERE patient.code_postal IN ( SELECT code_postal FROM centre_prelevement WHERE nom = 'centre ville');
SELECT * FROM patient WHERE LEFT(patient.code_postal,2) IN ( SELECT LEFT(code_postal,2) FROM centre_prelevement);
SELECT * from patient LEFT JOIN (SELECT id_patient as patient4 ,count(*) as nombre FROM analyse group by analyse.id_patient) AS sousrequete ON patient.id = sousrequete.patient4 where nombre = 4;
Création d'une nouvelle colonne pour centre_prelevement avec le département dans cette colonne...
```



## Partie additionnelle – Application Web et Weka

- Placer ce fichier dans le dossier wamp64\www\ de votre ordinateur transférer ce code en modifiant les premières lignes avec vos réglages de BD

```
<?php
$host  = "localhost";
$user  = "root";
$pass  = " si mot de passe sinon ne rien mettre";
$db_name = "";

//create connection
$connection = mysqli_connect($host, $user, $pass, $db_name);
//get results FROM database
$result = mysqli_query($connection, "SELECT * FROM patient");
while ($row = mysqli_fetch_assoc($result)) {
    echo ($row['id'] );
    echo ( ',' );
    echo ($row['age'] );
    echo ( '<br>' );
}
```

# Utilisation de Weka pour construire un arbre de décision

```

age,code_postal,sexe,glycemie,hyperglycemie
48,34000,0,6.92816,N
48,34000,0,16.4608,N
48,34000,0,0.292341,N
48,34000,0,5.6599,N
73,24000,1,14.43,0
73,24000,1,11.5988,0
71,53000,0,7.06819,N
71,53000,0,4.72441,N
71,53000,0,5.75819,N
71,53000,0,6.33936,N
72,2000,1,7.5945,N
    
```

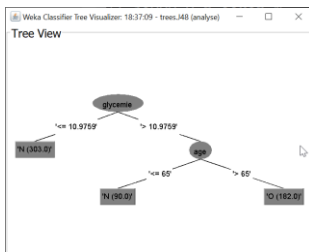
Weka Explorer - Preprocess Classify Cluster Associate Select attributes Visualize

Current relation: analyse  
Instances: 575  
Attributes: 5

Selected attribute:  
Name: hyperglycemie Type: Nominal  
Missin... 0 (0%... Distinct: ... Unique: 0 (0%)

No.	Label	Count
1	N	393
2	O	182

Class: hyperglycemie (Nom) Visualize All



Weka Explorer - Preprocess Classify Cluster Associate Select attributes Visualize

Classifier: Choose **J48 - C 0.25 - M 2**

Test options:  
 Use training set  
 Supplied test set  
 Cross-validation Folds: 10  
 Percentage split %: 66

Classifier output:

gamma statistic	0.996
Mean absolute error	0.0017
Root mean squared error	0.0417
Relative absolute error	0.4017 %
Root relative squared error	0.9459 %
Total number of instances	575

Detailed Accuracy By Class

	TP Rate	FP Rate	Precision	Recall	F-Measure
1	0.997	0	1	0.997	0.999
2	1	0.003	0.995	1	0.997
Weighted Avg.	0.998	0.001	0.998	0.998	0.998

Confusion Matrix

	a	b	classified as
a	392	1	a = N
b	1	181	b = O

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier: Choose

- weka
- Classifiers
- bayes
- functions
- lazy
- meta
- mi
- misc
- rules
- trees
  - ADTree
  - BFTree
  - DecisionStump
  - FT
  - J48**
  - J48draft
  - LADTree
  - LMT
  - NBTree
  - RandomForest
  - RandomTree
  - REPtree
  - SimpleCart



# Evolution avec Not Only SQL

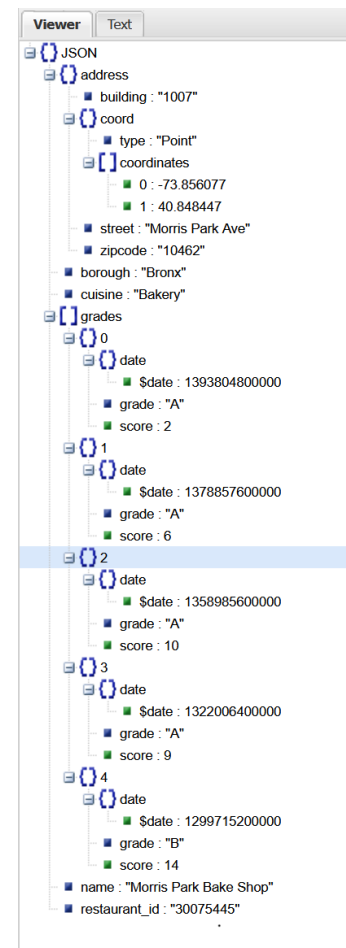


## Utilisation d'un formalisme JSON

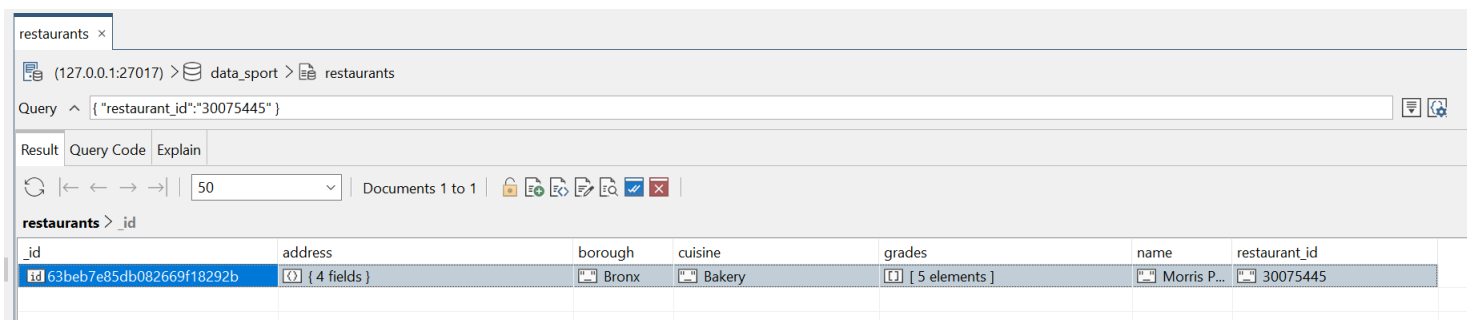
```
{
  "address": {
    "building": "1007",
    "coord": {
      "type": "Point",
      "coordinates": [-73.856077, 40.848447]
    },
    "street": "Morris Park Ave",
    "zipcode": "10462",
    "borough": "Bronx",
    "cuisine": "Bakery",
    "grades": [
      {
        "date": {
          "$date": "1393804800000"
        },
        "grade": "A",
        "score": 2
      },
      {
        "date": {
          "$date": "1378857600000"
        },
        "grade": "A",
        "score": 6
      },
      {
        "date": {
          "$date": "1358985600000"
        },
        "grade": "A",
        "score": 10
      },
      {
        "date": {
          "$date": "1322006400000"
        },
        "grade": "A",
        "score": 9
      },
      {
        "date": {
          "$date": "1299715200000"
        },
        "grade": "B",
        "score": 14
      }
    ],
    "name": "Morris Park Bake Shop",
    "restaurant_id": "30075445"
  }
}
```

Adapté pour les données en arborescence faiblement structurée

Exploitation dans les structures de données pour éviter des jointures coûteuses en temps



# Query with JSON



The screenshot shows a MongoDB query interface. The query is `{ "restaurant_id": "30075445" }`. The result is a table with the following columns: `_id`, `address`, `borough`, `cuisine`, `grades`, `name`, and `restaurant_id`. The first row contains the following values: `63beb7e85db082669f18292b`, `{ 4 fields }`, `Bronx`, `Bakery`, `[ 5 elements ]`, `Morris P...`, and `30075445`.

_id	address	borough	cuisine	grades	name	restaurant_id
63beb7e85db082669f18292b	{ 4 fields }	Bronx	Bakery	[ 5 elements ]	Morris P...	30075445

```
{ "restaurant_id": { $gt : "30075445" } }
```

```
{ "grades": { $elemMatch: { "grade": "A" } } }
```

```
{ "grades": { $elemMatch: { "grade": { $gt: "B" } } } }
```

```
{ "address.zipcode": { $gte: "11369" , $lte: "11372" } }
```