

3.9 Rechercher des données

Extra options

←T→

▼

id

NomActeur

PrenomActeur

DateNaissanceActeur

PaysActeur

<input type="checkbox"/>	Edit	Copy	Delete	3	Cage	Nicolas	7 janvier 1964	USA
<input type="checkbox"/>	Edit	Copy	Delete	8	Diaz	Cameron	30 aout 1972	USA
<input type="checkbox"/>	Edit	Copy	Delete	9	DiCaprio	Leonardo	11 novembre 1974	USA
<input type="checkbox"/>	Edit	Copy	Delete	10	Fox	Megan	16 mai 1986	USA
<input type="checkbox"/>	Edit	Copy	Delete	12	Johansson	Scarlett	22 novembre 1984	USA
<input type="checkbox"/>	Edit	Copy	Delete	13	LaBeouf	Shia	11 juin 1986	USA
<input type="checkbox"/>	Edit	Copy	Delete	14	MacDowell	Andie	21 avril 1958	USA
<input type="checkbox"/>	Edit	Copy	Delete	16	Marsden	James	18 septembre 1973	USA
<input type="checkbox"/>	Edit	Copy	Delete	17	Murray	Bill	21 septembre 1950	USA
<input type="checkbox"/>	Edit	Copy	Delete	20	Worthington	Sam	2 aout 1976	USA

←T→

▼

id

NomActeur

PrenomActeur

DateNaissanceActeur

PaysActeur

<input type="checkbox"/>	Edit	Copy	Delete	6	Cotillard	Marion	30 septembre 1975	France
<input type="checkbox"/>	Edit	Copy	Delete	16	Marsden	James	18 septembre 1973	USA
<input type="checkbox"/>	Edit	Copy	Delete	17	Murray	Bill	21 septembre 1950	USA

☐ Check all With selected: Edit Copy Delete Export

←T→

▼

id

NomActeur

PrenomActeur

DateNaissanceActeur

PaysActeur

←T→

▼

id

NomActeur

PrenomActeur

DateNaissanceActeur

PaysActeur

☐ Check all With selected: Edit Copy Delete Export

←

→

▼

id

TitreFilm


NomActeur


PrenomActeur


RoleFilm

SalaireFilm

☐

 Edit

 Copy

 Delete

8

Jennifers body

Fox

Megan

Jennifer Check

11000

←

→

▼

id

TitreFilm


NomActeur


PrenomActeur


RoleFilm

SalaireFilm

☐

 Edit

 Copy

 Delete

9

Le prestige

Bale

Christian


Alfred Borden


1700


↑


☐ Check all

With selected:

 Edit

 Copy

 Delete

 Export

←

→

▼

id

TitreFilm


NomActeur


PrenomActeur


RoleFilm

SalaireFilm

☐

 Edit

 Copy

 Delete

1

Equilibrium

Bale

Christian

John Preston

760

←

→

▼

id

TitreFilm


NomActeur


PrenomActeur


RoleFilm

SalaireFilm

☐

 Edit

 Copy

 Delete

4

Il etait une fois

Marsden

James

Le prince Edouard

500

←

→

▼

id

TitreFilm


NomActeur


PrenomActeur


RoleFilm

SalaireFilm

☐

 Edit

 Copy

 Delete

32

Un jour sans fin

Murray

Bill


Phil Connors


600


↑


☐ Check all

With selected:




 Edit







 Copy










 Delete

 Export



















3.10 Modification de table










<input type="checkbox"/>				19	Sherlock Holmes	Policier	2009	NULL	NULL	NULL
--------------------------	---	---	---	----	-----------------	----------	------	------	------	------

<input type="checkbox"/>				21	Downey	Robert	4 avril 1965	USA
<input type="checkbox"/>				22	Law	Jude	29 décembre 1972	Angleterre

<input type="checkbox"/>				33	Xmen origins : Wolverine	Jackman	Hugh	Logan	5000
<input type="checkbox"/>				34	Sherlock Holmes	Robert	Downey	Sherlock Holmes	2500
<input type="checkbox"/>				35	Sherlock Holmes	Jude	Law	John Watson	2000

3.11 Créer une table

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 id	int			No	None		AUTO_INCREMENT	  
<input type="checkbox"/>	2 nom	varchar(20)	utf8mb4_unicode_ci		No	None			  
<input type="checkbox"/>	3 codepostal	int			No	None			  
<input type="checkbox"/>	4 film	varchar(64)	utf8mb4_unicode_ci		No	None			  
<input type="checkbox"/>	5 datedebut	date			No	None			  
<input type="checkbox"/>	6 datefin	date			No	None			  


☐ Check all
 With selected:
 








				id	nom	codepostal	film	datedebut	datefin
<input type="checkbox"/>				1	ugc	92520	Equilibrium	2019-04-01	2019-04-09
<input type="checkbox"/>				2	gaumont	92700	Il etait une fois	2018-04-01	2018-04-09
<input type="checkbox"/>				3	gaumont	92700	Terminator renaissance	2019-02-01	2019-02-09

☐ Check all With selected: Edit Copy Delete Export

3.12 Créer un utilisateur

	User name	Host name	Password	Global privileges	Grant	Action
<input type="checkbox"/>	mysql.infoschema	localhost	Yes	SELECT	No	Edit privileges Export Unlock
<input type="checkbox"/>	mysql.session	localhost	Yes	SHUTDOWN, SUPER	No	Edit privileges Export Unlock
<input type="checkbox"/>	mysql.sys	localhost	Yes	USAGE	No	Edit privileges Export Unlock
<input type="checkbox"/>	readOnly	localhost	No	SELECT	No	Edit privileges Export Lock
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES	Yes	Edit privileges Export Lock

☐ Check all With selected: Export

4.3 Regarder la structure générale de la base de données : show

```

mysql> mysql show databases;
+-----+
| Database |
+-----+
| exosql   |
| information_schema |
| mysql    |
| performance_schema |
| sys      |
+-----+
5 rows in set (0.00 sec)

mysql> mysql show tables from exosql;
+-----+
| Tables_in_exosql |
+-----+
| acteurs           |
| cinema            |
| film              |
| realisateur       |
| role              |
+-----+
5 rows in set (0.00 sec)

mysql> mysql show columns from acteurs;
+----+
| Field |
+----+
| id     |
| NomActeur |
| PrenomActeur |
| DateNaissanceActeur |
| PaysActeur |
+----+
5 rows in set (0.00 sec)

mysql> mysql show create table exosql.acteurs;
+-----+
| Table | Create Table |
+-----+
| acteurs | CREATE TABLE `acteurs` (
  `id` int NOT NULL AUTO_INCREMENT,
  `NomActeur` char(32) NOT NULL DEFAULT '',
  `PrenomActeur` char(32) NOT NULL DEFAULT '',
  `DateNaissanceActeur` varchar(32) DEFAULT NULL,
  `PaysActeur` varchar(16) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=23 DEFAULT CHARSET=latin1 |
+-----+
1 row in set (0.00 sec)
  
```

4.4 Regarder le contenu de la base de données : select

ERROR 1054 (42S22): Unknown column 'USA' in 'where clause'

```
mysql> select * from acteurs where PaysActeur='USA';
```

id	NomActeur	PrenomActeur	DateNaissanceActeur	PaysActeur
3	Cage	Nicolas	7 janvier 1964	USA
8	Diaz	Cameron	30 aout 1972	USA
9	DiCaprio	Leonardo	11 novembre 1974	USA
10	Fox	Megan	16 mai 1986	USA
12	Johansson	Scarlett	22 novembre 1984	USA
13	LaBeouf	Shia	11 juin 1986	USA
14	MacDowell	Andie	21 avril 1958	USA
16	Marsden	James	18 septembre 1973	USA
17	Murray	Bill	21 septembre 1950	USA
20	Worthington	Sam	2 aout 1976	USA
21	Downey	Robert	4 avril 1965	USA

11 rows in set (0.00 sec)

```
mysql> select * from acteurs where PaysActeur='USA' or PaysActeur='France';
```

id	NomActeur	PrenomActeur	DateNaissanceActeur	PaysActeur
3	Cage	Nicolas	7 janvier 1964	USA
6	Cotillard	Marion	30 septembre 1975	France
7	Depardieu	Gerard	27 decembre 1948	France
8	Diaz	Cameron	30 aout 1972	USA
9	DiCaprio	Leonardo	11 novembre 1974	USA
10	Fox	Megan	16 mai 1986	USA
12	Johansson	Scarlett	22 novembre 1984	USA
13	LaBeouf	Shia	11 juin 1986	USA
14	MacDowell	Andie	21 avril 1958	USA
16	Marsden	James	18 septembre 1973	USA
17	Murray	Bill	21 septembre 1950	USA
19	Wilson	Lambert	3 aout 1958	France
20	Worthington	Sam	2 aout 1976	USA
21	Downey	Robert	4 avril 1965	USA

14 rows in set (0.00 sec)

```
mysql> select * from acteurs where DateNaissanceActeur LIKE '%septembre%';
```

id	NomActeur	PrenomActeur	DateNaissanceActeur	PaysActeur
6	Cotillard	Marion	30 septembre 1975	France
16	Marsden	James	18 septembre 1973	USA
17	Murray	Bill	21 septembre 1950	USA

3 rows in set (0.00 sec)

```
mysql> select * from acteurs where DateNaissanceActeur not like '%septembre%';
```

id	NomActeur	PrenomActeur	DateNaissanceActeur	PaysActeur
1	Bale	Christian	30 janvier 1974	Angleterre
2	Bean	Sean	17 avril 1959	Angleterre
3	Cage	Nicolas	7 janvier 1964	USA
4	Caine	Michael	14 mars 1933	Angleterre
5	Connery	Sean	25 aout 1930	Angleterre
7	Depardieu	Gerard	27 decembre 1948	France
8	Diaz	Cameron	30 aout 1972	USA
9	DiCaprio	Leonardo	11 novembre 1974	USA
10	Fox	Megan	16 mai 1986	USA
11	Jackman	Hugh	12 octobre 1968	Australie
12	Johansson	Scarlett	22 novembre 1984	USA
13	LaBeouf	Shia	11 juin 1986	USA
14	MacDowell	Andie	21 avril 1958	USA
15	MacGregor	Ewan	31 mars 1971	Angleterre
18	Prime	Optimus	Inconnue	Cybertron
19	Wilson	Lambert	3 aout 1958	France
20	Worthington	Sam	2 aout 1976	USA
21	Downey	Robert	4 avril 1965	USA
22	Law	Jude	29 decembre 1972	Angleterre

```
19 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM acteurs WHERE PaysActeur='USA' AND DateNaissanceActeur like '%1950%';
```

id	NomActeur	PrenomActeur	DateNaissanceActeur	PaysActeur
17	Murray	Bill	21 septembre 1950	USA

```
1 row in set (0.00 sec)
```

```
mysql> select SalaireFilm from role where NomActeur='Fox' and TitreFilm='Jennifers body';
```

SalaireFilm
11000

```
1 row in set (0.00 sec)
```

```
mysql> select SalaireFilm from role where NomActeur='Bale' and TitreFilm='Le prestige';
```

SalaireFilm
1700

For the right syntax to use near '1000' at line 1

```
mysql> select * from role where SalaireFilm between 400 and 800;
```

id	TitreFilm	NomActeur	PrenomActeur	RoleFilm	SalaireFilm
1	Equilibrium	Bale	Christian	John Preston	760
4	Il etait une fois	Marsden	James	Le prince Edouard	500
32	Un jour sans fin	Murray	Bill	Phil Connors	600

```
3 rows in set (0.00 sec)
```

```
mysql> select sum(SalaireFilm) from role group by(TitreFilm);
```

sum(SalaireFilm)
820
15000
500
20925
11000
4800
3501
10000
7500
21000
4000
3050
20000
26000
850
5000
4500

```
17 rows in set (0.00 sec)
```

```
mysql> select TitreFilm, sum(SalaireFilm) from role group by(TitreFilm);
```

TitreFilm	sum(SalaireFilm)
Equilibrium	820
Ghost rider	15000
Il etait une fois	500
Inception	20925
Jennifers body	11000
Le prestige	4800
Lhomme au masque de fer	3501
Matrix reloaded	10000
Rock	7500
Terminator renaissance	21000
The Box	4000
The island	3050
Transformers	20000
Transformers 2 : la revanche	26000
Un jour sans fin	850
Xmen origins : Wolverine	5000
Sherlock Holmes	4500

```
17 rows in set (0.00 sec)
```

```
mysql> select TitreFilm, sum(SalaireFilm) as TotalSalaire from role group by TitreFilm order by TotalSalaire;
```

TitreFilm	TotalSalaire
Il etait une fois	500
Equilibrium	820
Un jour sans fin	850
The island	3050
Lhomme au masque de fer	3501
The Box	4000
Sherlock Holmes	4500
Le prestige	4800
Xmen origins : Wolverine	5000
Rock	7500
Matrix reloaded	10000
Jennifers body	11000
Ghost rider	15000
Transformers	20000
Inception	20925
Terminator renaissance	21000
Transformers 2 : la revanche	26000

```
17 rows in set (0.00 sec)
```

```

mysql> select * from role order by SalaireFilm;
+-----+-----+-----+-----+-----+-----+
| id | TitreFilm | NomActeur | PrenomActeur | RoleFilm | SalaireFilm |
+-----+-----+-----+-----+-----+-----+
| 30 | Transformers 2 : la revanche | Prime | Optimus | Lui-meme | 0 |
| 27 | Transformers | Prime | Optimus | Lui-meme | 0 |
| 13 | L'homme au masque de fer | Depardieu | Gerard | Porthos | 1 |
| 2 | Equilibrium | Bean | Sean | Partridge | 60 |
| 5 | Inception | Caine | Michael | Miles | 75 |
| 22 | The island | Bean | Sean | Merrick | 150 |
| 31 | Un jour sans fin | MacDowell | Andie | Rita | 250 |
| 10 | Le prestige | Caine | Michael | Cutter | 350 |
| 4 | Il etait une fois | Marsden | James | Le prince Edouard | 500 |
| 32 | Un jour sans fin | Murray | Bill | Phil Connors | 600 |
| 1 | Equilibrium | Bale | Christian | John Preston | 760 |
| 6 | Inception | Cottillard | Marion | Mall | 850 |
| 23 | The island | Johansson | Scarlett | Jordan 2delta | 900 |
| 12 | Le prestige | Johansson | Scarlett | Olivia | 950 |
| 21 | The Box | Marsden | James | Arthur Lewis | 1000 |
| 16 | Rock | Cage | Nicolas | Stanley Goodspeed | 1500 |
| 9 | Le prestige | Bale | Christian | Alfred Borden | 1700 |
| 11 | Le prestige | Jackman | Hugh | Robert Angier | 1800 |
| 24 | The island | MacGregor | Ewan | Lincoln 6echo | 2000 |
| 35 | Sherlock Holmes | Jude | Law | John Watson | 2000 |
| 34 | Sherlock Holmes | Robert | Downey | Sherlock Holmes | 2500 |
| 20 | The Box | Diaz | Cameron | Norma Lewis | 3000 |
| 14 | L'homme au masque de fer | DiCaprio | Leonardo | Louis XIV | 3500 |
| 33 | Xmen origins : Wolverine | Jackman | Hugh | Logan | 5000 |
| 17 | Rock | Connery | Sean | John Patrick Mason | 6000 |
| 25 | Transformers | Fox | Megan | Mikaela Banes | 7000 |
| 19 | Terminator renaissance | Worthington | Sam | Marcus Wright | 9000 |
| 15 | Matrix reloaded | Wilson | Lambert | Le Merovingien | 10000 |
| 8 | Jennifers body | Fox | Megan | Jennifer Check | 11000 |
| 28 | Transformers 2 : la revanche | Fox | Megan | Mikaela Banes | 12000 |
| 18 | Terminator renaissance | Bale | Christian | John Connor | 12000 |
| 26 | Transformers | LaBeouf | Shia | Sam Witwicky | 13000 |
| 29 | Transformers 2 : la revanche | LaBeouf | Shia | Sam Witwicky | 14000 |
| 3 | Ghost rider | Cage | Nicolas | Johnny Blaze | 15000 |
| 7 | Inception | DiCaprio | Leonardo | Dom Cobb | 20000 |
+-----+-----+-----+-----+-----+-----+
35 rows in set (0.00 sec)

```

4.5 Assembler/joindre plusieurs tables


```
mysql> SELECT f.TitreFilm, r.NomAuteur, f.NomRealisateur FROM role r INNER JOIN film f ON r.TitreFilm = f.TitreFilm;
```

TitreFilm	NomAuteur	NomRealisateur
Equilibrium	Bale	Wimmer
Equilibrium	Bean	Wimmer
Ghost rider	Cage	Johnson
Il etait une fois	Marsden	Lima
Inception	Caine	Nolan
Inception	Cottillard	Nolan
Inception	DiCaprio	Nolan
Jennifers body	Fox	Kusama
Le prestige	Bale	Nolan
Le prestige	Caine	Nolan
Le prestige	Jackman	Nolan
Le prestige	Johansson	Nolan
Lhomme au masque de fer	Depardieu	Wallace
Lhomme au masque de fer	DiCaprio	Wallace
Matrix reloaded	Wilson	Wachowski
Rock	Cage	Bay
Rock	Connery	Bay
Terminator renaissance	Bale	MacGinty
Terminator renaissance	Worthington	MacGinty
The Box	Diaz	Kelly
The Box	Marsden	Kelly
The island	Bean	Bay
The island	Johansson	Bay
The island	MacGregor	Bay
Transformers	Fox	Bay
Transformers	LaBeouf	Bay
Transformers	Prime	Bay
Transformers 2 : la revanche	Fox	Bay
Transformers 2 : la revanche	LaBeouf	Bay
Transformers 2 : la revanche	Prime	Bay
Un jour sans fin	MacDowell	Ramis
Un jour sans fin	Murray	Ramis
Xmen origins : Wolverine	Jackman	Hood
Sherlock Holmes	Robert	NULL
Sherlock Holmes	Jude	NULL

```
35 rows in set (0.00 sec)
```

```
mysql> use exosql;
```

```
Database changed
```

```
mysql> select f.TitreFilm, a.NomAuteur, a.PaysAuteur, f.NomRealisateur from acteurs a inner join role r on a.NomAuteur= r.NomAuteur inner join film f on r.TitreFilm = f.TitreFilm;
```

TitreFilm	NomAuteur	PaysAuteur	NomRealisateur
Terminator renaissance	Bale	Angleterre	MacGinty
Le prestige	Bale	Angleterre	Nolan
Equilibrium	Bale	Angleterre	Wimmer
The island	Bean	Angleterre	Bay
Equilibrium	Bean	Angleterre	Wimmer
Rock	Cage	USA	Bay
Ghost rider	Cage	USA	Johnson
Le prestige	Caine	Angleterre	Nolan
Inception	Caine	Angleterre	Nolan
Rock	Connery	Angleterre	Bay
Lhomme au masque de fer	Depardieu	France	Wallace
The Box	Diaz	USA	Kelly
Lhomme au masque de fer	DiCaprio	USA	Wallace
Inception	DiCaprio	USA	Nolan
Transformers 2 : la revanche	Fox	USA	Bay
Transformers	Fox	USA	Bay
Jennifers body	Fox	USA	Kusama
Xmen origins : Wolverine	Jackman	Australie	Hood
Le prestige	Jackman	Australie	Nolan
The island	Johansson	USA	Bay
Le prestige	Johansson	USA	Nolan
Transformers 2 : la revanche	LaBeouf	USA	Bay
Transformers	LaBeouf	USA	Bay
Un jour sans fin	MacDowell	USA	Ramis
The island	MacGregor	Angleterre	Bay
The Box	Marsden	USA	Kelly
Il etait une fois	Marsden	USA	Lima
Un jour sans fin	Murray	USA	Ramis
Transformers 2 : la revanche	Prime	Cybertron	Bay
Transformers	Prime	Cybertron	Bay
Matrix reloaded	Wilson	France	Wachowski
Terminator renaissance	Worthington	USA	MacGinty

```
32 rows in set (0.00 sec)
```

4.7 Contenu du fichier source acteursFilm.sql

```
mysql> source C:/Users/Valentin/Documents/imie_cours/explotationBD_valentin_massonniere/acteursFilm.sql
Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

5.1 Chargement de la base

```
mysql> source C:/Users/Valentin/Documents/imie_cours/explotationBD_valentin_massonniere/employees.sql
Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 1 row affected (0.01 sec)

Database changed
+-----+
| INFO |
+-----+
+-----+
| CREATING DATABASE STRUCTURE |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected, 6 warnings (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

5.2 Analyser le contenu

```
mysql> show tables;
+-----+
| Tables_in_employees |
+-----+
| current_dept_emp     |
| departments          |
| dept_emp             |
| dept_emp_latest_date |
| dept_manager         |
| employees            |
| salaries             |
| titles               |
+-----+
8 rows in set (0.00 sec)

mysql>
```

```
mysql> select * from departments;
+-----+-----+
| dept_no | dept_name |
+-----+-----+
| d009    | Customer Service |
| d005    | Development      |
| d002    | Finance          |
| d003    | Human Resources  |
| d001    | Marketing        |
| d004    | Production       |
| d006    | Quality Management |
| d008    | Research         |
| d007    | Sales            |
+-----+-----+
9 rows in set (0.00 sec)
```

5.3 Réaliser des requêtes

```
mysql> select last_name, year(hire_date) as hire_year from employees where hire_date = (select max(hire_date) from employees);
+-----+-----+
| last_name | hire_year |
+-----+-----+
| Haddadi   | 1999      |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select last_name,min(year(hire_date)) from employees;
+-----+-----+
| last_name | min(year(hire_date)) |
+-----+-----+
| Facello   | 1985                |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select last_name, year(hire_date) as hire_year from employees where hire_date = (select min(hire_date) from employees);
+-----+-----+
| last_name | hire_year |
+-----+-----+
| Markovitch | 1985      |
| Alpin      | 1985      |
| Ossenbruggen | 1985     |
| Wegerle    | 1985      |
| Haggimont  | 1985      |
| Onuegbe    | 1985      |
+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> select last_name, year(hire_date) as hire_year from employees where hire_date = (select max(hire_date) from employees where gender='F');
+-----+-----+
| last_name | hire_year |
+-----+-----+
| Pettey    | 1997      |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select last_name, year(birth_date) as birth_date from employees where birth_date = (select min(birth_date) from employees where gender='F');
+-----+-----+
| last_name | birth_date |
+-----+-----+
| Peac      | 1952       |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select last_name, year(hire_date) as hire_year from employees where hire_date = (select max(hire_date) from employees where gender='M');
+-----+-----+
| last_name | hire_year |
+-----+-----+
| Haddadi   | 1999      |
+-----+-----+
1 row in set (0.00 sec)

mysql> _
```

```
mysql> select e.first_name, e.last_name from employees e inner join dept_manager m on e.emp_no = m.emp_no;
+-----+-----+
| first_name | last_name |
+-----+-----+
| Oscar      | Ghazalie  |
+-----+-----+
1 row in set (0.00 sec)

mysql> _
```

```
mysql> select * from employees where emp_no = 10001 union select * from employees where emp_no = 10003;
+-----+-----+-----+-----+-----+-----+
| emp_no | birth_date | first_name | last_name | gender | hire_date |
+-----+-----+-----+-----+-----+-----+
| 10001  | 1953-09-02 | Georgi     | Facello   | M      | 1986-06-26 |
| 10003  | 1959-12-03 | Parto      | Bamford   | M      | 1986-08-28 |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

```
mysql> select emp_no, salary from salaries where emp_no='10001' order by salary asc;
```

emp_no	salary
10001	60117
10001	62102
10001	66074
10001	66596
10001	66961
10001	71046
10001	74333
10001	75286
10001	75994
10001	76884
10001	80013
10001	81025
10001	81097
10001	84917
10001	85097
10001	85112
10001	88958

```
17 rows in set (0.00 sec)
```

```
mysql> select s.emp_no, s.salary, e.last_name, e.first_name from salaries s inner join employees e on s.emp_no = e.emp_no where s.emp_no='10001' order by salary asc;
```

emp_no	salary	last_name	first_name
10001	60117	Facello	Georgi
10001	62102	Facello	Georgi
10001	66074	Facello	Georgi
10001	66596	Facello	Georgi
10001	66961	Facello	Georgi
10001	71046	Facello	Georgi
10001	74333	Facello	Georgi
10001	75286	Facello	Georgi
10001	75994	Facello	Georgi
10001	76884	Facello	Georgi
10001	80013	Facello	Georgi
10001	81025	Facello	Georgi
10001	81097	Facello	Georgi
10001	84917	Facello	Georgi
10001	85097	Facello	Georgi
10001	85112	Facello	Georgi
10001	88958	Facello	Georgi

```
17 rows in set (0.00 sec)
```

```
mysql> select e.first_name, e.last_name, s.last_salary from employees e inner join (select emp_no, max(salary) as last_salary from salaries where salary <> 9999 group by emp_no) s on e.emp_no = s.emp_no order by s.last_salary desc;
```

first_name	last_name	last_salary
Georgi	Facello	88958
Berzel	Simmel	72527
Christian	Koblick	52119
Parto	Ramford	43699

```
4 rows in set (0.00 sec)
```

```
mysql> █
```

```
mysql> select e.first_name, e.last_name, t.title from titles t inner join employees e on t.emp_no = e.emp_no where t.to_date like '%9999%' order by e.first_name asc;
```

first_name	last_name	title
Adamantios	Portugali	Senior Staff
Alain	Chappellet	Senior Engineer
Alejandro	McAlpine	Senior Staff
Alejandro	Brender	Senior Staff
Anneke	Preusig	Senior Engineer
Anoosh	Peyn	Senior Engineer
Basil	Tramer	Senior Staff
Berhard	McFarlin	Senior Staff
Berni	Genin	Engineer
Berzel	Simmel	Staff
Bojan	Montemayor	Engineer
Breanna	Billingsley	Senior Staff
Brendon	Bernini	Senior Engineer
Christian	Koblick	Senior Engineer
Claudi	Stavenow	Senior Engineer
Cristinel	Bouloucos	Senior Staff
Divier	Reistad	Senior Engineer
Duangkaew	Piveteau	Engineer
Ebbe	Callaway	Senior Engineer
Eberhardt	Terkki	Senior Staff
Elvis	Demeyer	Senior Engineer
Georgi	Facello	Senior Engineer
Gino	Leonhardt	Senior Engineer
Heping	Nitsch	Senior Staff
Hidefumi	Caine	Senior Engineer
Huan	Lortz	Senior Staff
Jeong	Reistad	Senior Engineer
Karsten	Joslin	Senior Engineer
Kazuhide	Peha	Senior Engineer
Kazuhiro	Cappellletti	Staff
Kwee	Schusler	Senior Engineer
Kyoichi	Maliniak	Senior Staff
Lillian	Haddadi	Staff
Lucien	Rosenbaum	Senior Staff
Mayuko	Warwick	Engineer
Mayumi	Schueler	Senior Staff
Mingsen	Casley	Technique Leader
Moss	Shanbhogue	Engineer
Otmar	Herbst	Senior Engineer
Parto	Ramford	Senior Engineer

5.4.1 Créer une procédure

```

explotationBD_valentin_massonniere > procedure.sql
1 delimiter $
2 drop procedure if exists ageMinMax$
3 create procedure ageMinMax()
4 begin
5     select floor(datediff(now(), str_to_date(max(birth_date), '%Y-%m-%d')) / 365) as min_age, floor(datediff(now(), str_to_date(min(birth_date), '%Y-%m-%d')) / 365) as
6     max_age from employees;
7 end$
8 delimiter ;
9 call ageMinMax();

```

```

mysql> source C:/Users/Valentin/Documents/imie_cours/explotationBD_valentin_massonniere/procedure.sql
Query OK, 0 rows affected (0.01 sec)

```

```

Query OK, 0 rows affected (0.00 sec)

```

```

+-----+-----+
| min_age | max_age |
+-----+-----+
|      60 |      72 |
+-----+-----+
1 row in set (0.00 sec)

```

```

explotationBD_valentin_massonniere > procedure.sql
1 delimiter $
2 drop procedure if exists ageMinMax$
3 create procedure ageMinMax()
4 begin
5     select datediff(now(), min(birth_date)) as min_age, datediff(now(), max(birth_date)) as max_age from employees;
6 end$
7 delimiter ;
8 call ageMinMax();
9
10 delimiter $
11 drop procedure if exists salaireMaxi$
12 create procedure salaireMaxi(in gender char(1))
13 begin
14     select e.first_name, e.last_name, max(s.salary) as max_salary from employees e inner join salaries s on e.emp_no = s.emp_no where e.gender = gender group by e.
15     first_name, e.last_name;
16 end$
17 delimiter ;
18 call salaireMaxi('F');
19 call salaireMaxi('M');

```

```

+-----+-----+-----+
| first_name | last_name | max_salary |
+-----+-----+-----+
| Bezalel    | Simmel    |      72527 |
+-----+-----+-----+
1 row in set (0.00 sec)

```

```

Query OK, 0 rows affected (0.00 sec)

```

```

+-----+-----+-----+
| first_name | last_name | max_salary |
+-----+-----+-----+
| Georgi     | Facello   |      88958 |
| Parto      | Bamford   |      43699 |
| Chirstian   | Koblick   |      52119 |
+-----+-----+-----+
3 rows in set (0.00 sec)

```

```

Query OK, 0 rows affected (0.01 sec)

```

```

19
20 delimiter $
21 drop procedure if exists salaireMaxiOut$
22 create procedure salaireMaxiOut(in gender char(1), out salaryOut decimal(10,2))
23 begin
24     select subquery.max_salary into salaryOut
25     from (
26         select max(s.salary) as max_salary
27         from employees e
28         inner join salaries s on e.emp_no = s.emp_no
29         where e.gender = gender
30     ) as subquery;
31 end$
32 delimiter ;
33
34 call salaireMaxiOut('F', @salaryOut);
35 select @salaryOut;
36 call salaireMaxiOut('M', @salaryOut);
37 select @salaryOut;

```

```

+-----+
| @salaryOut |
+-----+
| 72527.00 |
+-----+
1 row in set (0.00 sec)

Query OK, 1 row affected (0.00 sec)

+-----+
| @salaryOut |
+-----+
| 88958.00 |
+-----+
1 row in set (0.00 sec)

mysql>

```

5.4.2 Créer un algorithme dans une procédures

```
8 delimiter $
9 drop procedure if exists salaireMaxi2$
1 create procedure salaireMaxi2(in gender char(1))
2 begin
3     declare maxSalary decimal(10,2);
4     declare genderOut varchar(50);
5
6     select max(s.salary) into maxSalary from employees e inner join salaries s on e.emp_no = s.emp_no where e.gender = gender;
7
8     if gender = 'M' then
9         set genderOut = 'pour les hommes';
10    elseif gender = 'F' then
11        set genderOut = 'pour les femmes';
12    else
13        set genderOut = 'genre non specifie';
14    end if;
15
16    select concat('Le salaire maximum est ', maxSalary, ' ', genderOut) as result;
17 end$
18 delimiter ;
19 call salaireMaxi2('F');
20 call salaireMaxi2('M');
```

```
+-----+
| result |
+-----+
| Le salaire maximum est 72527.00 pour les femmes |
+-----+
1 row in set (0.00 sec)
```

Query OK, 0 rows affected (0.00 sec)

```
+-----+
| result |
+-----+
| Le salaire maximum est 88958.00 pour les hommes |
+-----+
1 row in set (0.00 sec)
```

Query OK, 0 rows affected (0.00 sec)