

Document highlight:

- SELECT statement
- SELECT + WHERE statement
 - Logical operators (AND /OR)
 - Wildcards ("% and "_")
- SELECT + WHERE statement
 - Group By and Order By
 - Aggregate functions (ie., AVG, MIN, MAX, COUNT)
- SELECT + WHERE + HAVING statement
- LIMIT and ALIASING in MySQL

1. Setup MySQL Workbench (community)

Download: and install (follow installation guide):

<https://dev.mysql.com/downloads/installer/>

The screenshot shows the MySQL Community Downloads page. The browser address bar displays "dev.mysql.com/downloads/installer/". The page title is "MySQL Community Downloads" with a breadcrumb "MySQL Installer". There are tabs for "General Availability (GA) Releases", "Archives", and a download icon. The main section is titled "MySQL Installer 8.0.40". A note states: "Note: MySQL 8.0 is the final series with MySQL Installer. As of MySQL 8.1, use a MySQL product's MSI or Zip archive for installation. MySQL Server 8.1 and higher also bundle MySQL Configurator, a tool that helps configure MySQL Server." Below the note are two dropdown menus: "Select Version:" with "8.0.40" selected, and "Select Operating System:" with "Microsoft Windows" selected. There are two download options for Windows (x86, 32-bit), MSI Installer. The first is a 2.1M file with MD5: 42da0dc06ad328fe2451eeb3998fb016 and a link to the signature. The second is a 306.4M file with MD5: 8c1bf3a205d5e191e36dc334a10f55d2 and a link to the signature. A footer note suggests using MD5 checksums and GnuPG signatures to verify the integrity of the packages.

General Availability (GA) Releases Archives

MySQL Installer 8.0.40

Note: MySQL 8.0 is the final series with MySQL Installer. As of MySQL 8.1, use a MySQL product's MSI or Zip archive for installation. MySQL Server 8.1 and higher also bundle MySQL Configurator, a tool that helps configure MySQL Server.

Select Version:
8.0.40

Select Operating System:
Microsoft Windows

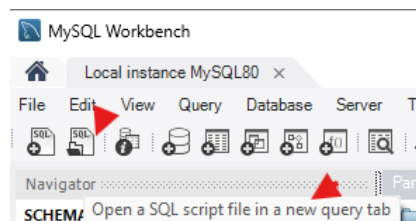
Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.40.0.msi)	8.0.40	2.1M	Download
		MD5: 42da0dc06ad328fe2451eeb3998fb016	Signature
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.40.0.msi)	8.0.40	306.4M	Download
		MD5: 8c1bf3a205d5e191e36dc334a10f55d2	Signature

Note: We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.

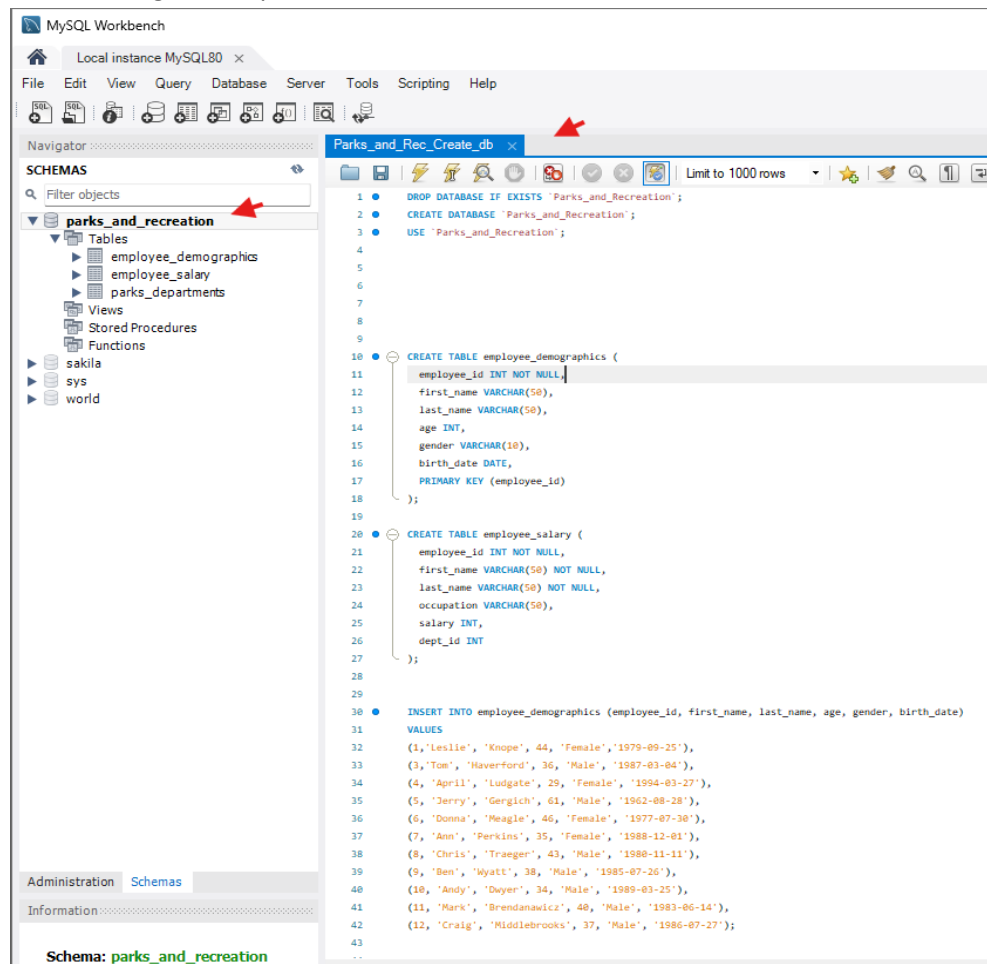
MySQL home screen



2. Create a database (create tables and load the set of data into it)



After running the script:



3. SELECT statement (flavors of queries and its sampling output)

Navigator: SCHEMAS Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```
1 #This is selecting all value(s) to display from a table
2 # --Query 01
3 • SELECT
4 *
5 FROM
6 parks_and_recreation.employee_demographics
7 ;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap C

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knobe	44	Female	1979-09-25
3	Tom	Haverford	36	Male	1987-03-04
4	April	Ludgate	29	Female	1994-03-27
5	Jerry	Gergich	61	Male	1962-08-28
6	Donna	Meagle	46	Female	1977-07-30
7	Ann	Perkins	35	Female	1988-12-01
8	Chris	Traeger	43	Male	1980-11-11
9	Ben	Wyatt	38	Male	1985-07-26
10	Andy	Dwyer	34	Male	1989-03-25
11	Mark	Brendanawicz	40	Male	1983-06-14
12	Craig	Middlebrooks	37	Male	1986-07-27
NULL	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```
9 #This is selecting specific column value(s) to display from a table
10 # --Query 01
11 • SELECT
12 first_name
13 FROM
14 parks_and_recreation.employee_demographics
15 ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: I A

first_name
Leslie
Tom
April
Jerry
Donna
Ann
Chris
Ben
Andy
Mark
Craig

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```

16 # --Query 02
17 • SELECT
18     first_name,
19     last_name,
20     birth_date
21 FROM
22     parks_and_recreation.employee_demographics
23 ;

```

Result Grid

first_name	last_name	birth_date
▶ Leslie	Knope	1979-09-25
Tom	Haverford	1987-03-04
April	Ludgate	1994-03-27
Jerry	Gergich	1962-08-28
Donna	Meagle	1977-07-30
Ann	Perkins	1988-12-01
Chris	Traeger	1980-11-11
Ben	Wyatt	1985-07-26
Andy	Dwyer	1989-03-25
Mark	Brendanawicz	1983-06-14
Craig	Middlebrooks	1986-07-27

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```

25 #This is selecting specific column value(s) to display from a table, with calculation (that follows the PEMDAS order/rule)
26 # --Query 01
27 • SELECT
28     first_name,
29     age,
30     (age + 10) * 10 + 10
31 FROM
32     parks_and_recreation.employee_demographics
33 ;
--

```

Result Grid

first_name	age	(age + 10) * 10 + 10
▶ Leslie	44	550
Tom	36	470
April	29	400
Jerry	61	720
Donna	46	570
Ann	35	460
Chris	43	540
Ben	38	490
Andy	34	450
Mark	40	510
Craig	37	480

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```

34
35 #This is selecting specific and distinct column value(s) to display from a table
36 # --Query 01
37 • SELECT DISTINCT
38     gender
39 FROM
40     parks_and_recreation.employee_demographics
41 ;

```

Result Grid

gender
▶ Female
Male

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

01-SELECT Statement

```

42 # --Query 02
43 • SELECT DISTINCT
44     first_name,
45     gender
46 FROM
47     parks_and_recreation.employee_demographics
48 ;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

first_name	gender
Leslie	Female
Tom	Male
April	Female
Jerry	Male
Donna	Female
Ann	Female
Chris	Male
Ben	Male
Andy	Male
Mark	Male
Craig	Male

4. SELECT + WHERE statement (flavors of queries, logical operators (AND /OR), wildcards ("% and "_"), and its sampling output)

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments

02-WHERE Statement (logical o...

```

1 #This is selecting all value(s) to display from a table, using:
2 # (a) WHERE to filter the output
3 # (b) AND OR NOT logical operators
4 # (c) LIKE condition + "%" and "_" wildcards
5

```

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

```

1  #This is selecting all value(s) to display from a table
2  # --Query 01
3  • SELECT
4      *
5  FROM
6      parks_and_recreation.employee_demographics
7  ;
8

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	1	Leslie	Knope	44	Female	1979-09-25
	3	Tom	Haverford	36	Male	1987-03-04
	4	April	Ludgate	29	Female	1994-03-27
	5	Jerry	Gergich	61	Male	1962-08-28
	6	Donna	Meagle	46	Female	1977-07-30
	7	Ann	Perkins	35	Female	1988-12-01
	8	Chris	Traeger	43	Male	1980-11-11
	9	Ben	Wyatt	38	Male	1985-07-26
	10	Andy	Dwyer	34	Male	1989-03-25
	11	Mark	Brendanawicz	40	Male	1983-06-14
	12	Craig	Middlebrooks	37	Male	1986-07-27
*	NULL	NULL	NULL	NULL	NULL	NULL

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

```

8
9  #This is selecting all value(s) to display from a table
10 # --Query 01
11 • SELECT
12     *
13 FROM
14     parks_and_recreation.employee_salary
15 ;

```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: |

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
	2	Ron	Swanson	Director of Parks and Recreation	70000	1
	3	Tom	Haverford	Entrepreneur	50000	1
	4	April	Ludgate	Assistant to the Director of Parks and Recreation	25000	1
	5	Jerry	Gergich	Office Manager	50000	1
	6	Donna	Meagle	Office Manager	60000	1
	7	Ann	Perkins	Nurse	55000	4
	8	Chris	Traeger	City Manager	90000	3
	9	Ben	Wyatt	State Auditor	70000	6
	10	Andy	Dwyer	Shoe Shiner and Musician	20000	NULL
	11	Mark	Brendana...	City Planner	57000	3
	12	Craig	Middlebro...	Parks Director	65000	1

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o...

```

6  # --Query 01 (sampling of specific first name)
7  • SELECT
8      *
9  FROM
10     parks_and_recreation.employee_salary
11 WHERE
12     first_name = 'Leslie'
13 ;
14

```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: |

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1

Navigator: SCHEMAS Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

15 # --Query 02 (sampling of greater than)
16 • SELECT
17 *
18 FROM
19     parks_and_recreation.employee_salary
20 WHERE
21     salary > 50000
22 ;
23

```

Result Grid Filter Rows: Export: Wrap Cell Content:

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
	2	Ron	Swanson	Director of Parks and Recreation	70000	1
	6	Donna	Meagle	Office Manager	60000	1
	7	Ann	Perkins	Nurse	55000	4
	8	Chris	Traeger	City Manager	90000	3
	9	Ben	Wyatt	State Auditor	70000	6
	11	Mark	Brendanawicz	City Planner	57000	3
	12	Craig	Middlebrooks	Parks Director	65000	1

Navigator: SCHEMAS Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

24 # --Query 03 (sampling of greater than or equal to)
25 • SELECT
26 *
27 FROM
28     parks_and_recreation.employee_salary
29 WHERE
30     salary >= 50000
31 ;
32

```

Result Grid Filter Rows: Export: Wrap Cell Content:

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
	2	Ron	Swanson	Director of Parks and Recreation	70000	1
	3	Tom	Haverford	Entrepreneur	50000	1
	5	Jerry	Gergich	Office Manager	50000	1
	6	Donna	Meagle	Office Manager	60000	1
	7	Ann	Perkins	Nurse	55000	4
	8	Chris	Traeger	City Manager	90000	3
	9	Ben	Wyatt	State Auditor	70000	6
	11	Mark	Brendanawicz	City Planner	57000	3
	12	Craig	Middlebrooks	Parks Director	65000	1

Navigator: SCHEMAS Filter objects

- ▼ parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

33 # --Query 04 (sampling of less than)
34 • SELECT
35 *
36 FROM
37     parks_and_recreation.employee_salary
38 WHERE
39     salary < 50000
40 ;
41

```

Result Grid Filter Rows: Export: Wrap Cell Content:

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	4	April	Ludgate	Assistant to the Director of Parks and Recreation	25000	1
	10	Andy	Dwyer	Shoe Shiner and Musician	20000	NULL

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x)

Limit to 1000 rows

```
60 # --Query 07 (sampling of greater than + AND)
61 • SELECT
62 *
63 FROM
64     parks_and_recreation.employee_demographics
65 WHERE
66     birth_date > '1985-01-01'
67 AND
68     gender = 'Male'
69 ;
```

Result Grid

	employee_id	first_name	last_name	age	gender	birth_date
▶	3	Tom	Haverford	36	Male	1987-03-04
	9	Ben	Wyatt	38	Male	1985-07-26
	10	Andy	Dwyer	34	Male	1989-03-25
	12	Craig	Middlebrooks	37	Male	1986-07-27
*	NULL	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x)

Limit to 1000 rows

```
71 # --Query 08 (sampling of greater than + OR)
72 • SELECT
73 *
74 FROM
75     parks_and_recreation.employee_demographics
76 WHERE
77     birth_date > '1985-01-01'
78 OR
79     gender = 'Male'
80 ;
```

Result Grid

	employee_id	first_name	last_name	age	gender	birth_date
▶	3	Tom	Haverford	36	Male	1987-03-04
	9	Ben	Wyatt	38	Male	1985-07-26
	10	Andy	Dwyer	34	Male	1989-03-25
	12	Craig	Middlebrooks	37	Male	1986-07-27
*	NULL	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x)

Limit to 1000 rows

```

82 # --Query 09 (sampling of AND + OR, following the PEMDAS order/rule)
83 • SELECT
84     *
85 FROM
86     parks_and_recreation.employee_demographics
87 WHERE
88     (first_name = 'Leslie'
89 AND
90     age = 44)
91 OR age > 55
92 ;
93

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knope	44	Female	1979-09-25
5	Jerry	Gergich	61	Male	1962-08-28
*	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x)

Limit to 1000 rows

```

94 # --Query 10 (sampling of LIKE condition, leveraging the wildcards)
95 • SELECT
96     *
97 FROM
98     parks_and_recreation.employee_demographics
99 WHERE
100    first_name LIKE 'Jer%'
101 ;
102

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
5	Jerry	Gergich	61	Male	1962-08-28
*	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x)

Limit to 1000 rows

```

103 # --Query 11 (sampling of LIKE condition, leveraging the wildcards)
104 • SELECT
105     *
106 FROM
107     parks_and_recreation.employee_demographics
108 WHERE
109    first_name LIKE '%om%'
110 ;
111

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
3	Tom	Haverford	36	Male	1987-03-04
*	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

112 # --Query 12 (sampling of LIKE condition, leveraging the wildcards)
113 • SELECT
114     *
115 FROM
116     parks_and_recreation.employee_demographics
117 WHERE
118     first_name LIKE '%o_%'
119 ;
120

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
3	Tom	Haverford	36	Male	1987-03-04
6	Donna	Meagle	46	Female	1977-07-30
NULL	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

121 # --Query 13 (sampling of LIKE condition, leveraging the wildcards)
122 • SELECT
123     *
124 FROM
125     parks_and_recreation.employee_demographics
126 WHERE
127     first_name LIKE '%o_'
128 ;
129

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
3	Tom	Haverford	36	Male	1987-03-04
NULL	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

130 # --Query 14 (sampling of LIKE condition, leveraging the wildcards)
131 • SELECT
132     *
133 FROM
134     parks_and_recreation.employee_demographics
135 WHERE
136     birth_date LIKE '1989%'
137 ;
138

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
10	Andy	Dwyer	34	Male	1989-03-25
NULL	NULL	NULL	NULL	NULL	NULL

Navigator

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

02-WHERE Statement (logical o... x

Limit to 1000 rows

```

139 # --Query 15 (sampling of LIKE condition, leveraging the wildcards)
140 • SELECT
141     *
142 FROM
143     parks_and_recreation.employee_demographics
144 WHERE
145     birth_date LIKE '197%'
146 ;
147

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knope	44	Female	1979-09-25
6	Donna	Meagle	46	Female	1977-07-30
NULL	NULL	NULL	NULL	NULL	NULL

5. SELECT + WHERE statement (flavors of queries, Group By, Order By, aggregate functions (ie., AVG, MIN, MAX, COUNT), and its sampling output)

Navigator

03-WHERE Statement (group by... x)

Limit to 1000 rows

```

1  #This is selecting all value(s) to display from a table, using:
2  # (a) WHERE to filter the output
3  # (b) GROUP BY and ORDER BY
4  # (c) Aggregate functions (ie., AVG, MAX, MIN, COUNT)
5

```

SCHEMAS

Filter objects

▼ parks_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments

SCHEMAS

Filter objects

▼ parks_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments
- Views
- Stored Procedures
- Functions

▼ sakila

▼ sys

▼ world

Limit to 1000 rows

```

1  #This is selecting all value(s) to display from a table
2  # --Query 01
3  • SELECT
4      *
5  FROM
6      parks_and_recreation.employee_demographics
7  ;
8

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knope	44	Female	1979-09-25
3	Tom	Haverford	36	Male	1987-03-04
4	April	Ludgate	29	Female	1994-03-27
5	Jerry	Gergich	61	Male	1962-08-28
6	Donna	Meagle	46	Female	1977-07-30
7	Ann	Perkins	35	Female	1988-12-01
8	Chris	Traeger	43	Male	1980-11-11
9	Ben	Wyatt	38	Male	1985-07-26
10	Andy	Dwyer	34	Male	1989-03-25
11	Mark	Brendanawicz	40	Male	1983-06-14
12	Craig	Middlebrooks	37	Male	1986-07-27
NULL	NULL	NULL	NULL	NULL	NULL

SCHEMAS

Filter objects

▼ parks_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments
- Views
- Stored Procedures
- Functions

▼ sakila

▼ sys

▼ world

Limit to 1000 rows

```

8
9  #This is selecting all value(s) to display from a table
10 # --Query 01
11 • SELECT
12     *
13 FROM
14     parks_and_recreation.employee_salary
15 ;

```

Result Grid

employee_id	first_name	last_name	occupation	salary	dept_id
1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
2	Ron	Swanson	Director of Parks and Recreation	70000	1
3	Tom	Haverford	Entrepreneur	50000	1
4	April	Ludgate	Assistant to the Director of Parks and Recreation	25000	1
5	Jerry	Gergich	Office Manager	50000	1
6	Donna	Meagle	Office Manager	60000	1
7	Ann	Perkins	Nurse	55000	4
8	Chris	Traeger	City Manager	90000	3
9	Ben	Wyatt	State Auditor	70000	6
10	Andy	Dwyer	Shoe Shiner and Musician	20000	NULL
11	Mark	Brendana...	City Planner	57000	3
12	Craig	Middlebro...	Parks Director	65000	1

Navigator: 03-WHERE Statement (group by...)

SCHEMAS

Filter objects

park_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments
- Views
- Stored Procedures
- Functions

sakila

sys

world

```
6 # --Query 01 (sampling of group by)
7 • SELECT
8     gender
9 FROM
10    employee_demographics
11 GROUP BY
12    gender
13 ;
14
```

Result Grid

gender
Female
Male

Navigator: 03-WHERE Statement (group by...)

SCHEMAS

Filter objects

park_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments
- Views
- Stored Procedures
- Functions

sakila

sys

world

```
15 # --Query 02 (sampling of group by with aggregate function)
16 • SELECT
17     gender AS Gender,
18     AVG(age) AS Average_Age
19 FROM
20    employee_demographics
21 GROUP BY
22    gender
23 ;
```

Result Grid

Gender	Average_Age
Female	38.5000
Male	41.2857

Navigator: 03-WHERE Statement (group by...)

SCHEMAS

Filter objects

park_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - parks_departments
- Views
- Stored Procedures
- Functions

sakila

sys

world

```
25 # --Query 03 (sampling of group by with aggregate function)
26 • SELECT
27     gender AS Gender,
28     AVG(age) AS Average_Age,
29     MAX(age) AS Maximum_Age,
30     MIN(age) AS Minimum_Age,
31     COUNT(age) AS Number_of_People
32 FROM
33    employee_demographics
34 GROUP BY
35    gender
36 ;
```

Result Grid

Gender	Average_Age	Maximum_Age	Minimum_Age	Number_of_People
Female	38.5000	46	29	4
Male	41.2857	61	34	7

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

03-WHERE Statement (group by...)

```

38  # --Query 04 (sampling of order by)
39  •  SELECT
40      *
41  FROM
42      employee_demographics
43  ORDER BY
44      first_name
45  ;
46

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	10	Andy	Dwyer	34	Male	1989-03-25
	7	Ann	Perkins	35	Female	1988-12-01
	4	April	Ludgate	29	Female	1994-03-27
	9	Ben	Wyatt	38	Male	1985-07-26
	8	Chris	Traeger	43	Male	1980-11-11
	12	Craig	Middlebrooks	37	Male	1986-07-27
	6	Donna	Meagle	46	Female	1977-07-30
	5	Jerry	Gergich	61	Male	1962-08-28
	1	Leslie	Knope	44	Female	1979-09-25
	11	Mark	Brendanawicz	40	Male	1983-06-14
	3	Tom	Haverford	36	Male	1987-03-04
*	NULL	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

03-WHERE Statement (group by...)

```

47  # --Query 05 (sampling of order by)
48  •  SELECT
49      *
50  FROM
51      employee_demographics
52  ORDER BY
53      first_name DESC
54  ;
55

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	3	Tom	Haverford	36	Male	1987-03-04
	11	Mark	Brendanawicz	40	Male	1983-06-14
	1	Leslie	Knope	44	Female	1979-09-25
	5	Jerry	Gergich	61	Male	1962-08-28
	6	Donna	Meagle	46	Female	1977-07-30
	12	Craig	Middlebrooks	37	Male	1986-07-27
	8	Chris	Traeger	43	Male	1980-11-11
	9	Ben	Wyatt	38	Male	1985-07-26
	4	April	Ludgate	29	Female	1994-03-27
	7	Ann	Perkins	35	Female	1988-12-01
	10	Andy	Dwyer	34	Male	1989-03-25
*	NULL	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- sakila
- sys
- world

03-WHERE Statement (group by...)

```

56 # --Query 06 (sampling of order by)
57 • SELECT
58     *
59 FROM
60     employee_demographics
61 ORDER BY
62     gender,
63     age
64 ;

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	4	April	Ludgate	29	Female	1994-03-27
	7	Ann	Perkins	35	Female	1988-12-01
	1	Leslie	Knope	44	Female	1979-09-25
	6	Donna	Meagle	46	Female	1977-07-30
	10	Andy	Dwyer	34	Male	1989-03-25
	3	Tom	Haverford	36	Male	1987-03-04
	12	Craig	Middlebrooks	37	Male	1986-07-27
	9	Ben	Wyatt	38	Male	1985-07-26
	11	Mark	Brendanawicz	40	Male	1983-06-14
	8	Chris	Traeger	43	Male	1980-11-11
	5	Jerry	Gergich	61	Male	1962-08-28
*	NULL	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- parks_and_recreation
 - Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- sakila
- sys
- world

03-WHERE Statement (group by...)

```

66 # --Query 07 (sampling of order by)
67 • SELECT
68     *
69 FROM
70     employee_demographics
71 ORDER BY
72     gender,
73     age DESC
74 ;

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	6	Donna	Meagle	46	Female	1977-07-30
	1	Leslie	Knope	44	Female	1979-09-25
	7	Ann	Perkins	35	Female	1988-12-01
	4	April	Ludgate	29	Female	1994-03-27
	5	Jerry	Gergich	61	Male	1962-08-28
	8	Chris	Traeger	43	Male	1980-11-11
	11	Mark	Brendanawicz	40	Male	1983-06-14
	9	Ben	Wyatt	38	Male	1985-07-26
	12	Craig	Middlebrooks	37	Male	1986-07-27
	3	Tom	Haverford	36	Male	1987-03-04
	10	Andy	Dwyer	34	Male	1989-03-25
*	NULL	NULL	NULL	NULL	NULL	NULL

6. SELECT + WHERE + HAVING statement (flavors of queries and its sampling output)

Navigator: 04-HAVING versus WHERE Sta... x

Limit to 1000 rows

```

1 #This is selecting all value(s) to display from a table, using:
2 # (a) WHERE to filter the output
3 # (b) HAVING to filter the output
4
5

```

SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments

Limit to 1000 rows

```

1 #This is selecting all value(s) to display from a table
2 # --Query 01
3 • SELECT
4 *
5 FROM
6 parks_and_recreation.employee_demographics
7 ;
8

```

Result Grid | Filter Rows: | Edit: | Export/Import:

	employee_id	first_name	last_name	age	gender	birth_date
▶	1	Leslie	Knope	44	Female	1979-09-25
	3	Tom	Haverford	36	Male	1987-03-04
	4	April	Ludgate	29	Female	1994-03-27
	5	Jerry	Gergich	61	Male	1962-08-28
	6	Donna	Meagle	46	Female	1977-07-30
	7	Ann	Perkins	35	Female	1988-12-01
	8	Chris	Traeger	43	Male	1980-11-11
	9	Ben	Wyatt	38	Male	1985-07-26
	10	Andy	Dwyer	34	Male	1989-03-25
	11	Mark	Brendanawicz	40	Male	1983-06-14
	12	Craig	Middlebrooks	37	Male	1986-07-27
*	NULL	NULL	NULL	NULL	NULL	NULL

Limit to 1000 rows

```

8
9 #This is selecting all value(s) to display from a table
10 # --Query 01
11 • SELECT
12 *
13 FROM
14 parks_and_recreation.employee_salary
15 ;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
	2	Ron	Swanson	Director of Parks and Recreation	70000	1
	3	Tom	Haverford	Entrepreneur	50000	1
	4	April	Ludgate	Assistant to the Director of Parks and Recreation	25000	1
	5	Jerry	Gergich	Office Manager	50000	1
	6	Donna	Meagle	Office Manager	60000	1
	7	Ann	Perkins	Nurse	55000	4
	8	Chris	Traeger	City Manager	90000	3
	9	Ben	Wyatt	State Auditor	70000	6
	10	Andy	Dwyer	Shoe Shiner and Musician	20000	NULL
	11	Mark	Brendana...	City Planner	57000	3
	12	Craig	Middlebro...	Parks Director	65000	1

Navigator: 04-HAVING versus WHERE Sta... x

Limit to 1000 rows

4

5 # --Query 01 (sampling of where VS having)

6 • SELECT

7 gender,

8 AVG(age) as AVERAGE_AGE

9 FROM

10 parks_and_recreation.employee_demographics

11 GROUP BY

12 gender

13 HAVING

14 AVG(age) > 40

15 ;

Result Grid

gender	AVERAGE_AGE
Male	41.2857

Navigator: 04-HAVING versus WHERE Sta... x

Limit to 1000 rows

17 # --Query 02 (sampling of where VS having)

18 • SELECT

19 occupation,

20 AVG(salary) AS AVERAGE_SALARY

21 FROM

22 parks_and_recreation.employee_salary

23 WHERE

24 occupation LIKE '%mana%'

25 GROUP BY

26 occupation

27 ;

28

Result Grid

occupation	AVERAGE_SALARY
Office Manager	55000.0000
City Manager	90000.0000

Navigator: 04-HAVING versus WHERE Sta... x

Limit to 1000 rows

29 # --Query 03 (sampling of where VS having)

30 • SELECT

31 occupation,

32 AVG(salary) AS AVERAGE_SALARY

33 FROM

34 parks_and_recreation.employee_salary

35 WHERE

36 occupation LIKE '%mana%'

37 GROUP BY

38 occupation

39 HAVING AVG(salary) > 75000

40 ;

Result Grid

occupation	AVERAGE_SALARY
City Manager	90000.0000

7. LIMIT and ALIASING in MySQL (flavors of queries and its sampling output)

Navigator

05-Limit and Aliasing in SQL*

Limit to 1000 rows

```
1 #This is selecting all value(s) to display from a table, using:
2 # (a) Limit in SQL
3 # (b) Aliasing in SQL
4
5
```

SCHEMAS

Filter objects

park_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - park_departments
- Views
- Stored Procedures
- Functions

sakila

sys

world

Limit to 1000 rows

```
1 #This is selecting all value(s) to display from a table
2 # --Query 01
3 • SELECT
4 *
5 FROM
6 park_and_recreation.employee_demographics
7 ;
8
```

Result Grid

	employee_id	first_name	last_name	age	gender	birth_date
▶	1	Leslie	Knope	44	Female	1979-09-25
	3	Tom	Haverford	36	Male	1987-03-04
	4	April	Ludgate	29	Female	1994-03-27
	5	Jerry	Gergich	61	Male	1962-08-28
	6	Donna	Meagle	46	Female	1977-07-30
	7	Ann	Perkins	35	Female	1988-12-01
	8	Chris	Traeger	43	Male	1980-11-11
	9	Ben	Wyatt	38	Male	1985-07-26
	10	Andy	Dwyer	34	Male	1989-03-25
	11	Mark	Brendanawicz	40	Male	1983-06-14
	12	Craig	Middlebrooks	37	Male	1986-07-27
*	NULL	NULL	NULL	NULL	NULL	NULL

SCHEMAS

Filter objects

park_and_recreation

- Tables
 - employee_demographics
 - employee_salary
 - park_departments
- Views
- Stored Procedures
- Functions

sakila

sys

world

Limit to 1000 rows

```
8
9 #This is selecting all value(s) to display from a table
10 # --Query 01
11 • SELECT
12 *
13 FROM
14 park_and_recreation.employee_salary
15 ;
```

Result Grid

	employee_id	first_name	last_name	occupation	salary	dept_id
▶	1	Leslie	Knope	Deputy Director of Parks and Recreation	75000	1
	2	Ron	Swanson	Director of Parks and Recreation	70000	1
	3	Tom	Haverford	Entrepreneur	50000	1
	4	April	Ludgate	Assistant to the Director of Parks and Recreation	25000	1
	5	Jerry	Gergich	Office Manager	50000	1
	6	Donna	Meagle	Office Manager	60000	1
	7	Ann	Perkins	Nurse	55000	4
	8	Chris	Traeger	City Manager	90000	3
	9	Ben	Wyatt	State Auditor	70000	6
	10	Andy	Dwyer	Shoe Shiner and Musician	20000	NULL
	11	Mark	Brendana...	City Planner	57000	3
	12	Craig	Middlebro...	Parks Director	65000	1

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

05-Limit and Aliasing in SQL

Limit to 1000 rows

```

5  # --Query 01 (sampling of limit in SQL)
6  • SELECT
7      *
8  FROM
9      parks_and_recreation.employee_demographics
10 LIMIT 3
11 ;

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knope	44	Female	1979-09-25
3	Tom	Haverford	36	Male	1987-03-04
4	April	Ludgate	29	Female	1994-03-27
*	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

05-Limit and Aliasing in SQL

Limit to 1000 rows

```

13 # --Query 02 (sampling of limit in SQL)
14 • SELECT
15     *
16 FROM
17     parks_and_recreation.employee_demographics
18 ORDER BY
19     age DESC
20 LIMIT 3
21 ;

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
5	Jerry	Gergich	61	Male	1962-08-28
6	Donna	Meagle	46	Female	1977-07-30
1	Leslie	Knope	44	Female	1979-09-25
*	NULL	NULL	NULL	NULL	NULL

Navigator: SCHEMAS

Filter objects

- ▼ parks_and_recreation
 - ▼ Tables
 - employee_demographics
 - employee_salary
 - parks_departments
 - Views
 - Stored Procedures
 - Functions
- ▼ sakila
- ▼ sys
- ▼ world

05-Limit and Aliasing in SQL

Limit to 1000 rows

```

23 # --Query 03 (sampling of limit in SQL)
24 • SELECT
25     *
26 FROM
27     parks_and_recreation.employee_demographics
28 ORDER BY
29     age DESC
30 LIMIT 2, 1
31 ;

```

Result Grid

employee_id	first_name	last_name	age	gender	birth_date
1	Leslie	Knope	44	Female	1979-09-25
*	NULL	NULL	NULL	NULL	NULL

Navigator

05-Limit and Aliasing in SQL

Limit to 1000 rows

32

33 # --Query 04 (sampling of aliasing in SQL)

34 • SELECT

35 gender,

36 AVG(age) AS AVERAGE_AGE

37 FROM

38 parks_and_recreation.employee_demographics

39 GROUP BY

40 gender

41 HAVING AVERAGE_AGE > 40

42 ;

Result Grid

	gender	AVERAGE_AGE
▶	Male	41.2857

*****END*****