

Tables

Characters table- data, code, a query

HeidisSQL 11.1.0.616

Host: hopper.wlu.ca Database: schu5560 Table: characters

character_id	game_name	character_name	species	main_ability	weapon
1	Infamous	Cole	Conduit	electricity	amp
2	Infamous	Zeke	Human	(NULL)	gun
3	Hollow Knight	The Hollow Knight	Vessel	desolate dive	nail
4	Hollow Knight	the knight	Vessel	desolate dive	nail
5	Diablo	Diablo	Demon	fire	(NULL)
6	Undertale	Undyne	Fish warrior	undying	spear
7	Life is Strange	Max	Human	rewind	power
8	Life is Strange	Chloe	Human	rewind	gun
9	Detroit Become Human	Connor	Android	reconstruct	(NULL)
10	Detroit Become Human	Kara	Android	scan	gun
11	Detroit Become Human	Markus	Android	preconstruct	(NULL)
12	Hollow Knight	Hornet	Bug	needle	
13	Little Nightmares	Six	Human	consume	(NULL)
14	Animal Crossing	villager	Human	pocket	axe
15	Until Dawn	josh	Human	(NULL)	gun
16	Until Dawn	Han	wendigo	(NULL)	(NULL)
17	Undertale	sans	skeleton	bone	bone
18	Undertale	Annoying Dog	dog	void	bark
19	Don't Starve together	webber	Human/spider	spiderfriends	axe
20	Don't Starve together	maxwell	Human	regain sanity	axe
21	Don't Starve together	charlie	Human?	can't see	darkness
22	Don't Starve together	terrortoad	shadow monster	attacks at low sanity	itself

```

21 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
22 SHOW ENGINES;
23 SHOW COLLATION;
24 SHOW CREATE TABLE `schu5560`.`characters`;
25 SELECT * FROM `schu5560`.`characters` LIMIT 1000;

```

Connected: 00:02 h MySQL 8.0.26 Uptime: 10 days, 02:00 h Server time: 1:35 PM Idle. 27°C Sunny 1:35 PM 8/29/2021

HeidisSQL 11.1.0.616

Host: hopper.wlu.ca Database: schu5560 Table: characters

CREATE TABLE `characters` (

```

1  `character_id` INT NOT NULL AUTO_INCREMENT,
2  `game_name` VARCHAR(50) NOT NULL COLLATE 'utf8mb4_0900_ai_ci',
3  `character_name` VARCHAR(50) NOT NULL DEFAULT 'Human' COLLATE 'utf8mb4_0900_ai_ci',
4  `species` VARCHAR(50) NULL DEFAULT '' COLLATE 'utf8mb4_0900_ai_ci',
5  `main_ability` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
6  `weapon` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
7  PRIMARY KEY (`character_id`) USING BTREE,
8  INDEX `characters_games` (`game_name`) USING BTREE,
9  CONSTRAINT `fk_characters_games` FOREIGN KEY (`game_name`) REFERENCES `schu5560`.`games` (`game_name`) ON UPDATE NO ACTION ON DELETE NO ACTION
10 ) COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=23
;

```

Columns:

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation	Expression	Virtuality
1	character_id	INT		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	AUTO_INCRE...	No default	utf8mb4_0900_ai_ci		
2	game_name	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		
3	character_name	VARCHAR	50	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	'Human'		utf8mb4_0900_ai_ci		
4	species	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		
5	main_ability	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		
6	weapon	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		

Help Discard Save

```

21 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
22 SHOW ENGINES;
23 SHOW COLLATION;
24 SHOW CREATE TABLE `schu5560`.`characters`;
25 SELECT * FROM `schu5560`.`characters` LIMIT 1000;

```

Connected: 00:03 h MySQL 8.0.26 Uptime: 10 days, 02:10 h Server time: 1:36 PM Idle. 27°C Sunny 1:36 PM 8/29/2021

CREATE TABLE `characters` (

```

`character_id` INT NOT NULL AUTO_INCREMENT,
`game_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
`character_name` VARCHAR(50) NOT NULL COLLATE 'utf8mb4_0900_ai_ci',
`species` VARCHAR(50) NULL DEFAULT 'Human' COLLATE 'utf8mb4_0900_ai_ci',
`main_ability` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
`weapon` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
PRIMARY KEY (`character_id`) USING BTREE,
INDEX `FK_characters_games`(`game_name`) USING BTREE,
CONSTRAINT `FK_characters_games` FOREIGN KEY(`game_name`) REFERENCES
`schu5560`.`games`(`game_name`) ON UPDATE NO ACTION ON DELETE NO ACTION
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=23
;

```

```

1 SELECT COUNT(*) AS num_characters
2 FROM characters
3 WHERE game_name = "hollow knight"

```

characters (1r x 1c)	
	num_characters
	3

```

28 /* Affected rows: 0 Found rows: 0 Warnings: 0 Duration for 0 of 1 query: 0.000 sec. */
29 SELECT COUNT(*) FROM characters WHERE game_name = "hollow knight";
30 /* Affected rows: 0 Found rows: 1 Warnings: 0 Duration for 1 query: 0.032 sec. */
31 SELECT COUNT(*) AS num_characters FROM characters WHERE game_name = "hollow knight";
32 /* Affected rows: 0 Found rows: 1 Warnings: 0 Duration for 1 query: 0.015 sec. */

```

Connected: 00:05 h MySQL 8.0.26 Uptime: 10 days, 02:12 h Server time: 1:38 PM Idle.

Type here to search

```

SELECT COUNT(*) AS num_characters
FROM characters
WHERE game_name = "hollow knight"

```

Company table- data, code, a query

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The 'company' table is open, displaying 10 rows of data:

company_name	company_id	game_name	composer_id	number_of_employees
Nintendo	1	Animal Crossing	(NULL)	6,580
Quantic Dreams	2	Detroit Become Human	(NULL)	2,460
Blizzard	3	Diablo	(NULL)	1,480
Sucker Punch	4	Infamous	(NULL)	1,370
Team Cherry	5	Hollow Knight	2	210
Square Enix	6	Life is Strange	(NULL)	420
(indie game)	7	Undertale	1	15
Supermassive Games	8	Until Dawn	(NULL)	287
Bandai	9	Little Nightmares	3	99
klei	10	Don't Starve togeather	4	147

Below the table, the SQL code for creating the 'company' table is shown:

```

36 SHOW INDEXES FROM `company` FROM `schu5560`;
37 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='company' AND REFERENCED_TABLE_NAME IS NOT NULL;
38 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='company' AND REFERENCED_TABLE_NAME IS NOT NULL;
39 SHOW CREATE TABLE `schu5560`.`company`;
40 SELECT * FROM `schu5560`.`company` LIMIT 1000;

```

The status bar at the bottom indicates: Connected: 00:07 h MySQL 8.0.26 Uptime: 10 days, 02:14 h Server time: 1:40 PM Idle. 140 PM 27°C Sunny 8/29/2021

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The 'company' table is open, showing the 'Basic' tab with the table definition:

```

CREATE TABLE `company` (
    `company_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
    `company_id` INT NOT NULL AUTO_INCREMENT,
    `game_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
    `composer_id` INT NULL,
    `number_of_employees` INT NULL DEFAULT '1',
    PRIMARY KEY (`company_id`),
    INDEX `game_id` (`game_name`) USING BTREE,
    INDEX `composer_id` (`composer_id`) USING BTREE,
    CONSTRAINT `FK_company_games` FOREIGN KEY (`game_name`) REFERENCES `schu5560`.`games` (`game_name`) ON UPDATE NO ACTION ON DELETE NO ACTION
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=12
;

```

Below the definition, the 'Columns' tab displays the table structure:

#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation	Expression	Virtuality
1	company_name	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		
2	company_id	INT		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	AUTO_INCRE...				
3	game_name	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default		utf8mb4_0900_ai_ci		
4	composer_id	INT		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	No default				
5	number_of_e...	INT		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	'1'				

Below the columns, the SQL code for creating the 'company' table is shown:

```

36 SHOW INDEXES FROM `company` FROM `schu5560`;
37 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='company' AND REFERENCED_TABLE_NAME IS NOT NULL;
38 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='company' AND REFERENCED_TABLE_NAME IS NOT NULL;
39 SHOW CREATE TABLE `schu5560`.`company`;
40 SELECT * FROM `schu5560`.`company` LIMIT 1000;

```

The status bar at the bottom indicates: Connected: 00:07 h MySQL 8.0.26 Uptime: 10 days, 02:14 h Server time: 1:40 PM Idle. 140 PM 27°C Sunny 8/29/2021

CREATE TABLE `company` (

```

`company_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
`company_id` INT NOT NULL AUTO_INCREMENT,
`game_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
`composer_id` INT NULL,

```

```

`number_of_employees` INT NULL DEFAULT '1',
PRIMARY KEY (`company_id`) USING BTREE,
INDEX `game_id` (`game_name`) USING BTREE,
INDEX `composer_id` (`composer_id`) USING BTREE,
CONSTRAINT `FK_company_games` FOREIGN KEY (`game_name`) REFERENCES
`schu5560`.`games` (`game_name`) ON UPDATE NO ACTION ON DELETE NO ACTION
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=12
;

```

The screenshot shows the HeidiSQL interface. On the left, the database structure is displayed with the following tree:

- checking (224.0 kB)
 - schu5560 (224.0 kB)
 - characters (32.0 kB)
 - company (48.0 kB)
 - counter
 - display
 - games (32.0 kB)
 - game_charac...
 - game_compa...
 - original_soun...
 - series
 - series_compa...
 - trigBeforeGa...
 - updater

In the center, a query window shows the following SQL code and results:

```

1 SELECT company_name
2 FROM company
3 WHERE number_of_employees >2000

```

company_name
Nintendo
Quantic Dreams

At the bottom, the status bar shows:

- i3 : c32 (67 B)
- Connected: 00:09 h
- MySQL 8.0.26
- Uptime: 10 days, 02:16 h
- Server time: 1:42 PM
- Idle.
- 27°C Sunny
- 1:42 PM
- 8/29/2021

```

SELECT company_name
FROM company
WHERE number_of_employees >2000

```

Games table- data, code

The screenshot shows the HeidiSQL interface with the database set to 'schu5560' and the table set to 'games'. The left sidebar shows the schema structure. The main area displays the 'games' table data:

game_name	game_description	main_genre	release_date	rating	price	main_world_name	last_updated	console_type
Animal Crossing	simply vibe	lifestyle	2021-03-20	4	21.99	island	2021-03-20 13:38:01	s
Detroit Become Human	robots with emotions	rpg	2018-10-20	5	21.89	Detroit	2019-01-20 13:38:00	p
Disable	fight evil	adventure	2009-03-20	5	21.77	tristam	2011-01-10 13:37:57	p
Don't Starve together	eat with friends	survival	2018-03-29	4	12.99	island	2021-03-29 12:42:15	p
Hollow Knight	infected world	platformer	2018-03-20	5	19.99	Hallownest	2018-03-20 13:37:56	p
Infamous	become superhero or villan	openworld	2011-03-20	5	25.77	Empire City	2011-03-20 13:37:55	p
Life is Strange	highschooler with time powers	rpg	2015-03-20	5	18.99	Arcadia Bay	2021-03-20 13:37:59	p
Little Nightmares	spooky puzzles	puzzle	2016-03-20	4	15.99	The Maw	2017-03-20 13:40:17	p
Undertale	befriend enemies	puzzle	2016-03-20	5	12.78	underground	2017-03-20 13:37:58	c
Until Dawn	survive	horror	2017-03-20	4	50.99	cabin	2018-03-20 13:38:01	p

Below the table, the SQL code for creating the 'games' table is shown:

```

51 SHOW INDEXES FROM `games` FROM 'schu5560';
52 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='games' AND REFERENCED_TABLE_NAME IS NOT NULL;
53 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='games' AND REFERENCED_TABLE_NAME IS NOT NULL;
54 SHOW CREATE TABLE `schu5560`.`games`;
55 SELECT `game_name`, `game_description`, LEFT(`main_genre`, 256), `release_date`, `rating`, `price`, `main_world_name`, `last_updated`, `console_type` FROM `schu5560`.`games` LIMIT 1000;

```

The bottom status bar shows the connection details: Connected: 00:13 h MySQL 8.0.26 Uptime: 10 days, 02:20 h Server time: 1:46 PM Idle.

The screenshot shows the HeidiSQL interface with the database set to 'schu5560' and the table set to 'games'. The left sidebar shows the schema structure. The main area displays the 'games' table structure and its CREATE TABLE SQL code:

```

CREATE TABLE `games` (
  `game_name` VARCHAR(50) NOT NULL DEFAULT '' COLLATE 'utf8mb4_0900_ai_ci',
  `game_descrip...` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
  `main_genre` LONGTEXT NULL COLLATE 'utf8mb4_0900_ai_ci',
  `release_dat...` DATE NULL,
  `rating` DECIMAL(10,0) NULL,
  `price` DOUBLE NULL,
  `main_world_n...` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
  `last_updated` DATETIME NULL,
  `console_type` CHAR(50) NULL DEFAULT 'p' COLLATE 'utf8mb4_0900_ai_ci',
  PRIMARY KEY(`game_name`) USING BTREE,
  UNIQUE INDEX `game_description` (`game_description`) USING BTREE
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
;

```

Below the code, the table structure is shown in a grid:

#	Name	Datatype	Length/Set	Unsigne...	Allow N...	Zerofill	Default	Comment	Collation	Expression	Virtuality
1	game_name	VARCHAR	50				''		utf8mb4_0900_ai_ci		
2	game_descrip...	VARCHAR	50					No default	utf8mb4_0900_ai_ci		
3	main_genre	LONGTEXT						No default	utf8mb4_0900_ai_ci		
4	release_dat...	DATE						No default			
5	rating	DECIMAL	10,0					No default			
6	price	DOUBLE						No default			
7	main_world_n...	VARCHAR	50					No default	utf8mb4_0900_ai_ci		
8	last_updated	DATETIME						No default			
9	console_type	CHAR	50				'p'		utf8mb4_0900_ai_ci		

Below the structure, the SQL code for creating the 'games' table is shown:

```

51 SHOW INDEXES FROM `games` FROM 'schu5560';
52 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='games' AND REFERENCED_TABLE_NAME IS NOT NULL;
53 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='games' AND REFERENCED_TABLE_NAME IS NOT NULL;
54 SHOW CREATE TABLE `schu5560`.`games`;
55 SELECT `game_name`, `game_description`, LEFT(`main_genre`, 256), `release_date`, `rating`, `price`, `main_world_name`, `last_updated`, `console_type` FROM `schu5560`.`games` LIMIT 1000;

```

The bottom status bar shows the connection details: Connected: 00:14 h MySQL 8.0.26 Uptime: 10 days, 02:21 h Server time: 1:47 PM Idle.

CREATE TABLE `games` (

```

`game_name` VARCHAR(50) NOT NULL DEFAULT "" COLLATE 'utf8mb4_0900_ai_ci',
`game_descrip...` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',

```

```
`main_genre` LONGTEXT NULL COLLATE 'utf8mb4_0900_ai_ci',
`release_date` DATE NULL,
`rating` DECIMAL(10,0) NULL,
`price` DOUBLE NULL,
`main_world_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
`last_updated` DATETIME NULL,
`console_type` CHAR(50) NULL DEFAULT 'p' COLLATE 'utf8mb4_0900_ai_ci',
PRIMARY KEY (`game_name`) USING BTREE,
UNIQUE INDEX `game_description` (`game_description`) USING BTREE
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
;
```

Game_characters view - data, code

The screenshot shows the HeidiSQL interface with the database set to 'schu5560' and the table 'game_characters'. The table structure is as follows:

game_name	character_name	species	main_ability	weapon	main_world_name	price
Animal Crossing	villager	Human	pocket	axe	island	21.99
Detroit Become Human	Connor	Android	reconstruct	(NULL)	Detroit	21.89
Detroit Become Human	Kara	Android	scan	gun	Detroit	21.89
Detroit Become Human	Markus	Android	preconstruct	(NULL)	Detroit	21.89
Diablo	Diablo	Demon	fire	(NULL)	tristam	21.77
Don't Starve together	charlie	Human?	can't see	darkness	island	12.99
Don't Starve together	maxwell	Human	regain sanity	axe	island	12.99
Don't Starve together	terriblebeak	shadow monster	attacks at low sanity	itself	island	12.99
Don't Starve together	webber	Human/spider	spiderfriends	axe	island	12.99
Hollow Knight	Hornet	Bug		needle	Hallownest	19.99
Hollow Knight	The Hollow Knight	Vessel		nail	Hallownest	19.99
Hollow Knight	the knight	Vessel	desolate dive	nail	Hallownest	19.99
Infamous	Cole	Conduit	electricity	amp	Empire City	25.77
Infamous	Zeke	Human	(NULL)	gun	Empire City	25.77
Life is Strange	Chloe	Human		gun	Arcadia Bay	18.99
Life is Strange	Max	Human	rewind	power	Arcadia Bay	18.99
Little Nightmares	Six	Human	consume	(NULL)	The Maw	15.99
Undertale	Annoying Dog	dog	void	bark	underground	12.78
Undertale	sans	skeleton	bone	bone	underground	12.78
Undertale	Undyne	Fish warrior	undying	spear	underground	12.78
Until Dawn	Han	wendigo	(NULL)	(NULL)	cabin	50.99
Until Dawn	josh	Human	(NULL)	gun	cabin	50.99

The bottom pane shows the SQL code for creating the view:

```

58 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='game_characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
59 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='game_characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
60 SHOW CREATE VIEW `schu5560`.`game_characters`;
61 SELECT CAST(LOAD_FILE(CONCAT(IFNULL(@GLOBAL.datadir, CONCAT(@GLOBAL.basedir, 'data/')), 'schu5560/game_characters.frm')) AS CHAR CHARACTER SET utf8);
62 SELECT * FROM `schu5560`.`game_characters` LIMIT 1000;

```

The screenshot shows the HeidiSQL interface with the database set to 'schu5560' and the table 'game_characters' selected as a view. The view definition is:

```

Name: game_characters
Definer: schu5560@%
SQL security: Definer
Algorithm: UNDEFINED
Check option for updates: None
Select statement:
1 select `G`.`game_name` AS `game_name`,`C`.`character_name` AS
`character_name`,`C`.`species` AS `species`,`C`.`main_ability` AS `main_ability`,`C`.`weapon` AS
`weapon`,`G`.`main_world_name` AS `main_world_name`,`G`.`price` AS `price` from
(`games` `G` join `characters` `C`) where (`G`.`game_name` = `C`.`game_name`) order by
`G`.`game_name`,`C`.`character_name`

```

The bottom pane shows the SQL code for creating the view:

```

58 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='game_characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
59 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='game_characters' AND REFERENCED_TABLE_NAME IS NOT NULL;
60 SHOW CREATE VIEW `schu5560`.`game_characters`;
61 SELECT CAST(LOAD_FILE(CONCAT(IFNULL(@GLOBAL.datadir, CONCAT(@GLOBAL.basedir, 'data/')), 'schu5560/game_characters.frm')) AS CHAR CHARACTER SET utf8);
62 SELECT * FROM `schu5560`.`game_characters` LIMIT 1000;

```

```

select `G`.`game_name` AS `game_name`,`C`.`character_name` AS
`character_name`,`C`.`species` AS `species`,`C`.`main_ability` AS `main_ability`,`C`.`weapon` AS
`weapon`,`G`.`main_world_name` AS `main_world_name`,`G`.`price` AS `price` from
(`games` `G` join `characters` `C`) where (`G`.`game_name` = `C`.`game_name`) order by
`G`.`game_name`,`C`.`character_name`

```

Game_company_ost view - data, code

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The 'game_company_ost' view is displayed in the main pane, showing a list of songs with their details. Below the table, the SQL code for creating the view is visible.

game_name	OST_name	company_name	composer_name	game_description	price
Don't Starve together	title theme	klei	slye	eat with friends	12.99
Hollow Knight	Haunted Foes	Team Cherry	Chris	infected world	19.99
Hollow Knight	Nightmare King	Team Cherry	Chris	infected world	19.99
Hollow Knight	Nosk	Team Cherry	Chris	infected world	19.99
Hollow Knight	Sealed Vessel	Team Cherry	Chris	infected world	19.99
Little Nightmares	Six's song	Bandai	Tobias	spooky puzzles	15.99
Undertale	Amalgam	(indie game)	Toby	befriend enemies	12.78
Undertale	battle against a true hero	(indie game)	Toby	befriend enemies	12.78
Undertale	Bird That Carries You Over A Disproportionately Small Gap	(indie game)	Toby	befriend enemies	12.78
Undertale	Spear of Justice	(indie game)	Toby	befriend enemies	12.78
Undertale	Your Best Nightmare	(indie game)	Toby	befriend enemies	12.78

```

65 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='game_company_ost' AND REFERENCED_TABLE_NAME IS NOT NULL;
66 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='game_company_ost' AND REFERENCED_TABLE_NAME IS NOT NULL;
67 SHOW CREATE VIEW `schu5560`.`game_company_ost`;
68 SELECT CAST(LOAD_FILE(CONCAT(IFNULL(@GLOBAL.datadir, CONCAT(@GLOBAL_basedir, 'data/')), 'schu5560/game_company_ost.frm')) AS CHAR CHARACTER SET utf8);
69 SELECT * FROM `schu5560`.`game_company_ost` LIMIT 1000;

```

The status bar at the bottom shows the connection details: r1:c1, Connected: 00:16 h, MySQL 8.0.26, Uptime: 10 days, 02:23 h, Server time: 1:49 PM, Idle. It also shows the system status: 27°C Sunny, 1:49 PM, 8/29/2021.

```

select `G`.`game_name` AS `game_name`, `O`.`OST_name` AS
`OST_name`, `C`.`company_name` AS `company_name`, `O`.`composer_name` AS
`composer_name`, `G`.`game_description` AS `game_description`, `G`.`price` AS `price` from
(`games` `G` join `company` `C`) join `original_sound_track` `O` where ((`G`.`game_name` =
`C`.`game_name`) and (`C`.`composer_id` = `O`.`composer_id`)) order by
`G`.`game_name`, `O`.`OST_name`

```

Other tables - data, code

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The 'original_sound_track' table is open, displaying 11 rows of data:

OST_name	OST_id	composer_name	composer_id
Your Best Nightmare	1	Toby	1
Spear of Justice	2	Toby	1
Sealed Vessel	3	Chris	2
Haunted Foes	4	Chris	2
Nosk	5	Chris	2
Six's song	6	Tobias	3
Nightmare King	7	Chris	2
battle against a true hero	8	Toby	1
Amalgam	9	Toby	1
title theme	10	skyte	4
BirdThatCarriesYouOverADisproportionatelySmallGap	11	Toby	1

Below the table, the schema creation code is shown:

```

71 SHOW INDEXES FROM `original_sound_track` FROM `schu5560`;
72 SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='original_sound_track' AND REFERENCED_TABLE_NAME IS NOT NULL;
73 SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='original_sound_track' AND REFERENCED_TABLE_NAME IS NOT NULL;
74 SHOW CREATE TABLE `schu5560`.`original_sound_track`;
75 SELECT * FROM `schu5560`.`original_sound_track` LIMIT 1000;
    
```

The bottom status bar shows the connection details: Connected: 00:17 h, MySQL 8.0.26, Uptime: 10 days, 02:24 h, Server time: 1:50 PM, Idle.

```

CREATE TABLE `original_sound_track` (
  `OST_name` VARCHAR(50) NOT NULL DEFAULT 'song' COLLATE
  'utf8mb4_0900_ai_ci',
  `OST_id` INT NOT NULL AUTO_INCREMENT,
  `composer_name` VARCHAR(50) NOT NULL DEFAULT 'composer' COLLATE
  'utf8mb4_0900_ai_ci',
  `composer_id` INT NOT NULL,
  PRIMARY KEY (`OST_id`) USING BTREE,
  UNIQUE INDEX `OST_name` (`OST_name`) USING BTREE,
  INDEX `composer_id` (`composer_id`) USING BTREE,
  CONSTRAINT `FK_original_sound_track_company` FOREIGN KEY (`composer_id`)
  REFERENCES `schu5560`.`company` (`composer_id`) ON UPDATE NO ACTION ON DELETE
  NO ACTION
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=12
; 
```

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The left sidebar shows the database structure with the 'series' table highlighted. The main pane displays the 'series' table data, which includes columns: series_id, game_name, series_game_name, release_date, same_main_character, and price. The data consists of 9 rows. Below the table is a SQL query window showing the schema creation code for the 'series' table. The system tray at the bottom right indicates the computer is connected to the internet, the server has been up for 10 days, and the current time is 1:51 PM on August 29, 2021.

series_id	game_name	series_game_name	release_date	same_main_character	price
1	infamous	infamous 2	2018-03-28	yes	11.99
2	Infamous	infamous second son	2019-03-28	no	12.99
3	Animal Crossing	animal Crossing city folk	2021-03-28	yes	34.09
4	Hollow Knight	silk song	2021-03-28	no	55.99
5	Life is Strange	life is strange 2	2018-03-28	no	55.99
6	Life is Strange	life is strange before the storm	2019-03-28	no	50.99
7	Little Nightmares	Little nightmares 2	2021-03-28	no	13.99
8	Undertale	Delta runs	2019-03-28	no	21.99
9	Don't Starve together	Dont starve	2020-03-29	yes	25.99

```

77 SHOW INDEXES FROM `series` FROM `schu5560`;
78 SELECT * FROM INFORMATION_SCHEMA.REFERENTIAL_CONSTRAINTS WHERE CONSTRAINT_SCHEMA='schu5560' AND TABLE_NAME='series' AND REFERENCED_TABLE_NAME IS NOT NULL;
79 SELECT * FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='schu5560' AND TABLE_NAME='series' AND REFERENCED_TABLE_NAME IS NOT NULL;
80 SHOW CREATE TABLE `schu5560`.`series`;
81 SELECT * FROM `schu5560`.`series` LIMIT 1000;

```

```

CREATE TABLE `series` (
    `series_id` INT NOT NULL AUTO_INCREMENT,
    `game_name` VARCHAR(50) NULL COLLATE 'utf8mb4_0900_ai_ci',
    `series_game_name` VARCHAR(50) NOT NULL DEFAULT '0' COLLATE
'utf8mb4_0900_ai_ci',
    `release_date` DATE NULL,
    `same_main_character` VARCHAR(50) NULL DEFAULT 'yes' COLLATE
'utf8mb4_0900_ai_ci',
    `price` DOUBLE NULL,
    PRIMARY KEY (`series_id`) USING BTREE,
    INDEX `FK_series_games`(`game_name`) USING BTREE,
    CONSTRAINT `FK_series_games` FOREIGN KEY(`game_name`) REFERENCES
`schu5560`.`games`(`game_name`) ON UPDATE RESTRICT ON DELETE RESTRICT
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
AUTO_INCREMENT=11
;

```

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. The 'series_company' table is open, displaying 9 rows of data:

series_id	company_id
3	1
1	4
2	4
4	5
5	6
6	6
8	7
7	9
9	10

The left sidebar shows the schema structure of the database, including tables like 'characters', 'company', 'counter', 'display', 'games', 'game_charac...', 'game_compa...', 'original_soun...', 'series', and 'series_comp...'. The 'series_company' table is highlighted.

The bottom status bar shows the connection details: r1 : c1, Connected: 00:19 h, MySQL 8.0.26, Uptime: 10 days, 02:25 h, Server time: 1:52 PM, 27°C Sunny, 1:52 PM, 8/29/2021.

```

CREATE TABLE `series_company` (
    `series_id` INT NOT NULL,
    `company_id` INT NOT NULL,
    PRIMARY KEY (`series_id`, `company_id`) USING BTREE,
    INDEX `FK_series_company_company` (`company_id`) USING BTREE,
    CONSTRAINT `FK_series_company_company` FOREIGN KEY (`company_id`)
    REFERENCES `schu5560`.`company` (`company_id`) ON UPDATE CASCADE ON DELETE
    CASCADE,
    CONSTRAINT `FK_series_company_series` FOREIGN KEY (`series_id`)
    REFERENCES `schu5560`.`series` (`series_id`) ON UPDATE CASCADE ON DELETE
    CASCADE
)
COLLATE='utf8mb4_0900_ai_ci'
ENGINE=InnoDB
;

```

Function, procedure

Function

The screenshot shows the HeidiSQL interface for MySQL 8.0.26. A new function named 'counter' is being created under the database 'schu5560'. The routine body contains SQL code to select the count of characters from the 'characters' table where the game name is 'hollow knight'. The 'Run routine(s)' button at the bottom right is highlighted.

```
CREATE DEFINER='schu5560'@'%' FUNCTION `counter`()
RETURNS int
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
    RETURN (SELECT COUNT(*)
    FROM characters
    WHERE game_name = "hollow knight");
END
```

```
/* Affected rows: 0 Found rows: 2 Warnings: 0 Duration for 1 query: 0.016 sec. */
44 SELECT CURRENT_USER();
45 SHOW CREATE FUNCTION `schu5560`.`counter`;
46 SHOW CREATE PROCEDURE `schu5560`.`display`;
47 SHOW CREATE FUNCTION `schu5560`.`counter`;
```

The screenshot shows the configuration of the 'counter' function in HeidiSQL. The 'Name' field is set to 'counter' and the 'Definer' field is set to 'schu5560@%'. The 'Type' is set to 'Function (returns a result)', 'Returns' is set to 'int', and 'Data access' is 'Contains SQL'. The 'Routine body' section is identical to the one in the previous screenshot. The 'Run routine(s)' button is again highlighted.

```
Name: counter Definer: schu5560@%
Comment:
Type: Function (returns a result) Data access: Contains SQL
Returns: int SQL Security: Definer
Deterministic
```

```
/* Affected rows: 0 Found rows: 1 Warnings: 0 Duration for 1 query: 0.015 sec. */
43 SELECT company_name FROM company WHERE number_of_employees > 2000;
44 /* Affected rows: 0 Found rows: 2 Warnings: 0 Duration for 1 query: 0.016 sec. */
45 SELECT CURRENT_USER();
46 SHOW CREATE FUNCTION `schu5560`.`counter`;
```

BEGIN

RETURN (SELECT COUNT(*)
FROM characters

```
WHERE game_name = "hollow knight");
```

END

Procedure

The screenshot shows two instances of the HeidiSQL MySQL client interface. Both instances are connected to the database 'schu5560' on the host 'hopper.wlu.ca'.

Top Window (Procedure Definition):

- Database Filter:** schu5560
- Table Filter:** display
- Query:** Procedure display
- Routine body:**

```
1 CREATE DEFINER='schu5560@%' PROCEDURE `display`()
2 LANGUAGE SQL
3 NOT DETERMINISTIC
4 CONTAINS SQL
5 SQL SECURITY DEFINER
6 COMMENT ''
7 BEGIN
8
9     SELECT COUNT(*)
10    FROM original_sound_track
11   WHERE composer_name = "toby";
12
13
14 END
```
- Buttons:** Help, Discard, Save, Run routine(s) ...

Bottom Window (Procedure Creation Log):

- Query:** Procedure display
- Log:**

```
45 SELECT CURRENT_USER();
46 SHOW CREATE FUNCTION `schu5560`.`counter`;
47 SHOW CREATE PROCEDURE `schu5560`.`display`;
48 SHOW CREATE FUNCTION `schu5560`.`counter`;
49 SHOW CREATE PROCEDURE `schu5560`.`display`;
```
- Information:** Connected: 00:12 h MySQL 8.0.26 Uptime: 10 days, 02:19 h Server time: 1:45 PM Idle. 27°C Sunny 1:45 PM 8/29/2021

Second Window (Procedure Definition):

- Database Filter:** schu5560
- Table Filter:** display
- Query:** Procedure display
- Form Fields:**
 - Name: display
 - Definer: schu5560@%
 - Comment:
 - Type: Procedure (doesn't return a result)
 - Data access: Contains SQL
 - Returns:
 - Deterministic checkbox (unchecked)
- Routine body:**

```
1 BEGIN
2
3     SELECT COUNT(*)
4    FROM original_sound_track
5   WHERE composer_name = "toby";
6
7
8 END
```
- Buttons:** Help, Discard, Save, Run routine(s) ...

Bottom Window (Procedure Creation Log):

- Query:** Procedure display
- Log:**

```
43 SELECT company_name FROM company WHERE number_of_employees >2000;
44 /* Affected rows: 0  Found rows: 2  Warnings: 0  Duration for 1 query: 0.016 sec. */
45 SELECT CURRENT_USER();
46 SHOW CREATE FUNCTION `schu5560`.`counter`;
47 SHOW CREATE PROCEDURE `schu5560`.`display`;
```
- Information:** Connected: 00:11 h MySQL 8.0.26 Uptime: 10 days, 02:18 h Server time: 1:44 PM Idle. 27°C Sunny 1:44 PM 8/29/2021

BEGIN

```
SELECT COUNT(*)
FROM original_sound_track
WHERE composer_name = "toby";
```

END

Trigger, event

Trigger

The screenshot shows the HeidiSQL interface for MySQL 8.0.26. The database is schu5560. A trigger named 'trigBeforeGames' is being created on the 'games' table with a 'BEFORE INSERT' event. The trigger body contains logic to check if the 'rating' column value is greater than 5, set an error message, and signal it. The trigger definition is:

```
1 CREATE TRIGGER `trigBeforeGames` BEFORE INSERT ON `games` FOR EACH ROW BEGIN
2
3     IF NEW.rating > 5 THEN
4         SIGNAL SQLSTATE '45000'
5         SET MESSAGE_TEXT = 'rating cant be over 5';
6         table_name = 'games', column_name = 'rating';
7     END IF;
8
9 END
```

The status bar at the bottom indicates the connection is 00:22 h, MySQL 8.0.26, Uptime: 10 days, 02:29 h, Server time: 1:55 PM, Idle, 27°C Sunny, 1:55 PM, 8/29/2021.

The screenshot shows the HeidiSQL interface for MySQL 8.0.26. The database is schu5560. A trigger named 'trigBeforeGames' is being created on the 'games' table with a 'BEFORE INSERT' event. The trigger body contains logic to check if the 'rating' column value is greater than 5, set an error message, and signal it. The trigger definition is:

```
1 CREATE DEFINER='schu5560'@'%' TRIGGER `trigBeforeGames` BEFORE INSERT ON `games` FOR EACH ROW BEGIN
2
3     IF NEW.rating > 5 THEN
4         SIGNAL SQLSTATE '45000'
5         SET MESSAGE_TEXT = 'rating cant be over 5';
6         table_name = 'games', column_name = 'rating';
7     END IF;
8
9 END
```

The status bar at the bottom indicates the connection is 00:23 h, MySQL 8.0.26, Uptime: 10 days, 02:29 h, Server time: 1:55 PM, Idle, 27°C Sunny, 1:55 PM, 8/29/2021.

`CREATE DEFINER='schu5560'@'%' TRIGGER `trigBeforeGames` BEFORE INSERT ON `games` FOR EACH ROW BEGIN`

```

if NEW.rating>5 then
  SIGNAL SQLSTATE '45000'
  SET MESSAGE_TEXT = 'rating cant be over 5',
  table_name = 'games', column_name = 'rating';
END if;

```

END

Event

The screenshot shows the HeidiSQL interface with the database 'schu5560' selected. In the left sidebar, a tree view shows various tables and triggers under the schema 'schu5560'. The 'updater' trigger is highlighted.

Event Settings:

- Name: updater
- Definer: schu5560@%
- Comment:
- Drop event after expiration:
- State: Enable Disable Disable on slave

Execution body:

```

1 BEGIN
2
3 SELECT games.price ,games.price -1
4 FROM games;
5
6 END

```

Log:

```

89 SHOW CREATE TRIGGER `schu5560`.`trigBeforeGames`;
90 SHOW CREATE EVENT `schu5560`.`updater`;
91 SHOW TRIGGERS FROM `schu5560`;
92 SHOW CREATE TRIGGER `schu5560`.`trigBeforeGames`;
93 SHOW CREATE EVENT `schu5560`.`updater`;

```

At the bottom, the status bar shows: Connected: 00:23 h MySQL 8.0.26 Uptime: 10 days, 02:30 h Server time: 1:56 PM Idle. 27°C Sunny 1:56 PM 8/29/2021

The screenshot shows two instances of the HeidiSQL application interface. Both instances are connected to the same MySQL database, schu5560, on the host hopper.wlu.ca.

Top Window (Event Creation):

- Database Filter:** Database filter, Table filter
- Host:** hopper.wlu.ca
- Database:** schu5560
- Event:** updater
- Timing:** CREATE code
- Execution body:**

```

1 BEGIN
2
3 SELECT games.price ,games.price -1
4 FROM games;
5
6 END

```
- Buttons:** Help, Discard, Save
- Status Bar:** Connected: 00:24 h, MySQL 8.0.26, Uptime: 10 days, 02:31 h, Server time: 1:57 PM, Idle, 27°C Sunny, 1:57 PM, 8/29/2021

Bottom Window (Event Definition):

- Database Filter:** Database filter, Table filter
- Host:** hopper.wlu.ca
- Database:** schu5560
- Event:** updater
- Timing:** CREATE code
- Execution body:**

```

1 CREATE DEFINER='schu5560'@'%' EVENT `updater`
2 ON SCHEDULE
3 | EVERY 1 MONTH STARTS '2021-03-31 23:48:59'
4 | ON COMPLETION PRESERVE
5 | ENABLE
6 | COMMENT ''
7 DO BEGIN
8 | | SELECT games.price ,games.price -1
9 | | FROM games;
10 | |
11 END

```
- Buttons:** Help, Discard, Save
- Status Bar:** Connected: 00:24 h, MySQL 8.0.26, Uptime: 10 days, 02:30 h, Server time: 1:57 PM, Idle, 27°C Sunny, 1:57 PM, 8/29/2021

```

CREATE DEFINER='schu5560'@'%' EVENT `updater`
ON SCHEDULE
EVERY 1 MONTH STARTS '2021-03-31 23:48:59'
ON COMPLETION PRESERVE
ENABLE
COMMENT "

```

DO BEGIN

```
SELECT games.price ,games.price -1  
FROM games;
```

END