|  |  |  |
| --- | --- | --- |
|  | manually | automatic |
| create db connection |  | Machine generated alternative text:    Machine generated alternative text: MySQL Connections ® (S) Local instance MySQL8O I root Iocalhost:3306    # These are the mandatory fields for a database connection:    Machine generated alternative text: Connection Name: Local instance MySQL8O Connection Remote Management System Profile I Standard (TCP/IP) Connection Method: I_________________________ Parameters SSL Advanced Hosthame: localhost , Username: root Password: ‘ ‘ ‘ Store in Vault ... Clear ________ Method to use to conned to the ROOMS Port: jjjjj Name or IP address of the server host - and _________ TCP/IP port, Name of the user to conned with. The user’s password. Will be requested later if Ws not set.    # Test connection    Machine generated alternative text: MySQL Workbench requested Iat O Successfully made the MySOL connection schema. Lea Information related to this connection: Host: locaihost Port: 3306 Usen root SSL enabled with DHE-RSA-AEs12S-GcM-sHA256 A successful MySQL connection was made with the parameters defined for this connection. OK we Down Test connecton |
| create a model |  | Machine generated alternative text:    Machine generated alternative text: Models ®  # (optional) if no schema present, create new schema like this (press "+" button):    Machine generated alternative text: Model Overview Add Diagram Name: __________ Specify the name of the schema here. You can Rename References R.efactc model, changing al refwences found n view, triggies, stŒed procedies and functions from Charset/Collabon: Default Charsel E Default Collaboi , The character set and *s colation selected here Comments: r |
| add diagram |  | Machine generated alternative text: Model Overview Md Diagram |
| create table in diagram |  | The enhanced entity–relationship (**EER**) model (or extended entity–relationship model) in computer science is a high-level or conceptual data model used in the design of databases.    Machine generated alternative text: 0 Birds Eye Diagram Zoom: lOOZ 0 0% Catalog Tree V  mydb 3 Tables Views Routine Groups      # define columns    Machine generated alternative text: Q D C oltablel Vj O O D tablel - Table X V > Ta bic Name: Sch ema: ] mydb Column Name Datatype PK D NN D UQ D B D UN D ZF D AI D G D Defäi.d ( > |
| save model |  | # save model as .mwb file (to **src/main/resources**)    Machine generated alternative text: File Edit View Arrange Model Database |
| **forward engineering** |  | # generate tables based on the EER diagram    Machine generated alternative text: Database Tools Scripting Help Connect to Database... I Manage Connections... Reverse Engineer.. L FOEward Engineer... Ctrl+1J Ctrl+R Cti-l+G    # next > next > … next |
| create schema | CREATE SCHEMA `new\_shcema` ; | Machine generated alternative text: Navigator :::. new_schema- Schema x [ MANAGEMENT Name: Inewche I Q Server Status Rename References . Client Connections I Users and Privileges Charsetjcollation: Default Charset ‘ Default Collaoi E Status and System Variables |
| create table | CREATE TABLE `sql\_tutorial`.`departments` (  `id` INT NOT NULL,  `name` VARCHAR(45) NULL,  PRIMARY KEY (`id`)); | Machine generated alternative text: Tab’e Name: emp4oyees Colimin Name Datatype PIC  UQ B UN ZF AI G INT EEEEEE <)first_name VARCHAR(45) El El E E] E E E E] Iast_name VARCHAR(45) E E E E E E E E EEEEEEEE |
| create foreign key |  | Machine generated alternative text: employees - Table x Lì Table Name: Charset/Collabon: Comments: employees Schema: sql_tul ImoÐß utf8mb4  utfn4_O9OO_ai_ci . Rerenced Column Rerenced Table Column sql_tutorialdepartmer E id E first_name E last_name id_departments Foreign Key Name id_departments id    Machine generated alternative text: Demiiovees ? id INT(11) ‘fret_nan e VMCHAR(45) / læt_name VPRCt-IAR(45) ‘) id_dq,aiments INT(11) D departments’ ¡d INT(11) ‘name VAlHAR(45) I. |
| **reverse engineering** |  | # generate EER diagram based on tables    Machine generated alternative text: Database Server Tools Scripting Help Connect to Database... Ctrl+U Manage Connections... Reverse Engineer... Ctrl÷R    Machine generated alternative text: Select the schernas you want to include: db_name hibernate sqI_tutDrial sys    next > next > … > **execute** > next > **finish** |
| create views | # views  CREATE VIEW employees\_with\_dept\_name AS  SELECT  employees.\*, departments.name AS department\_name  FROM  employees  LEFT JOIN  departments USING (id\_departments);    # select view  SELECT  \* FROM employees\_with\_dept\_name; | Machine generated alternative text: new_view - Væw 1 The naine cf the view is parsed autcinaticaly frcrn the DDI. I statement. The DDI. s parsed autcmaticaly wlide ai type. Name: mew_view DDL:   11 u 1 • CREATE VIEW employees_with_dept_name AS 2 SELECT 3 employees.t, departments.name AS department_name 4 FROM 5 employees 6 LEFT JOIN 7 departments USING (id_departments); 8 |
| create function | DELIMITER $$  CREATE FUNCTION add\_two\_ints(  a INT , -- parametr a of type int  b INT -- parametr b of type int  ) RETURNS INT -- function returns an int  DETERMINISTIC -- <https://dev.mysql.com/doc/refman/8.0/en/stored-programs-logging.html>  READS SQL DATA  BEGIN -- body of the function  RETURN a + b;  END$$    DETERMINISTIC READS SQL DATA trebuie adaugat pentru ca functia ar putea modifica starea bazei de date si este o masura de protectie pentru MySql 8. Se poate seta o proprietate globala ca sa nu mai fie nevoie sa aduagati comanda la fiecare functie. Momentan o folosim asa.    # find the created function here    Machine generated alternative text: ‘ Functions fo add_three_ints | Machine generated alternative text: ylJ Functi” ac  Create Function.. fo ac Refresh JI fOac.    Machine generated alternative text: new function - Routine x9 Name: new_functon DDL: 1 • ÿCREATE FUNCTION ‘add_numbers’ d 2 HINT, HINT 4 L.) S RETURNS INTEGER 6 DETERMINISTIC READS SQL DATA 7  BEGIN L RETURNa+b; 9 END 10    **apply** |
| create procedure | # procedure  DELIMITER $$  CREATE PROCEDURE insert\_employee(  p\_first\_name VARCHAR(45),  p\_last\_name VARCHAR(45),  p\_department\_id INT)  BEGIN  INSERT INTO employees  (first\_name , last\_name , id\_departments)  VALUES  (p\_first\_name , p\_last\_name , p\_department\_id);  END$$    # call procedure  CALL insert\_employee('Kate', 'Joseph', 3);    # find the created procedure here    Machine generated alternative text: V  Stored Procedures insertemployee | A group of one or more database statements stored in the database’s data dictionary.  SQL stored procedures can be called from a program outside the database server such as a web-server or a client program.    Machine generated alternative text: Stored Procedures Functi Create Stored Procedtxe... sys Refresh tJI    Machine generated alternative text: • insert_employee - Routine x9 sanie Of the rosie B parsed automatcally from the Name: insert employee I statement. The DDI. is parsed automatically mMe you t DDL: 1 • SICREATE DEFINER=root’localhost’ PROCEDURE insert_employe&( 2 I p_first_name VARCNAR( ), 3 I p_last_name VARCI-tAR(- ), 4 Lp_department_id INT) s S BEGIN 6 I INSERT INTO employees 7 I (first_name , last_name , id_departments) 8 I VALUES g (p_first_name , p_last_name , p_department_id); 10 END    **apply**    # you can save the procedure in an **sql** file  Machine generated alternative text: insert_employee LJ •QbjLiLj |
| create triggers | CREATE TABLE `sql\_tutorial`.`employees\_history` (  `id` INT NOT NULL,  `old\_first\_name` VARCHAR(45) NULL,  `old\_last\_name` VARCHAR(45) NULL,  `modification\_date` VARCHAR(45) NULL,  `new\_first\_name` VARCHAR(45) NULL,  `new\_last\_name` VARCHAR(45) NULL,  PRIMARY KEY (`id`));    # trigger  DELIMITER $$  CREATE TRIGGER store\_history BEFORE UPDATE ON employees  FOR EACH ROW  BEGIN  INSERT INTO employees\_history  VALUES (new\_id\_employees , NOW(),  old\_first\_name , old\_last\_name,  new\_first\_name , new\_last\_name);  END$$    # fire trigger  UPDATE employees SET first\_name='Lucy' WHERE first\_name='Kate';    # find the created trigger here    Machine generated alternative text: @ sql tutorial V Tables E customers E departments y E employe Columns I.  Indexes Foreign Keys V31 Triggers El employees_AF store_histoy | Machine generated alternative text: Table Name: (Pbovees Schema: sqLtutoñal BEFORE INSERT 4 Q, [[J  V AFTER INSERT 1 • CREATE DEFINER — CURRENT_USER TRIGGER sql_tutorial .employees_AFTER_INSERT employees_AFTER_INSERT 2 AFTER INSERT ON employees BEFORE UPDATE 3 FOR EACH ROW AFTER UPDATE 4 BEGIN BEFORE DELETE 5 t :: logic AFTER DELETE 6 END 7 ( Columns Indexes ForeignKeys Tí,is Partitioning OptiorG |