

Actividad / Práctica: Práctica 4 Gestión Académica

Alumno/a: Guillermo Boquizo Sánchez

Objetivo: Diseñar una base de datos para la gestión académica de los alumnos del departamento de informática.

Herramientas: Mysqlworkbench.

Fichero de alumnos.

1. Análisis de requisitos.
 2. Diseño conceptual. Mysqlworkbench
 3. Diseño lógico. Mysqlworkbench
 4. Diseño físico. Mysqlworkbench
 5. Introducir los datos actuales en el sistema de bases de datos.
-
1. En esta práctica realizaremos todo el proceso del desarrollo de una base de datos, desde su fase preliminar de análisis de requisitos hasta acabar introduciendo los datos en el sistema de bases de datos, orientada en torno a la gestión académica del centro.

Los diseños conceptuales, lógicos y físicos se llevan a cabo directamente con Mysqlworkbench, por la gran versatilidad que tiene y por la capacidad de generar sentencia sql a partir del modelado.

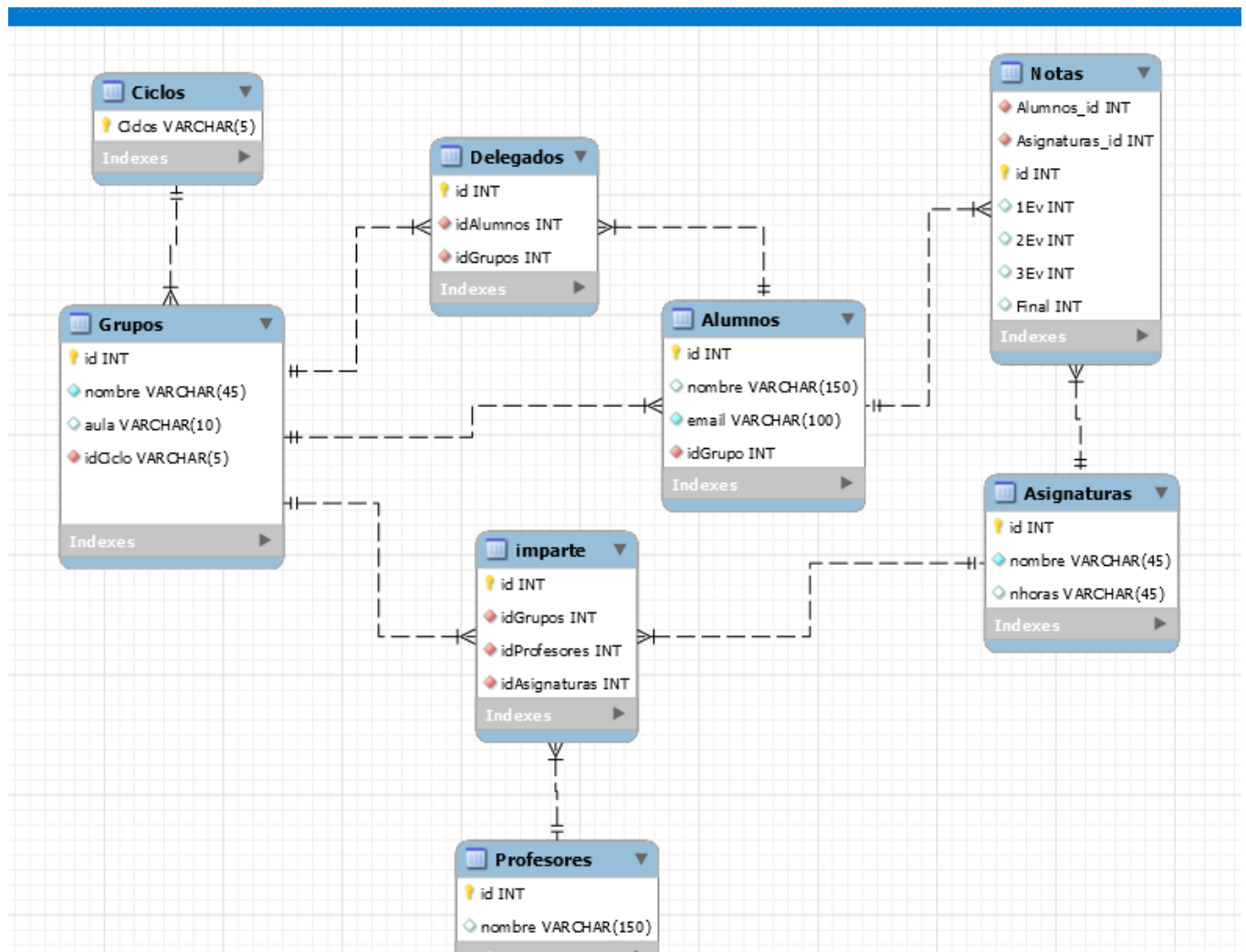
Tenemos que tener previsto que nuestra base de datos debe al menos ser capaz de:

- Almacenar información de los alumnos de los ciclos, debidamente identificados
- Éstos deben pertenecer a un grupo
- Los grupos lo son de determinados ciclos, los constituyen alumnos y a su vez a ellos los profesores les imparten las asignaturas.
- Cada grupo tiene un delegado/a
- Las asignaturas se evalúan mediante notas, que lo son de un alumno en relación a una asignatura.

Dejaremos sin recoger la semántica que permita a algún alumno pertenecer al mismo tiempo a varios ciclos, dado que mysql no nos va a permitir solventarlo.

2-3: -

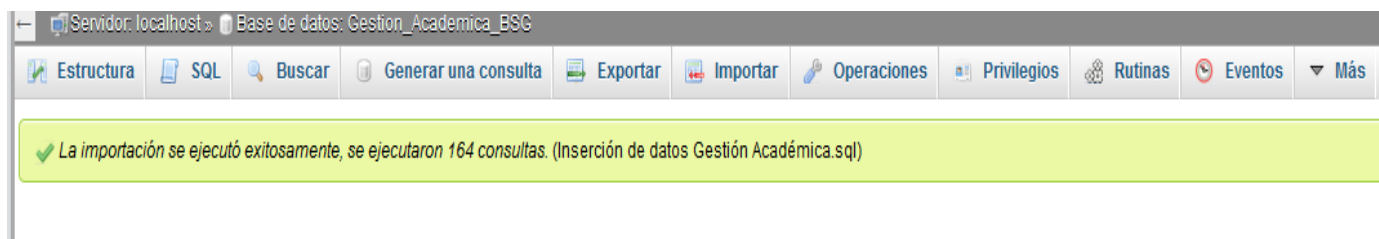
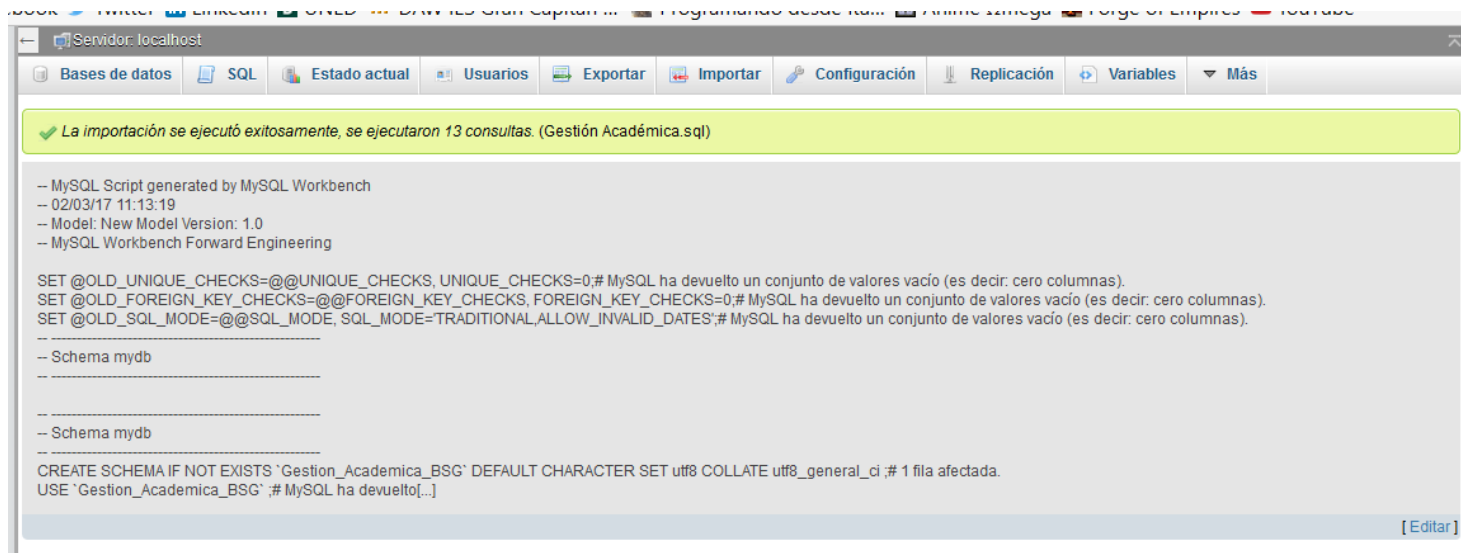
Con estas características, y en función de todo lo trabajado previamente, podemos emplear el siguiente modelo creado en Workbench:



4-5

Una vez generado el modelo, podemos a su vez generar de manera automática un script sql, útil para la creación de nuestra base de datos.

Nuevamente emplearemos phpMyAdmin para insertar tanto el script autogenerado, como el script de inserción de datos.



El script autogenerated es el que sigue, se le ha editado el nombre de la base de datos a la que hace referencia la sentencia CREATE...IF EXIST, que genera una base de datos si no estuviese creada.

```
D:/Guillermo/Escritorio/Sin título-1.sql (Primeros Pasos) - Brackets
Navegación Desarrollo Ayuda Emmet

Sin título-1.sql

1  -- MySQL Script generated by MySQL Workbench
2  -- 02/03/17 11:13:19
3  -- Model: New Model Version: 1.0
4  -- MySQL Workbench Forward Engineering
5
6  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
7  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
8  SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';
9
10 -----
11 -- Schema Gestion_Academica
12 -----
13
14 -----
15 -- Schema Gestion_Academica
16 -----
17 CREATE SCHEMA IF NOT EXISTS `Gestion_Academica_BSG` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci ;
18 USE `Gestion_Academica_BSG` ;
19
20 -----
21 -- Table `Gestion_Academica`.`Ciclos`
22 -----
23 CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Ciclos` (
24   `Ciclos` VARCHAR(5) NOT NULL,
25   PRIMARY KEY (`Ciclos`))
26 ENGINE = InnoDB;
27
28 -----
29
30 -- Table `Gestion_Academica`.`Alumnos`
31 -----
32 CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Alumnos` (
33   `id` INT NOT NULL AUTO_INCREMENT,
34   `nombre` VARCHAR(150) NULL,
35   `email` VARCHAR(100) NOT NULL,
36   PRIMARY KEY (`id`),
37   /*UNIQUE INDEX `email_UNIQUE` (`email` ASC),*/
38   INDEX `idAlumnos` (`nombre` ASC))
39 ENGINE = InnoDB;
40
41
```

```

42  -- -----
43  -- Table `Gestion_Academica`.`Grupos`
44  -- -----
45  CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Grupos` (
46      `id` INT NOT NULL AUTO_INCREMENT,
47      `nombre` VARCHAR(45) NOT NULL,
48      `aula` VARCHAR(10) NULL,
49      `ciclo` VARCHAR(5) NOT NULL,
50      `idAlumno` INT NOT NULL,
51      PRIMARY KEY (`id`),
52      INDEX `fk_Grupos_Ciclos_idx` (`ciclo` ASC),
53      INDEX `idDelegado_idx` (`idAlumno` ASC),
54      CONSTRAINT `fk_Grupos_Ciclos`
55      FOREIGN KEY (`ciclo`)
56      REFERENCES `Gestion_Academica_BSG`.`Ciclos` (`Ciclos`)
57      ON DELETE NO ACTION
58      ON UPDATE NO ACTION,
59      CONSTRAINT `idDelegado`
60      FOREIGN KEY (`idAlumno`)
61      REFERENCES `Gestion_Academica_BSG`.`Alumnos` (`id`)
62      ON DELETE NO ACTION
63      ON UPDATE NO ACTION)
64  ENGINE = InnoDB;
65
66
67  -- -----
68  -- Table `Gestion_Academica`.`Asignaturas`
69  -- -----
70  CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Asignaturas` (
71      `id` INT NOT NULL,
72      `nombre` VARCHAR(45) NOT NULL,
73      `nhoras` VARCHAR(45) NULL,
74      PRIMARY KEY (`id`),
75      UNIQUE INDEX `nombre_UNIQUE` (`nombre` ASC))
76  ENGINE = InnoDB;
77
78
79  -- -----
80  -- Table `Gestion_Academica`.`Notas`
81  -- -----

```

```

80  -- Table `Gestion_Academica`.`Notas`
81  -----
82  CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Notas` (
83    `Alumnos_id` INT NOT NULL,
84    `Asignaturas_id` INT NOT NULL,
85    `id` INT NOT NULL,
86    `1Ev` INT NULL,
87    `2Ev` INT NULL,
88    `3Ev` INT NULL,
89    `Final` INT NULL,
90    PRIMARY KEY (`id`),
91    INDEX `fk_Alumnos_has_Asignaturas_Asignaturas1_idx` (`Asignaturas_id` ASC),
92    INDEX `fk_Alumnos_has_Asignaturas_Alumnos1_idx` (`Alumnos_id` ASC),
93    CONSTRAINT `fk_Alumnos_has_Asignaturas_Alumnos1`
94    FOREIGN KEY (`Alumnos_id`)
95    REFERENCES `Gestion_Academica_BSG`.`Alumnos` (`id`)
96    ON DELETE CASCADE
97    ON UPDATE NO ACTION,
98    CONSTRAINT `fk_Alumnos_has_Asignaturas_Asignaturas1`
99    FOREIGN KEY (`Asignaturas_id`)
100   REFERENCES `Gestion_Academica_BSG`.`Asignaturas` (`id`)
101   ON DELETE NO ACTION
102   ON UPDATE NO ACTION)
103 ENGINE = InnoDB;
104
105
106  -----
107  -- Table `Gestion_Academica`.`Delegado`
108  -----
109  CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Delegado` (
110    `id` INT NOT NULL AUTO_INCREMENT,
111    `idGrupos` INT NOT NULL,
112    `idAlumnos` INT NOT NULL,
113    PRIMARY KEY (`id`),
114    UNIQUE INDEX `id_UNIQUE` (`id` ASC),
115    INDEX `idGrupos_idx` (`idGrupos` ASC),
116    INDEX `idAlumnos_idx` (`idAlumnos` ASC),

```

```
115 INDEX `idGrupos_idx` (`idGrupos` ASC),
116 INDEX `idAlumnos_idx` (`idAlumnos` ASC),
117 CONSTRAINT `idGrupos`
118 FOREIGN KEY (`idGrupos`)
119 REFERENCES `Gestion_Academica_BSGCA`.`Grupos` (`id`)
120 ON DELETE NO ACTION
121 ON UPDATE NO ACTION,
122 CONSTRAINT `idAlumnos`
123 FOREIGN KEY (`idAlumnos`)
124 REFERENCES `Gestion_Academica_BSG`.`Alumnos` (`id`)
125 ON DELETE NO ACTION
126 ON UPDATE NO ACTION)
127 ENGINE = InnoDB;
128
129
130 -----
131 -- Table `Gestion_Academica`.`Profesores`
132 -----
133 CREATE TABLE IF NOT EXISTS `Gestion_Academica_BSG`.`Profesores` (
134   `id` INT NOT NULL AUTO_INCREMENT,
135   `nombre` VARCHAR(150) NULL,
136   PRIMARY KEY (`id`))
137 ENGINE = InnoDB;
138
139
140 -----
141 -- Table `Gestion_Academica`.`imparte`
142 -----
143 CREATE TABLE `imparte` (
144   `id` INT NOT NULL AUTO_INCREMENT,
145   `idGrupos` INT NOT NULL,
146   `idProfesores` INT NOT NULL,
147   `idAsignaturas` INT NOT NULL,
148   PRIMARY KEY (`id`),
149   INDEX `idGrupos_idx` (`idGrupos` ASC),
150   INDEX `idProfesores_idx` (`idProfesores` ASC),
151   INDEX `idAsignaturas_idx` (`idAsignaturas` ASC),
152   FOREIGN KEY (`idGrupos`) REFERENCES `Gestion_Academica_BSG`.`Grupos` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
153   FOREIGN KEY (`idProfesores`) REFERENCES `Gestion_Academica_BSG`.`Profesores` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
154   FOREIGN KEY (`idAsignaturas`) REFERENCES `Gestion_Academica_BSG`.`Asignaturas` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION)
155
```

Línea 106, Columna 57 — 155 líneas

INS SQL ● Espaci

Por otro lado, necesitamos a su vez de un script de inserción, que se anexará a la siguiente documentación.

Finalmente, crearemos los índices para nuestras bases de datos, y tendremos el proceso concluido.

