

Actividad 4 challenge 8

```
CREATE DATABASE IF NOT EXISTS `developers` /*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci
USE `developers`;
-- MySQL dump 10.13 Distrib 8.0.31, for Win64 (x86_64)
--
-- Host: localhost    Database: developers
-- -----
-- Server version    8.0.31

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!50503 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `hub`
--

DROP TABLE IF EXISTS `hub`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!50503 SET character_set_client = utf8 */;
CREATE TABLE `hub` (
  `id` tinyint unsigned NOT NULL AUTO_INCREMENT,
  `nombre` varchar(45) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
  `provincia` char(2) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_hub_provincia_idx` (`provincia`),
  CONSTRAINT `fk_hub_provincia` FOREIGN KEY (`provincia`) REFERENCES `provincia` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

```

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `hub`
--

LOCK TABLES `hub` WRITE;
/*!40000 ALTER TABLE `hub` DISABLE KEYS */;
INSERT INTO `hub` VALUES (1,'Campus Madrid by Google','28'),(
/*!40000 ALTER TABLE `hub` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `programador`
--

DROP TABLE IF EXISTS `programador`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!50503 SET character_set_client = utf8 */;
CREATE TABLE `programador` (
  `idProgramador` tinyint unsigned NOT NULL AUTO_INCREMENT,
  `dni` varchar(9) CHARACTER SET utf8 COLLATE utf8_spanish_ci,
  `nombreP` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci,
  `ap1` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci,
  `ap2` varchar(100) CHARACTER SET utf8 COLLATE utf8_spanish_ci,
  `salario` smallint unsigned NOT NULL,
  `hub` tinyint unsigned NOT NULL,
  `seccion` tinyint unsigned NOT NULL,
  PRIMARY KEY (`idProgramador`),
  UNIQUE KEY `dni` (`dni`),
  KEY `fk_programador_hub1_idx` (`hub`),
  KEY `fk_programador_seccion1_idx` (`seccion`),
  CONSTRAINT `fk1` FOREIGN KEY (`hub`) REFERENCES `hub` (`id`),
  CONSTRAINT `fk2` FOREIGN KEY (`seccion`) REFERENCES `seccion` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

--

```

```

-- Dumping data for table `programador`
--

LOCK TABLES `programador` WRITE;
/*!40000 ALTER TABLE `programador` DISABLE KEYS */;
INSERT INTO `programador` VALUES (1,'25498262W','Javier','Gal
/*!40000 ALTER TABLE `programador` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `provincia`
--

DROP TABLE IF EXISTS `provincia`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!50503 SET character_set_client = utf8 */;
CREATE TABLE `provincia` (
  `codigo` char(2) CHARACTER SET utf8 COLLATE utf8_spanish_ci
  `nombreProvincia` varchar(45) CHARACTER SET utf8 COLLATE ut
  PRIMARY KEY (`codigo`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8_spanish_
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `provincia`
--

LOCK TABLES `provincia` WRITE;
/*!40000 ALTER TABLE `provincia` DISABLE KEYS */;
INSERT INTO `provincia` VALUES ('01','Álava'),('02','Albacete
/*!40000 ALTER TABLE `provincia` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `seccion`
--

DROP TABLE IF EXISTS `seccion`;

```

```

/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!50503 SET character_set_client = utf8 */;
CREATE TABLE `seccion` (
  `idSeccion` tinyint unsigned NOT NULL AUTO_INCREMENT,
  `nombreS` varchar(99) NOT NULL,
  `partidaEconomica` double unsigned NOT NULL,
  `desembolso` double unsigned NOT NULL,
  PRIMARY KEY (`idSeccion`)
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8 COLLATE=
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `seccion`
--

LOCK TABLES `seccion` WRITE;
/*!40000 ALTER TABLE `seccion` DISABLE KEYS */;
INSERT INTO `seccion` VALUES (1,'Frontend',230000,7000),(2,'B
/*!40000 ALTER TABLE `seccion` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2023-02-28 10:52:01

```

developersok(1).sql

1. Seleccionar el nombre de la sección con la menor partida económica:

```
SELECT nombres
FROM seccion
WHERE partidaEconomica = (SELECT MIN(partidaEconomica) FROM seccion)
```

1. Seleccionar el nombre de las secciones cuyo desembolso sea menor que el desembolso medio de todas las secciones:

```
SELECT nombres
FROM seccion
WHERE desembolso < (SELECT AVG(desembolso) FROM seccion);
```

The screenshot shows the MySQL Workbench interface. The 'Schemas' panel on the left lists various databases, with 'developers' expanded to show 'Tables'. The 'Query Editor' in the center contains the following SQL query:

```
1 • SELECT nombres
2 FROM seccion
3 WHERE desembolso < (SELECT AVG(desembolso) FROM seccion);
4
```

The 'Result Grid' at the bottom displays the results of the query. It shows 6 rows of data:

#	nombres
1	Frontend
2	Backend
3	Full Stack
4	Centro de Atención al Usuario
5	Bases de Datos
6	Pruebas

Below the result grid, the 'Action Output' panel shows a list of actions performed during the session, including setting foreign key checks, unique checks, character set client, character set results, collation connection, SQL notes, and using the 'developers' schema. The final action is the execution of the query, which returned 6 rows.

1. Seleccionar el nombre completo del programador con el mayor salario:

```
SELECT nombreP, ap1, ap2
FROM programador
WHERE salario = (SELECT MAX(salario) FROM programador);
```

The screenshot shows the MySQL Workbench interface. The 'Query Editor' window displays the following SQL query:

```
1 • SELECT nombreP, ap1, ap2
2 FROM programador
3 WHERE salario = (SELECT MAX(salario) FROM programador);
```

The 'Result Grid' window shows the results of the query:

#	nombreP	ap1	ap2
1	David	Santana	NULL

The 'Action Output' window shows the execution log, including the following message:

#	Time	Action	Message
63	03:02:03	SELECT nombreP, ap1, ap2 FROM programador WHERE s...	1 row(s) returned

2. Seleccionar el nombre completo de los programadores cuyo salario sea mayor que el salario medio de todos los desarrolladores que trabajan en Madrid:

```
SELECT p.nombreP, p.ap1, p.ap2
FROM programador p
JOIN hub h ON p.hub = h.id
```

```
WHERE p.salario > (SELECT AVG(salario) FROM programador pr
AND h.provincia = '28';
```

The screenshot shows the MySQL Workbench interface. The 'Query Editor' window displays the following SQL query:

```
1 • SELECT p.nombreP, p.ap1, p.ap2
2 FROM programador p
3 JOIN hub h ON p.hub = h.id
4 WHERE p.salario > (SELECT AVG(salario) FROM programador pr JOIN hub hb
5 AND h.provincia = '28');
```

The 'Result Grid' shows the following data:

#	nombreP	ap1	ap2
1	Javier	Galet	de las Heras
2	Juan Carlos	Martín	Carrasco
3	Rocío	Sánchez	Ferrero
4	Paula	Rodríguez	Terrón

The 'Action Output' window shows the following messages:

#	Time	Action	Message
57	03:00:20	/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARAC...	0 row(s) affected
58	03:00:20	/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARA...	0 row(s) affected
59	03:00:20	/*!40101 SET COLLATION_CONNECTION=@OLD_COLLAT...	0 row(s) affected
60	03:00:20	/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */	0 row(s) affected
61	03:00:41	use developers	0 row(s) affected
62	03:01:00	SELECT nombreS FROM seccion WHERE desembolso < (S...	6 row(s) returned
63	03:02:03	SELECT nombreP, ap1, ap2 FROM programador WHERE s...	1 row(s) returned
64	03:02:54	SELECT p.nombreP, p.ap1, p.ap2 FROM programador p J...	4 row(s) returned

1. Seleccionar el nombre completo del programador cuyo salario sea menor que el salario medio de todos los programadores que trabajan en Backend o Frontend:

```
SELECT nombreP, ap1, ap2
FROM programador
WHERE salario < (SELECT AVG(salario) FROM programador WHERE
```

MySQL Workbench

Local instance 3306

File Edit View Query Database Server Tools Scripting Help

Schemas

SCHEMAS

Filter objects

- abylatella
- Alminarock
- ceutacar
- ✓ developers
 - Tables
 - Views
 - Stored Procedures
 - Functions
- marinadog618
- ✓ smallCompany
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sys
- vboxuser

Query 8

Limit to 1000 rows

```

1 • SELECT nombreP, ap1, ap2
2 FROM programador
3 WHERE salario < (SELECT AVG(salario) FROM programador WHERE seccion IN

```

Result Grid

#	nombreP	ap1	ap2
1	Javier	Galet	de las Heras
2	Juan Carlos	Martin	Carrasco
3	Ángel	Vilchez	NULL
4	José María	Ostale	Méndez
5	José Ángel	Valriberas	Pizarro
6	Julio	Mata	Guerrero

programador 4

Action Output

#	Time	Action	Message
✓ 58	03:00:20	/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS;*/	0 row(s) affected
✓ 59	03:00:20	/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION;*/	0 row(s) affected
✓ 60	03:00:20	/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */	0 row(s) affected
✓ 61	03:00:41	use developers	0 row(s) affected
✓ 62	03:01:00	SELECT nombres FROM seccion WHERE desembolso < (\$...)	6 row(s) returned
✓ 63	03:02:03	SELECT nombreP, ap1, ap2 FROM programador WHERE s...	1 row(s) returned
✓ 64	03:02:54	SELECT p.nombreP, p.ap1, p.ap2 FROM programador p J...	4 row(s) returned
✓ 65	03:03:33	SELECT nombreP, ap1, ap2 FROM programador WHERE s...	10 row(s) returned

Object Info Session

0 object selected

Query Completed

CTRL DERECHA

2. Seleccionar la suma de todos los salarios cuyo hub se encuentre en Málaga:

```

SELECT SUM(salario)
FROM programador p
JOIN hub h ON p.hub = h.id
WHERE h.provincia = (SELECT codigo FROM provincia WHERE no

```


MySQL Workbench

Local instance 3306

File Edit View Query Database Server Tools Scripting Help

Schemas

SCHEMAS

Filter objects

- abylatella
- Alminarock
- ceutacar
- ✓ developers
 - Tables
 - Views
 - Stored Procedures
 - Functions
- marinadog618
- ✓ smallCompany
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sys
- vboxuser

Query 8

```

1 • SELECT SUM(salario)
2 FROM programador p
3 JOIN hub h ON p.hub = h.id
4 WHERE h.provincia = (SELECT codigo FROM provincia WHERE nombreProvincia = 'Barcelona')
5

```

Limit to 1000 rows

Result Grid

#	SUM(salario)
1	11750

Result 6

Action Output

	Message	Duration / Fetch
ap2SELECT SUM(salario) FROM pr...	FROM programador p JOIN hub h ON p.hub = h.id WHERE h.provincia = (' at line 1 Error Code: 1064. You have an error in your SQL s...	0.00013 sec
ap2 SELECT SUM(salario) FROM p...	FROM programador p JOIN hub h ON p.hub = h.id WHERE h.provin' at line 1	0.00010 sec
FROM programador p JOIN hub h O...	1 row(s) returned	0.0023 sec / 0.000
FROM programador p JOIN hub h O...	1 row(s) returned	0.00037 sec / 0.00

Object Info Session

no object selected

Query Completed

CTRL DERECHA