

IISI-LAB-Examen-2020-01-RESUELTO...



DonPitoCorleone



Introducción a la Ingeniería del Software y los Sistemas de Información I



2º Grado en Ingeniería Informática - Tecnologías Informáticas



**Escuela Técnica Superior de Ingeniería Informática
Universidad de Sevilla**

MÁSTER EN

**Energías
Renovables**

MADRID

Ahora
25%
DE DESCUENTO

EOI Escuela de
organización
industrial

Estudia el máster líder en
energías renovables según el

**Ranking 250
Masters de:**

EL MUNDO Expansión

Info y descuentos



SIGUE ESTUDIANDO, HASTA LLEGAR A SER QUIÉN SIEMPRE HAS QUERIDO.

TI

IISSE-1 Prueba de Laboratorio. Sesión 1	Curso 2019-20 Enero de 2020
Apellidos, Nombre:	Grupo:

Pregunta 0. (1 punto)

Utilice HeidiSQL para establecer una conexión a localhost con usuario "root" y sin clave (contraseña en blanco) que se usará para realizar todos los ejercicios de este examen. Cree una nueva base de datos llamada <<UVUS>>_sesion1. Ejecute el script create_db en la base de datos creada. Para asegurar que todo es correcto ejecute la consulta SELECT count(*) FROM Students; y compruebe que el resultado que devuelve es 21.

Pregunta 1. (2 puntos)

Añada el requisito de información **Beca**. Una beca es una ayuda económica que recibe un estudiante para realizar sus estudios en un año académico particular. Sus atributos son: el estudiante beneficiario de la beca, la cuantía de la beca, el año académico, la fecha de inicio y la duración en meses. Hay que tener en cuenta las siguientes restricciones:

- Un estudiante sólo puede ser beneficiario de una beca para un año académico, pero puede tener varias si son en distintos cursos académicos.
- La cuantía de la beca no puede ser inferior a 500€ ni superior a 2500€.
- Todos los atributos son obligatorios salvo la fecha de inicio y la duración.

Pregunta 2. (1 punto)

Cree y ejecute un procedimiento almacenado llamado pInsertGrants() que cree las siguientes becas:

- Beca de 500€, de 3 meses, en el año 2019, para el estudiante con ID=5.
- Beca de 700€, de 8 meses, en el año 2019, para el estudiante con ID=6.
- Beca de 1000€, que comienza el 01/01/2019, en el año 2019, para el estudiante con ID=7.

Pregunta 3. (1 punto)

Cree un procedimiento almacenado llamado pUpdateGrants(s, a) que actualiza la cuantía de las becas del estudiante con ID=s con el valor a. Ejecute la llamada a pUpdateGrants(5,600)

Cree un procedimiento almacenado llamado pDeleteGrants(s) que elimina las becas del estudiante con ID=s. Ejecute la llamada pDeleteGrants(7)



**TODOS
NECESITAMOS
UNA SEÑAL
PARA SEGUIR**



AQUARIUS

*AQUARIUS ES FUENTE DE ZINC Y SELENO

AQUARIUS es una marca registrada de The Coca-Cola Company.®

WUOLAH

Pregunta 4. (1 punto)

Cree una consulta que devuelva el nombre del grado, el nombre de la asignatura, el número de créditos de la asignatura y su tipo, para todas las asignaturas que pertenecen a todos los grados. Ordene los resultados por el nombre del grado. Un ejemplo de resultado de esta consulta es el siguiente:

Resultado #1 (4x13)			
degreeName	subjectName	credits	type
Ingeniería de Computadores	Introducción a la Matematica Discreta	6	Formacion Basica
Ingeniería de Computadores	Redes de Computadores	6	Obligatoria
Ingeniería de Computadores	Teoría de Grafos	6	Obligatoria
Ingeniería de Computadores	Aplicaciones de Soft Computing	6	Optativa
Ingeniería del Software	Diseño y Pruebas	12	Obligatoria
Ingeniería del Software	Acceso Inteligente a la Informacion	6	Optativa
Ingeniería del Software	Optimizacion de Sistemas	6	Optativa
Ingeniería del Software	Ingeniería de Requisitos	6	Obligatoria
Ingeniería del Software	Análisis y Diseño de Datos y Algoritmos	12	Obligatoria
Tecnologías Informáticas	Fundamentos de Programación	12	Formacion Basica
Tecnologías Informáticas	Lógica Informatica	6	Optativa
Tecnologías Informáticas	Gestión y Estrategia Empresarial	12	Optativa
Tecnologías Informáticas	Trabajo de Fin de Grado	12	Obligatoria

Pregunta 5. (1 punto)

Cree una consulta que devuelva la nota media del grupo cuyo ID=2. Un ejemplo de resultado de esta consulta es el siguiente:

Resultado #1 (1x1)	
averageGradeGroup2	3,750000

Pregunta 6. (1 punto)

Cree una consulta que devuelva la nota media de cada grupo, ordenados por su nota media de mayor a menor. Un ejemplo de resultado de esta consulta es el siguiente:

grades (2x5)	
groupId	averageGrade
7	9,950000
10	8,750000
1	3,875000
2	3,750000
18	0,583333

Pregunta 7. (1 punto)

Cree una consulta que devuelva la nota más alta y la nota más baja de cada grupo de Teoría de la asignatura con ID=1. Un ejemplo del resultado de esta consulta es el siguiente:

groups (3x2)		
groupId	maxGrade	minGrade
1	4,50	3,25
2	5,00	2,50

Pregunta 8. (1 punto)

Cree una consulta que devuelva el nombre y los apellidos del estudiante que ha sacado la nota más alta del grupo con ID=10. Un ejemplo del resultado de esta consulta es el siguiente:

students (2x1)	
firstName	surname
Rafael	Ramírez

SIGUE ESTUDIANDO, HASTA LLEGAR A SER QUIÉN SIEMPRE HAS QUERIDO.

IISSI-1 Prueba de Laboratorio. Sesión 1	Curso 2019-20 Enero de 2020
Apellidos, Nombre:	Grupo:

Pregunta 0. (1 punto)

Utilice HeidiSQL para establecer una conexión a localhost con usuario "root" y sin clave (contraseña en blanco) que se usará para realizar todos los ejercicios de este examen. Cree una nueva base de datos llamada <<UVUS>>_sesion1. Ejecute el script create_db en la base de datos creada. Para asegurar que todo es correcto ejecute la consulta `SELECT count(*) FROM Students;` y compruebe que el resultado que devuelve es 21.

```
CREATE DATABASE if NOT EXISTS UVUS_sesion1;
```

```
USE UVUS_sesion1;
```

```
SELECT COUNT(*) FROM estudiantes;
```

```
COUNT(*)
```

```
21
```

Pregunta 1. (2 puntos)

Añada el requisito de información **Beca**. Una beca es una ayuda económica que recibe un estudiante para realizar sus estudios en un año académico particular. Sus atributos son: el estudiante beneficiario de la beca, la cuantía de la beca, el año académico, la fecha de inicio y la duración en meses. Hay que tener en cuenta las siguientes restricciones:

- Un estudiante sólo puede ser beneficiario de una beca para un año académico, pero puede tener varias si son en distintos cursos académicos.
- La cuantía de la beca no puede ser inferior a 500€ ni superior a 2500€.
- Todos los atributos son obligatorios salvo la fecha de inicio y la duración.

```
DROP TABLE if EXISTS Beca;
```

```
CREATE OR REPLACE TABLE Beca(
```

```
    BecaId INT NOT NULL AUTO_INCREMENT,  
    estudianteId INT NOT NULL,  
    cuantia INT NOT null,  
    anyo YEAR NOT null,  
    fechaInicio DATE,  
    meses INT,  
    UNIQUE (estudianteId, anyo),  
    PRIMARY KEY (becaId),  
    FOREIGN KEY (estudianteId) REFERENCES estudiantes(estudianteId),  
    CONSTRAINT cuantiaR CHECK(cuantia>=500 AND cuantia<=2500)
```

```
);
```



TODOS
NECESITAMOS
UNA SEÑAL
PARA SEGUIR



AQUARIUS

*AQUARIUS ES FUENTE DE ZINC Y SELENO

AQUARIUS es una marca registrada de The Coca-Cola Company.®

WUOLAH

Pregunta 2. (1 punto)

Cree y ejecute un procedimiento almacenado llamado pInsertGrants() que cree las siguientes becas:

- Beca de 500€, de 3 meses, en el año 2019, para el estudiante con ID=5.
- Beca de 700€, de 8 meses, en el año 2019, para el estudiante con ID=6.
- Beca de 1000€, que comienza el 01/01/2019, en el año 2019, para el estudiante con ID=7.

```
DELIMITER //
CREATE OR REPLACE PROCEDURE pInsertGrants(estId INT, c INT, anyo YEAR,
| | | | | | | | | | fecha DATE, duracion int)
BEGIN
    INSERT INTO beca(estudianteId,cuantidad, anyo, fechaInicio, meses)
    VALUES (estId, c, anyo, fecha, duracion);
END //
DELIMITER ;

CALL pInsertGrants(5, 500, 2019, NULL, 3);
CALL pInsertGrants(6, 700, 2019, NULL, 8);
CALL pInsertGrants(7, 1000, 2019, '2019/01/01', NULL);
```

Pregunta 3. (1 punto)

Cree un procedimiento almacenado llamado pUpdateGrants(s, a) que actualiza la cuantía de las becas del estudiante con ID=s con el valor a. Ejecute la llamada a pUpdateGrants(5,600)

Cree un procedimiento almacenado llamado pDeleteGrants(s) que elimina las becas del estudiante con ID=s. Ejecute la llamada pDeleteGrants(7)

```
DELIMITER //
CREATE OR REPLACE PROCEDURE pUpdateGrants(estId INT, dinero int)
BEGIN
    UPDATE beca
    SET cantidad = dinero
    WHERE estId = estudianteId;
END //
DELIMITER ;

DELIMITER //
CREATE OR REPLACE PROCEDURE pDeleteGrants(estId INT)
BEGIN
    DELETE FROM beca
    WHERE estudianteId = estId;
END //
DELIMITER ;

CALL pUpdateGrants(5, 600);
CALL pDeleteGrants(7);
```



***SIGUE ESTUDIANDO,
HASTA LLEGAR A SER
QUIÉN SIEMPRE
HAS QUERIDO.***



AQUARIUS®

TODOS NECESITAMOS UNA SEÑAL PARA SEGUIR

*AQUARIUS ES FUENTE DE ZINC Y SELENIO

AQUARIUS es una marca registrada de The Coca-Cola Company®

Pregunta 4. (1 punto)

Cree una consulta que devuelva el nombre del grado, el nombre de la asignatura, el número de créditos de la asignatura y su tipo, para todas las asignaturas que pertenecen a todos los grados. Ordene los resultados por el nombre del grado. Un ejemplo de resultado de esta consulta es el siguiente:

Resultado #1 (4x13)			
degreeName	subjectName	credits	type
Ingeniería de Computadores	Introducción a la Matematica Discreta	6	Formacion Basica
Ingeniería de Computadores	Redes de Computadores	6	Obligatoria
Ingeniería de Computadores	Teoría de Grafos	6	Obligatoria
Ingeniería de Computadores	Aplicaciones de Soft Computing	6	Optativa
Ingeniería del Software	Diseño y Pruebas	12	Obligatoria
Ingeniería del Software	Acceso Inteligente a la Informacion	6	Optativa
Ingeniería del Software	Optimizacion de Sistemas	6	Optativa
Ingeniería del Software	Ingeniería de Requisitos	6	Obligatoria
Ingeniería del Software	Análisis y Diseño de Datos y Algoritmos	12	Obligatoria
Tecnologías Informáticas	Fundamentos de Programación	12	Formacion Basica
Tecnologías Informáticas	Lógica Informatica	6	Optativa
Tecnologías Informáticas	Gestión y Estrategia Empresarial	12	Optativa
Tecnologías Informáticas	Trabajo de Fin de Grado	12	Obligatoria

```
SELECT G.nombre, A.nombre, credits, tipo
FROM asignaturas A JOIN grados G ON A.gradoId = G.gradoId
ORDER BY G.nombre;
```

Pregunta 5. (1 punto)

Cree una consulta que devuelva la nota media del grupo cuyo ID=2. Un ejemplo de resultado de esta consulta es el siguiente:

Resultado #1 (1x1)	
averageGradeGroup2	
3,750000	

```
SELECT AVG(valor) FROM calificaciones WHERE grupoId = 2;
```


SIGUE ESTUDIANDO, HASTA LLEGAR A SER QUIÉN SIEMPRE HAS QUERIDO.

Pregunta 6. (1 punto)

Cree una consulta que devuelva la nota media de cada grupo, ordenados por su nota media de mayor a menor. Un ejemplo de resultado de esta consulta es el siguiente:

groupId	averageGrade
7	9,950000
10	8,750000
1	3,875000
2	3,750000
18	0,583333

```
SELECT groupId, AVG(valor) AS notaMedia
FROM calificaciones
GROUP BY groupId
ORDER BY notaMedia desc;
```

Pregunta 7. (1 punto)

Cree una consulta que devuelva la nota más alta y la nota más baja de cada grupo de Teoría de la asignatura con ID=1. Un ejemplo del resultado de esta consulta es el siguiente:

groupId	maxGrade	minGrade
1	4,50	3,25
2	5,00	2,50

```
SELECT groupId, MAX(valor), MIN(valor)
FROM calificaciones
NATURAL JOIN grupos JOIN asignaturas A
ON grupos.asignaturaId = A.asignaturaId WHERE
(actividad = 'Teoría' AND A.asignaturaId = 1)
GROUP BY groupId;
```

Pregunta 8. (1 punto)

Cree una consulta que devuelva el nombre y los apellidos del estudiante que ha sacado la nota más alta del grupo con ID=10. Un ejemplo del resultado de esta consulta es el siguiente:

firstName	surname
Rafael	Ramírez

```
SELECT E.nombre, E.apellido FROM estudiantes E
JOIN calificaciones C ON C.estudianteId = E.estudianteId
WHERE groupId = 10
ORDER by valor DESC
LIMIT 1;
```



TODOS
NECESITAMOS
UNA SEÑAL
PARA SEGUIR



AQUARIUS

*AQUARIUS ES FUENTE DE ZINC Y SELENO

AQUARIUS es una marca registrada de The Coca-Cola Company.®

WUOLAH