

Normativa

1. La práctica se realizará en equipos con un mínimo de 4 alumnos y un máximo 5 alumnos. No obstante, el tamaño final de los equipos será fijado por el profesor de prácticas en función de las necesidades académicas de cada grupo de matrícula.
2. Todos los integrantes del equipo de prácticas tienen que estar **matriculados en el mismo grupo de clase**.
3. La práctica consiste en 3 apartados distintos: 1) modelado del sistema, 2) creación y explotación de la base de datos y 3) programación contra la base de datos.
4. La evaluación de la práctica se realizará en el aula por parte del profesorado de la asignatura durante la última semana de curso (semana del 16 de diciembre). Es obligatoria la presencia en el aula y responder a las preguntas planteadas por el profesor el día de la evaluación. Los alumnos que no estén presentes recibirán una calificación de “No presentado”. La práctica se evaluará una única vez.
5. Los profesores se reservan el derecho a realizar una evaluación individualizada de la práctica a los estudiantes de la asignatura con el fin de comprobar la participación de cada estudiante en el equipo de prácticas.
6. Los resultados de la realización de la práctica deberán ser entregados a través de Moodle antes de su evaluación en un formato electrónico.
7. De acuerdo con la Normativa de Evaluación de la Universidad Politécnica de Madrid, la comprobación de fraude académico durante la realización de la práctica supondrá la obtención de una calificación de cero en la misma y podrá implicar consecuencias más severas en función de la gravedad de los hechos.

Enunciado

En el año 2912, la humanidad ha ampliado sus horizontes más allá de la Tierra, estableciendo colonias en diversos planetas y lunas del universo. Las ciudades espaciales flotantes se han transformado en centros vibrantes de investigación y comercio, donde la vida cotidiana se entrelaza con tecnología avanzada y una rica diversidad cultural. Viajar entre planetas se ha vuelto algo habitual, y la exploración de nuevos mundos se considera una prioridad, aunque no sin desafíos. A pesar de los avances, persisten tensiones sociales y desigualdades, ya que algunos disfrutan de las comodidades del espacio mientras otros enfrentan la dura realidad de la vida en las periferias. Sin embargo, cada nuevo descubrimiento alimenta la esperanza de un futuro más unido y sostenible para la humanidad.

En la sede de la Agencia Espacial Europea, se ha recibido un mensaje alarmante de una fuente situada a cuatro años luz de distancia. La comunicación revela una situación preocupante relacionada con la Nave Espacial Titanic, un transatlántico interestelar que partió hace un mes con más de 8.000 pasajeros a bordo. La nave se encontraba en su viaje inaugural, transportando emigrantes de nuestro sistema solar hacia tres exoplanetas recién habitables que orbitan en estrellas cercanas.

Mientras rodeaba Alfa Centauri en ruta a su primer destino, el cálido 55 Cancri E, la Nave Espacial Titanic colisionó con una anomalía del espacio-tiempo oculta dentro de una nube de polvo. Aunque la nave permaneció intacta, casi la mitad de los pasajeros fueron transportados a una dimensión alterna.

Para ayudar a los equipos de rescate y recuperar a los pasajeros perdidos, es crucial restaurar el sistema informático de la nave, especialmente su base de datos, que contiene información detallada sobre cada pasajero. En el momento del embarque, cada pasajero realizó

un *check-in*, donde se le asignó un identificador único tras proporcionar su nombre, edad, planeta natal y la documentación que acreditaba su estatus como VIP. Además, quienes viajaban con menores debieron indicar quién se encargaría de tutelar a cada niño en caso de cualquier eventualidad.

Dentro de la Nave Espacial Titanic, cada pasajero fue acomodado en una cabina. Estas cabinas, identificadas por un número y el lado en el que se encuentran (P o S), están situadas en las diferentes cubiertas de las que dispone la nave, las cuales se identifican por una única letra (A, B, C, ...) y pertenecen a primera, segunda o tercera clase. El confort de los pasajeros era primordial, por lo que las cabinas eran limpiadas por robots de última generación que se movían a través de una compleja red de túneles ocultos en los falsos techos. Cada robot tenía un número único y un diseño específico según las tareas que debía realizar, como limpiar espejos o lavar sábanas. Estos robots, altamente eficientes, eran asignados a la limpieza de varias cabinas, y cualquier avería que sufrieran los robots era registrada, junto con la fecha de la misma, por los mecánicos de la nave, quienes también contaban con identificadores únicos otorgados en el momento del embarque al presentar documentación acreditativa de su nombre.

La Nave Espacial Titanic transportaba a sus pasajeros a uno de los siguientes destinos: TRAPPIST-1e, PS0 J318.5-22 y 55 Cancri e. Estos planetas se hallan en diferentes sistemas de la Vía Láctea, y a pesar de los esfuerzos de los astrónomos por asignar nombres únicos, algunos planetas comparten nombres en distintos sistemas. Además, algunos planetas tienen satélites cuyos nombres también pueden repetirse, siempre que no pertenezcan al mismo planeta. Para garantizar un descenso suave en la atmósfera de cada destino, la base de datos de la nave contiene los elementos químicos que componen la atmósfera de cada planeta (de cada elemento químico su nombre único, su símbolo y su peso atómico), permitiendo que la inteligencia artificial a cargo de la navegación ajuste velocidades y ángulos de entrada en función de cada atmósfera con el fin de asegurar la seguridad y comodidad de los pasajeros. Asimismo, la inteligencia artificial requiere consultar la masa y el radio del planeta, por lo que esta información también se encuentra almacenada en la base de datos.

A pesar de contar con los últimos avances en propulsión espacial, la Nave Espacial Titanic debe enfrentar un largo viaje hacia sus destinos. Algunos pasajeros optaron por entrar en crio-sueño en sus cabinas, mientras que otros prefirieron mantenerse despiertos. Para estos últimos, la nave ofrece diversas opciones de entretenimiento, como un spa, un centro comercial, servicio a las habitaciones y una zona de restaurantes donde pueden gastar su dinero. El gasto total de cada pasajero en las distintas zonas de entretenimiento es almacenado en la base de datos. Estos espacios de entretenimiento son gestionados por el personal humano de la nave, que, al igual que los mecánicos, tiene un identificador único asignado tras aportar documentación acreditativa de su nombre durante el embarque. Dentro del personal, existe una jerarquía organizativa que permite una operativa eficiente, donde cada miembro solo puede tener un jefe.

Finalmente, tras la colisión con la anomalía del espacio-tiempo, se llevó a cabo un registro completo de la nave para determinar qué pasajeros habían desaparecido y cuáles permanecían a bordo.

Apartado 1

Completar las siguientes tareas:

1. Definir el modelado conceptual que cumpla con los requisitos escritos en el enunciado de la práctica. Será obligatorio utilizar el modelo Entidad-Relación con notación Chen, siendo motivo de suspenso si se utiliza otro modelado.

2. Definir todos los elementos de “Semántica no contemplada” que se detecten.
3. Definir los dominios de todos los atributos que se incluyan en las entidades pertenecientes a los **pasajeros**, **cabinas** y **cubiertas**.
4. Realizar el paso a tablas del Modelo Entidad-Relación generado.

Apartado 2

Completar las siguientes tareas:

1. Define las instrucciones SQL para crear todas las tablas necesarias del paso a tablas que se realizó en el Apartado 1.4. Presta especial atención a las configuraciones de integridad referencial para garantizar que se cumplen todas las restricciones planteadas en el enunciado de la práctica.
2. Una vez restaurada la base de datos, la Agencia Espacial Europea desea automatizar algunas consultas y nos pide que nos centremos en los procesos solamente de algunas tablas. Estas tablas son las que afectan a los cabinas, cubiertas, planetas, sistemas, pasajeros, entretenimientos y los gastos de los pasajeros en dichas zonas de entretenimiento. Estas tablas han sido nombradas respectivamente: **cabinas**, **cubiertas**, **planetas**, **sistemas**, **pasajeros**, **entretenimientos** y **gastos**. Se pide¹:
 - a) Usando el fichero denominado `titanic_spaceship.sql` disponible en Moodle, crear la base de datos `titanic_spaceship` y cargar todos los datos disponibles en dicho fichero en las tablas mencionadas y que van a ser objeto de procesos en puntos posteriores.
 - b) Estudiar, mediante ingeniería inversa, la estructura propuesta en este fichero, así como sus relaciones y atributos, y realizar el modelo relacional incluido en esta estructura.
 - c) Resolver en SQL la consulta: “Obtener el listado de los pasajeros que han embarcado obteniendo para cada uno su nombre, cabina asignada (con formato: Letra Cubierta-Número Cabina-Lado Cabina, por ejemplo “B-1-P”) y planeta destino al que viajaban (incluyendo el sistema al que pertenece), de aquellos pasajeros cuya cabina esta localizada en una cubierta de tercera clase desde la letra A a la D y que no están en criosueño”.
 - d) Resolver en SQL la consulta: “Obtener para cada zona de entretenimiento el gasto total que han tenido los pasajeros VIP en ella mostrando el nombre de la zona, el gasto total y el tipo de pasajero (“VIP”), haciendo lo mismo con el gasto correspondiente a los pasajeros que no sean VIP para los cuales se mostrará el valor “NO VIP” en el tipo de pasajero. Tipo de pasajero se mostrará como una columna que podrá tomar los dos posibles valores mencionados anteriormente (“VIP” o “NO VIP”)”.
 - e) Resolver en SQL la consulta: “Obtener el nombre de los planetas, y el sistema al que pertenecen, de aquellos planetas en los que han nacido menos pasajeros que el promedio de nacimientos en todos los planetas”.

¹Se ha verificado que todas las consultas que se piden pueden ser realizadas con la base de datos proporcionada. Igualmente se garantiza que todas las consultas requeridas devuelven al menos un resultado con el conjunto de datos provisto. Aunque el tiempo de ejecución para todas consultas es “breve”, esto puede variar en función de la implementación realizada. Si una consulta tarda demasiado tiempo (por ejemplo, más de 5 minutos), es porque se está desarrollando una solución poco eficiente de la misma. Reflexionar sobre este hecho en caso de producirse para encontrar una solución adecuada.

- f) Resolver en SQL la consulta: “Obtener el nombre de los pasajeros y su gasto total por zona de entretenimiento de aquellos pasajeros alojados en la cubierta ‘A’ y que han realizado gastos en todas las zonas de entretenimiento, ordenados por nombre de pasajero en orden alfabético y por cada pasajero que aparezcan los gastos de mayor importe primero”.
- g) Resolver en SQL la consulta: “Obtener las cabinas (mostrando la información de cubierta, número y lado) ocupadas por pasajeros que han gastado en el entretenimiento más popular, es decir en aquel en el que han participado más pasajeros”.
- h) Resolver en SQL la consulta: “Obtener los nombres de los planetas, así como el sistema al que pertenecen, de aquellos planetas a los que no han viajado pasajeros VIP y que además son los planetas donde han nacido alguno de los pasajeros con mayor edad”.
- i) Resolver en SQL la consulta: “Obtener aquellas cabinas (mostrando la información de cubierta, número y lado) con más de 3 pasajeros NO VIP y cuyos pasajeros han gastado más del doble del gasto promedio total por cabina”.
- j) Resolver en SQL la consulta: “Obtener los planetas donde la cantidad de pasajeros nacidos en él, y que han gastado en al menos 4 tipos diferentes de entretenimiento, es mayor al 25 % del total de pasajeros de dicho planeta, mostrando además del nombre del planeta y sistema al que pertenece, tanto el total de pasajeros nacidos en él, así como el número de pasajeros del mismo que gastaron en los tipos de entretenimiento”.
- k) Debido al gran impacto que ha generado la anomalía espacio-temporal que afectó a la Nave Espacial Titanic entre diversas agencias espaciales, la Agencia Espacial Europea ha decidido llevar a cabo una investigación exhaustiva sobre el suceso. Para ello, ha contratado a los mejores científicos de datos del universo, con el objetivo de desarrollar un modelo predictivo que permita determinar si, en caso de una nueva anomalía, un pasajero será teletransportado o no. Para facilitar su labor, se ha creado una vista que les permitirá consultar los datos necesarios para su análisis.

Se pide crear una nueva vista denominada **entretenimiento_pasajeros** que contenga los siguientes datos:

- id, nombre, y edad de cada pasajero.
- Si el pasajero fue o no transportado.
- nombre y sistema del planeta al que está viajando el pasajero.
- numero, lado y cubierta de la cabina en la que está hospedado y la clase de la cubierta asociada.
- nombre del entretenimiento en el que más ha gastado el pasajero, junto con la cantidad gastada en dicho entretenimiento.

Además, se pide crear un usuario nuevo conocido como **analista_datos** y restringir el acceso a dicho usuario para que únicamente pueda acceder a la vista en modo lectura.

- l) Para facilitar una identificación más rápida y precisa de los tutores legales de los menores en cada cabina, se ha determinado la necesidad de añadir un parámetro adicional en el registro de cada pasajero. Este parámetro permitirá identificar al tutor legal en caso de que el pasajero sea menor de edad. Para implementar este cambio, es necesario actualizar la tabla **pasajeros** y añadir un nuevo campo denominado **tutor**.

Tras modificar la estructura de la tabla, se desarrollará un procedimiento para asignar automáticamente el tutor de cada menor. Este procedimiento tomará en cuenta que el tutor de cada pasajero menor de 18 años debe ser un adulto de

mayor edad que se encuentre en la misma cabina (con la misma letra, número y lado) que el menor. Si no hubiera un adulto en la misma cabina que pueda cumplir esta función, el menor deberá colocarse en criosueño, salvo que ya se encuentre en esta condición. Los menores que ya estén en criosueño no requerirán asignación de tutor, permaneciendo en esta condición sin cambios adicionales.

- m) Se quiere crear una función que permita obtener el porcentaje de pasajeros que viajan a un destino concreto (sistema y planeta), para una cubierta determinada, cuyos datos pasaremos como parámetro. A fin de comprobar su correcto funcionamiento, crear una consulta que muestre la ocupación de todas las cubiertas disponibles para el destino “55 Cancri e” sistema “Copernico” ordenada alfabéticamente por la letra de la cubierta.
- n) Para evitar que pueda haber algún ciber-delincuente en alguno de los entretenimientos ofrecidos por en la nave, se quieren crear tantos triggers como sean necesarios para evitar que existan gastos en entretenimientos con saldo negativo, es decir, la cantidad en gastos sea menor que 0. En caso de producirse el ciber-ataque, deberá mostrarse el siguiente mensaje de error: “*ERROR: Bienvenido ciber-delincuente, pero no puedes sacar dinero en un entretenimiento*”.
- ñ) Para mejorar la satisfacción de los pasajeros, se ha decidido asignar automáticamente la condición de VIP a cualquier pasajero cuyo gasto total en entretenimientos supere la cantidad de 5000. Actualice la condición de VIP a todos los pasajeros que ya hayan superado los 5000 en gastos mediante una sentencia UPDATE y, además, cree los triggers necesarios para otorgar dicha condición a futuros pasajeros.

Apartado 3

JDBC

Se deberá completar el desarrollo de tres métodos que se enunciarán a continuación, haciendo uso de MySQL Connector/J 8.0. Los métodos que se deben implementar tienen que comunicarse con la base de datos, que ya debería estar creada tras completar el Apartado 2:

1. Añadir los exo-planetas de la Tabla 1 a la base de datos. Para ello complete previamente el método:

`nuevoPlaneta(String nombre, double masa, int radio, String sistema).`

Cuadro 1: Exoplanetas y sus características

Nombre del Planeta	Masa (kg)	Radio (km)	Sistema
Kepler-186f	$3,3 \times 10^{24}$	8.800	Copernico
HD 209458 b (Osiris)	$1,4 \times 10^{27}$	100.000	Beta Pictoris
LHS 1140 b	$8,3 \times 10^{24}$	8.800	Copernico

2. Mostrar por pantalla la lista de pasajeros (nombre) del planeta **Tierra** en criosueño de la cabina **A-60-S** (nombre y edad). Para ello complete previamente el método:

`listaPasajerosCabina (String cubierta, int cabina, String lado).`

3. Mostrar por pantalla una lista con los sistemas, los planetas que tiene cada uno, y el número de pasajeros que provienen de cada planeta. Para ello complete previamente el método: `listaOrigenes()`.

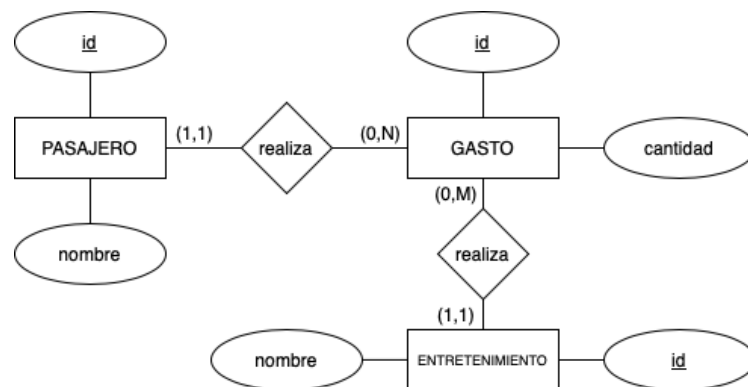
Para llevar a cabo la implementación de los métodos anteriores, se deberá completar las siguientes tareas:

1. Descargar la plantilla de código fuente para completar este sub-apartado y cargarlo en un proyecto Maven de IntelliJ. El código fuente se encuentra en GitHub: <https://github.com/bbddetsisi/practica-bd2425-jdbc>
2. Inicialice el servidor de MySQL.
3. Complete las funciones proporcionadas en la clase **Main**. Para ello analice el código del proyecto y complételo siguiendo las instrucciones de los comentarios **@TODO**.

Hibernate

Para completar este sub-apartado se deberá completar las siguientes tareas:

1. Descargar la plantilla de código fuente para completar este sub-apartado y cargarlo en un proyecto Maven de IntelliJ. El código fuente se encuentra en GitHub: <https://github.com/bbddetsisi/practica-bd2425-hibernate>.
2. Analice el fichero **hibernate.cfg.xml** para determinar el nombre del *schema* en el que hibernate creará la base de datos.
3. Realice las anotaciones en las clases pertinentes para que respeten las restricciones del siguiente Modelo Entidad-Relación². Únicamente será necesario incluir el código de las anotaciones, el resto de código de las clases no necesita ser modificado.



4. Complete las instrucciones **@TODO** del método **main** de la clase **Main**. Tal y como se indica deberá:
 - a) Crear un nuevo **Pasajero** llamado “Din Djarin” y un nuevo **Entretenimiento** llamado “Bounty Hunting”, y asignarle un **Gasto** en dicho entretenimiento de 100. Guardarlo todo en la base de datos.
 - b) Leer el fichero CSV **gastos.csv** que se encuentra en el directorio **resources** y recorrerlo usando **CSVParser** para crear los gastos que en él se encuentran. Dichos gastos deberán ser asignados al pasajero/a y al entretenimiento correspondientes. Se deben guardar todos estos datos en la base de datos.

²Como puedes observar, este modelo es una modificación del Modelo Entidad-Relación obtenido a partir del modelo relacional del apartado 2.2.b. Esta modificación consiste en añadir un nuevo atributo **id** a la entidad **Gastos**, con el propósito de evitar el uso de una clave primaria compuesta. Sin esta modificación, las anotaciones necesarias en las clases de Java incluirían elementos que no se han estudiado en clase y que quedan fuera del alcance de la asignatura. Asimismo, se han eliminado atributos de **Pasajero** para reducir el número de anotaciones a llevar a cabo.