****

Módulo Profesional 02:  
Desarrollo de aplicaciones web **Actividad UF4**

**CICLO FORMATIVO DE GRADO SUPERIOR EN**

**DESARROLLO DE APLICACIONES WEB**

**MODALIDAD ONLINE**

**JUAN BELLO FERNANDEZ**

**Actividad UF4:**

|  |  |
| --- | --- |
| **Objetivos** | |
| Interpretar el diseño lógico de bases de datos, analizando y cumpliendo las especificaciones relativas a su aplicación, para gestionar bases de datos.  Seleccionar lenguajes, objetos y herramientas, interpretando las especificaciones para  desarrollar aplicaciones web con acceso a bases de datos.  **Competencias asociadas:**   * Gestionar bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos. | |
| **Metodología** | **Entrega** |
| * Preparación individual * Discusión de la actividad | 10/05/2022 y formato de entrega en documento PDF. |
| **Dedicación estimada** | **Documentos de referencia** |
| 12 horas | Plataforma online. |
| **Resultados de aprendizaje** | |
| * Gestiona la información almacenada en bases de datos objeto-relacionales, evaluando y utilizando las posibilidades que proporciona el sistema gestor. | |
| **Criterios de evaluación** | |
| * Identificar las características de las bases de datos objeto-relacionales. * Creación de tipos de datos objeto, sus atributos y métodos. * Creación de tablas de objetos y tablas de columnas tipo objeto. * Creación de tipos de datos colección. * Realización de consultas. | |

**Desarrollo de la actividad**

Contenido

[1 Preguntas de desarrollo 3](#_Toc101430667)

[1.1 Detractores SGBDOR 3](#_Toc101430668)

[1.2 Elabora un cuadro de comparación de SQL con MongoDB. 4](#_Toc101430669)

[1.3 Busca información sobre las Geodatabase. ¿Cuál es su arquitectura y cómo justificamos que sea una base de datos orientada a objetos? 4](#_Toc101430670)

[2 Instalación Oracle 21c Express Edition y SQL Developer 5](#_Toc101430671)

[3 Base de datos Oracle 5](#_Toc101430672)

[3.1 Creación de la base de datos, usuario y privilegios 6](#_Toc101430673)

[3.2 Creación atributos y métodos superclase, subclases, tablas de cada tipo, inserción registros y consultas. 8](#_Toc101430674)

# Preguntas de desarrollo

## Detractores SGBDOR

Los detractores de los Sistemas Gestores de Bases de Datos Orientado a Objetos nos presentan una serie de desventajas, entre ellas:

* Carencia de un modelo de datos universal
* Carencia de experiencia
* Carencia de estándares
* Competencia, con respecto a los SGBDR y los SGBDOR
* La optimización de consultas compromete la encapsulación
* El modelo de objetos aún no tiene una teoría matemática coherente que le sirva de base

Escoge dos de ellos y explícalos en detalle.

|  |
| --- |
| **Carencia de estándares** |
| Una de las desventajas mencionadas por los detractores de las SGBDOR es la falta de estándares que permitan realizar los proyectos de una manera mucho más homogénea, más estandarizada lo que lleva a la resolución del mismo problema de diferentes maneras, pudiendo crear problemas de incompatibilidad en futuras versiones o migraciones de datos, entre otros.  El conocimiento profundo de la implementación y utilización de objetos se hace más necesario que nunca, ya que estos objetos pueden crear a su vez otros objetos con sus diferentes métodos lo que su utilización puede llegar a ser compleja, lo que sumado a la falta de estándares puede dar verdaderos dolores de cabeza sobre todo en supervisiones y revisiones posteriores. |
| **Carencia de experiencia** |
| Por otra parte, los detractores de las SGBDOR, mencionan la carencia de experiencia en este tipo de sistemas gestores. Su período de utilización y de implementación, unido a la falta de estándares comentado en el punto siguiente, no nos permite evaluar su efectividad y sostenibilidad en el medio/largo plazo.  Esto se hace evidente en la poca documentación que, actualmente, podemos encontrar en cualquier búsqueda en la web. Si extrapolamos esta situación a una empresa cualquiera, conllevaría gastos extra en formación de personal, así como migraciones de datos que podrían verse comprometidos, así como un aumento considerable de tiempos de gestión e implementación. |

## Elabora un cuadro de comparación de SQL con MongoDB.

|  |  |  |  |
| --- | --- | --- | --- |
| **CARACTERISTICA** | | **SQL** | **MongoDB** |
| **Aislamiento** | | **Si** | **No** |
| **Análisis de performance** | | **No** | **Si** |
| **Atomicidad** | | **Si** | **Condicional** |
| **CAP2** | | **CA** | **CP** |
| **Cloud, SaaS1, Web** | | **SI** | **SI** |
| **Conversión DB** | | **No** | **Si** |
| **Desarrolladores** | | **Oracle Corporation** | **MongoDB Inc.** |
| **Escalabilidad horizontal** | | **Condicional** | **Si** |
| **Integridad referencial** | | **Si** | **No** |
| **Lenguaje query** | | **SQL** | **Javascript** |
| **Mapa reducido** | | **No** | **Si** |
| **Modelo de integridad** | | **ACID** | **BASE** |
| **Modo de replicación** | | **Maestro-Maestro/Esclavo** | **Maestro-Esclavo** |
| **Sistema Operativo** | | **Multiplataforma** | **Multiplataforma** |
| **Transacciones** | | **Si** | **Si (a partir de Mongo 4.0)** |
| **Virtualización** | | **No** | **Si** |
|  |  | | |
| **1.SaaS:** El software como servicio (SaaS) permite a los usuarios conectarse a aplicaciones basadas en la nube a través de Internet y usarlas.  Fuente: <https://azure.microsoft.com/es-es/overview/what-is-saas/> | | | |
| 2.**CAP**:  el teorema CAP, también llamado Conjetura de Brewer, enuncia que es imposible para un [sistema de cómputo distribuido](https://es.wikipedia.org/wiki/Computaci%C3%B3n_distribuida) garantizar simultáneamente:​   * La [*consistencia*](https://es.wikipedia.org/wiki/Consistencia_de_datos)*(Consistency)*, es decir, cualquier lectura recibe como respuesta la escritura más reciente o un error. * La [*disponibilidad*](https://es.wikipedia.org/wiki/Disponibilidad)*(Availability)*, es decir, cualquier petición recibe una respuesta no errónea, pero sin la garantía de que contenga la escritura más reciente. * La [*tolerancia al particionado*](https://es.wikipedia.org/w/index.php?title=Tolerancia_al_particionado&action=edit&redlink=1)*(Partition Tolerance)*, es decir, el sistema sigue funcionando incluso si un número arbitrario de mensajes son descartados (o retrasados) entre nodos de la red.   Fuente: <https://es.wikipedia.org/wiki/Teorema_CAP> | | | |

## Busca información sobre las Geodatabase. ¿Cuál es su arquitectura y cómo justificamos que sea una base de datos orientada a objetos?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Arquitectura de una geodatabase** | | El modelo de almacenamiento de la geodatabase está basado en una serie simple, aunque esencial de conceptos de bases de datos relacionales y aprovecha los puntos fuertes del sistema de administración de base de datos (DBMS\*) subyacente.  Las tablas simples y los tipos de atributos bien definidos se utilizan para almacenar los datos de esquema, regla, base y atributos espaciales de cada dataset geográfico. Este enfoque proporciona un modelo formal para el almacenamiento y trabajo con los datos. A través de este enfoque, el lenguaje estructurado de consultas (SQL), se puede utilizar para crear, modificar y consultar tablas y sus elementos de datos. | | **La geodatabase como objeto relacional** | | La geodatabase se implementa utilizando la misma arquitectura de aplicación de varios niveles que se encuentra en otras aplicaciones DBMS avanzadas. La arquitectura de varios niveles de la geodatabase se conoce a veces como **modelo relacional de objetos**. Los objetos de geodatabase se almacenan como filas en tablas DBMS que tienen identidad, y el comportamiento se proporciona a través de la lógica de aplicación de la geodatabase. La separación de la lógica de la aplicación del almacenamiento es lo que permite admitir diversos DBMS y formatos de datos distintos. | | \**Un* ***DBMS*** *(DataBase Management System, por sus siglas en inglés) o sistema gestión de bases de datos (SGBD) o es una colección de software muy específico, orientado al manejo de base de datos, cuya función es servir de interfaz entre la base de datos, el usuario y las distintas aplicaciones utilizadas*. | |

# Instalación Oracle 21c Express Edition y SQL Developer

Siguiendo el manual de instalación aportado en la plataforma instalar Oracle 21c Express Edition y SQL Developer

|  |
| --- |
| Capturas de pantalla Oracle 21c Express Edition |
| E:\Documentos\ShareX\Screenshots\2022-04\VirtualBoxVM_TyhfuaNolj.pngE:\Documentos\ShareX\Screenshots\2022-04\VirtualBoxVM_x5S5QCwP6w.pngE:\Documentos\ShareX\Screenshots\2022-04\QrkDQ4gdaJ.pngE:\Documentos\ShareX\Screenshots\2022-04\OrMajayVPm.pngE:\Documentos\ShareX\Screenshots\2022-04\H9MdUzclxl.pngE:\Documentos\ShareX\Screenshots\2022-04\wPOmFifrKg.pngE:\Documentos\ShareX\Screenshots\2022-04\lefI5Z2RhO.pngE:\Documentos\ShareX\Screenshots\2022-04\sWp5DIOj8V.png |
| E:\Documentos\ShareX\Screenshots\2022-04\0wr1iviDJ6.pngE:\Documentos\ShareX\Screenshots\2022-04\GLsTfMWcwR.pngE:\Documentos\ShareX\Screenshots\2022-04\itNe4IuLnq.pngE:\Documentos\ShareX\Screenshots\2022-04\UV0tK25h9b.pngE:\Documentos\ShareX\Screenshots\2022-04\9M5RFGuUpZ.pngE:\Documentos\ShareX\Screenshots\2022-04\IXM5lyWlw5.pngE:\Documentos\ShareX\Screenshots\2022-04\phwtTsvwX2.png |
| E:\Documentos\ShareX\Screenshots\2022-04\DTCeatbSae.pngE:\Documentos\ShareX\Screenshots\2022-04\FkQDQijr8n.png |
| **E:\Documentos\ShareX\Screenshots\2022-04\aWpIkXKwO5.png** |
|  |
| Capturas de pantalla SQL Developer |
| **E:\Documentos\ShareX\Screenshots\2022-04\eMIqh4pclm.pngE:\Documentos\ShareX\Screenshots\2022-04\M9NK1Fd3WW.pngE:\Documentos\ShareX\Screenshots\2022-04\QgmJlZA93g.pngE:\Documentos\ShareX\Screenshots\2022-04\fNZl8TR4Et.pngE:\Documentos\ShareX\Screenshots\2022-04\4GkfYBd7dK.pngE:\Documentos\ShareX\Screenshots\2022-04\DFRzLuKuz0.pngE:\Documentos\ShareX\Screenshots\2022-04\NAS7FGm9bk.pngE:\Documentos\ShareX\Screenshots\2022-04\RKV0tt5uiw.pngE:\Documentos\ShareX\Screenshots\2022-04\BszPhyueCO.pngE:\Documentos\ShareX\Screenshots\2022-04\n4ILWcQCic.png** |
| **En el primer intento de prueba me dio un error por no cambiar el rol** |
| **E:\Documentos\ShareX\Screenshots\2022-04\5BK7KXuyKH.png** |
| **Al modificar el rol a SYSDBA la prueba de conexión es satisfactoriaE:\Documentos\ShareX\Screenshots\2022-04\kcMZsZPdM3.pngE:\Documentos\ShareX\Screenshots\2022-04\ncOxMCfow0.png** |
|  |

# Base de datos Oracle

Se aconseja visualizar el video de SGBDOR ubicado en la plataforma, así como el material complementario para realizar este apartado.

Queremos crear una base de datos objeto-relacional llamada VEHÍCULOS en Oracle. Se pide seguir los siguientes pasos:

## Creación de la base de datos, usuario y privilegios

|  |
| --- |
| Crear un tablespace llamado VEHÍCULOS de un tamaño de 30 MB. |
|  |
|  |
| Crear un usuario asociado llamado CONDUCTORTunombre asociado al tablespace con cuota ilimitada. |
|  |
| Dar privilegios de DBA al tablespace VEHÍCULOS |
|  |

En este punto se describe el diseño de la BDOR. No tienes que hacer ninguna acción.

La estructura de la base de datos estará formada por una superclase llamada VEHÍCULOS que tendrá 3 subclases llamadas respectivamente COCHE, AVION, AUTOBUS

La superclase VEHICULO tiene los siguientes atributos y métodos:

|  |  |
| --- | --- |
| VEHICULO | |
| Atributos  Matricula  Propietario  Marca  Modelo  PotenciaCV  Nplazas  Velocidad  Fechamatriculación  Ruedas | Métodos  GetMatricula (mostrar matricula, propietario, marca y modelo)  GetModelo (mostrar modelo, marca, matricula)  GetPotenciaCV (mostrar sólo potencia)  Getnplazas (mostrar nº plazas) |

La subclase COCHE tiene los siguientes atributos y métodos:

|  |  |
| --- | --- |
| COCHE | |
| Atributos  Color  Descapotable | Método  GetModelo (mostrar modelo,marca, matricula,color y descapotable) |

La subclase AVION tiene los siguientes atributos y métodos:

|  |  |
| --- | --- |
| AVION | |
| Atributos  Nmotores  Maxaltitud | Método  GetModelo(mostrar modelo,marca, matrícula, nmotores y maxaltltud) |

La subclase AUTOBUS tiene los siguientes atributos y métodos:

|  |  |
| --- | --- |
| AUTOBUS | |
| Atributos  Baño  NTV | Metodo  GetModelo(mostrar modelo,marca, matrícula y baño) |

## Creación atributos y métodos superclase, subclases, tablas de cada tipo, inserción registros y consultas.

Poner las sentencias a ejecutar para la creación de los elementos solicitados, así como las capturas de pantalla en el SQL Developer

|  |
| --- |
| Definir los atributos y métodos de la superclase VEHÍCULO con un archivo de texto plano. Recordar que hay que crear el objeto y su body |
|  |
| /\*CREACION OBJETO VEHICULO\*/  CREATE OR REPLACE TYPE Vehiculo AS OBJECT(  matricula VARCHAR2(7),  propietario VARCHAR2(45),  marca VARCHAR2(20),  modelo VARCHAR2(20),  potenciaCV NUMBER,  nPlazas NUMBER,  velocidad NUMBER,  fechaMatriculacion DATE,  ruedas NUMBER,  MEMBER FUNCTION GetMatricula RETURN VARCHAR2,  MEMBER FUNCTION GetModelo RETURN VARCHAR2,  MEMBER FUNCTION GetpotenciaCV RETURN NUMBER,  MEMBER FUNCTION GetnPlazas RETURN NUMBER,  PRAGMA RESTRICT\_REFERENCES(GetMatricula,WNDS),  PRAGMA RESTRICT\_REFERENCES(GetModelo,WNDS),  PRAGMA RESTRICT\_REFERENCES(GetPotenciaCV,WNDS),  PRAGMA RESTRICT\_REFERENCES(GetnPlazas,WNDS)  ) NOT FINAL;  /  /\*CREACION BODY VEHICULO\*/  CREATE OR REPLACE TYPE BODY Vehiculo AS  /\*GetMatricula\*/  MEMBER FUNCTION GetMatricula RETURN VARCHAR2 AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Matricula: '||matricula);  DBMS\_OUTPUT.PUT\_LINE('Propietario: '||propietario);  DBMS\_OUTPUT.PUT\_LINE('Marca: '||marca);  DBMS\_OUTPUT.PUT\_LINE('Modelo: ' ||modelo);  END GetMatricula;  /\*GetModelo\*/  MEMBER FUNCTION GetModelo RETURN VARCHAR2 AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Modelo: ' ||modelo);  DBMS\_OUTPUT.PUT\_LINE('Marca: '|| marca);  DBMS\_OUTPUT.PUT\_LINE('Matricula: '||matricula);  END GetModelo;  /\*GetPotenciaCV\*/  MEMBER FUNCTION GetPotenciaCV RETURN NUMBER AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Potencia en CV: '||potenciaCV);  END GetPotenciaCV;  /\*GetnPlazas\*/  MEMBER FUNCTION GetnPlazas RETURN NUMBER AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Número de plazas: '||nPlazas);  END GetnPlazas;  END;  / |
| Ejecutar desde Oracle SQL Developer las líneas de código SQL para crear el objeto raíz y su body. Una vez se haya creado correctamente pasar al siguiente punto. Captura de pantalla en SQL Developer |
|  |
| Definir las subclases COCHE, AVION, AUTOBUS con su herencia (heredará los atributos y los métodos de la superclase VEHÍCULOS) añadiendo los atributos y los polimorfismos (modificación de los métodos heredados) con un archivo de texto plano. Recordar que hay que crear el objeto con su herencia y su body con las modificaciones de los métodos (polimorfismo). |
|  |
| **/\*OBJETO COCHE\*/**  CREATE OR REPLACE TYPE coche UNDER Vehiculo (  color VARCHAR2(10),  descapotable CHAR(1),  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2,  PRAGMA RESTRICT\_REFERENCES(GetModelo,WNDS));  **/\*BODY COCHE\*/**  CREATE OR REPLACE TYPE BODY coche AS  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2 AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Modelo: '||modelo);  DBMS\_OUTPUT.PUT\_LINE('Marca: '|| marca);  DBMS\_OUTPUT.PUT\_LINE('Matrícula: '||matricula);  DBMS\_OUTPUT.PUT\_LINE('Color: '||color);  DBMS\_OUTPUT.PUT\_LINE('Descapotable: '||descapotable);  END GetModelo;  END;  **================================================================**  **/\*OBJETO AVION\*/**  CREATE OR REPLACE TYPE avion UNDER Vehiculo (  nMotores NUMBER,  maxAltitud NUMBER,  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2,  PRAGMA RESTRICT\_REFERENCES(GetModelo,WNDS));  **/\*BODY AVION\*/**  CREATE OR REPLACE TYPE BODY avion AS  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2 AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Modelo: '||modelo);  DBMS\_OUTPUT.PUT\_LINE('Marca: '|| marca);  DBMS\_OUTPUT.PUT\_LINE('Matricula: '||matricula);  DBMS\_OUTPUT.PUT\_LINE('Número de motores: '||nMotores);  DBMS\_OUTPUT.PUT\_LINE('Altitud máxima: '||maxAltitud);  END GetModelo;  END;  **================================================================**  **/\*OBJETO AUTOBUS\*/**  CREATE OR REPLACE TYPE autobus UNDER Vehiculo (  bano CHAR(1),  nTV NUMBER ,  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2,  PRAGMA RESTRICT\_REFERENCES(GetModelo,WNDS));  **/\*BODY AUTOBUS\*/**  CREATE OR REPLACE TYPE BODY autobus AS  OVERRIDING MEMBER FUNCTION GetModelo RETURN VARCHAR2 AS  BEGIN  DBMS\_OUTPUT.PUT\_LINE('Modelo: '||modelo);  DBMS\_OUTPUT.PUT\_LINE('Marca: '|| marca);  DBMS\_OUTPUT.PUT\_LINE('Matricula: '||matricula);  DBMS\_OUTPUT.PUT\_LINE('Número de baños: '||bano);  END GetModelo;  END; |
| Ejecutar desde Oracle SQL Developer las líneas de código SQL para crear los objetos heredados de la raíz con sus body. Una vez se haya creado correctamente pasar al siguiente punto. Captura de pantalla en SQL Developer |
| **SUBCLASE COCHE**      **SUBCLASE AVION**      **SUBCLASE AUTOBUS** |
| Crear las tablas de cada tipo creado (VEHÍCULO, COCHE, AUTOBUS y AVION con un archivo de texto plano. Una vez se haya creado correctamente pasar al siguiente punto. Captura de pantalla en SQL Developer |
| **/\*Vehiculo\*/**  CREATE TABLE tabla\_vehiculos OF vehiculo;  /  **/\*Coche\*/**  CREATE TABLE tabla\_coche OF coche;  /  **/\*Avion\*/**  CREATE TABLE tabla\_avion OF avion;  /  **/\*Autobus\*/**  CREATE TABLE tabla\_autobus OF autobus;  / |
| Insertar 4 registros de cada una de las subclases creadas de coches, aviones y autobuses en la base de datos objeto-relacional. Una vez se hayan insertado correctamente pasar al siguiente punto. Captura de pantalla en SQL Developer |
| **TABLA COCHE**      **TABLA AVION**      **TABLA AUTOBUS** |
| Realizar consultas de las 4 tablas creadas desde consola ó SQL Developer para verificar la correcta inserción de datos. Captura de pantalla |
| **TABLA COCHE**      **TABLA AVION**      **TABLA AUTOBUS** |

|  |
| --- |
| **Insertar ficheros de texto apartados 2.1 y 2.2** |
|  |

|  |  |
| --- | --- |
| **Ponderación actividad UF4 sobre 10** | **NOTA sobre 10** |
| 1. Preguntas de desarrollo    1. Detractores SGBDOR…………………….……….2 puntos    2. Cuadro comparación………………………………1 punto    3. Geodatabase………………………………………….1 punto | 4 |
| 1. Instalación Oracle 21c Express Edition y SQL Developer | 1 |
| 1. Base de datos Oracle    1. Creación de la base de datos, usuario y privilegios…………………………………………. ….1 punto    2. Creación atributos y métodos superclase, subclases, tablas de cada tipo, inserción registros y consultas ………………………………………………………….…..4 puntos | 5 |