Dirección Xeral de Educación, Formación Profesional e Innovación Educativa

Material para

a formación profesional inicial

A02. Consultas con datos de máis dunha táboa

|  |  |  |
| --- | --- | --- |
| Familia profesional | IFC | Informática e comunicacións |
| Ciclo formativo | CSIFC03  CSIFC02 | Desenvolvemento de aplicacións web  Desenvolvemento de aplicacións multiplataforma |
| Grao |  | Superior |
| Módulo profesional | MP0484 | Bases de datos |
| Unidade didáctica | UD05 | Consultas de datos |
| **Actividade** | **A02** | **Consultas con datos de máis dunha táboa** |
| Autores |  | Marta Fernández García  María del Carmen Fernández Lameiro  Miguel Fraga Vila  María Carmen Pato González  Andrés del Río Rodríguez |
| Nome do arquivo |  | CSIFC02\_ MP0484\_V000502\_UD05\_A02\_ConsultasMultitaboa.docx |
| © 2015 Xunta de Galicia.  Consellería de Cultura, Educación e Ordenación Universitaria.  Este traballo foi realizado durante unha licenza de formación retribuída pola Consellería de Cultura, Educación e Ordenación Universitaria e ten licenza CreativeCommons BY-NC-SA (recoñecemento - non comercial - compartir igual). Para ver unha copia desta licenza, visitar a ligazón http://creativecommons.org/licenses/by-nc-sa/3.0/es/. | | |

Índice

[1. Ficha técnica 5](#_Toc439854363)

[Contexto da actividade 5](#_Toc439854364)

[Título da actividade 6](#_Toc439854365)

[Resultados de aprendizaxe do currículo 6](#_Toc439854366)

[Obxectivos didácticos e título e descrición da actividade 6](#_Toc439854367)

[Criterios de avaliación 6](#_Toc439854368)

[Contidos 6](#_Toc439854369)

[Actividades de ensino e aprendizaxe e de avaliación, métodos, recursos e instrumentos de avaliación 7](#_Toc439854370)

[2. A02. Consultas con datos de máis dunha táboa 8](#_Toc439854371)

[2.1 Introdución 8](#_Toc439854372)

[2.1.1 Obxectivos 8](#_Toc439854373)

[2.1.2 Software 8](#_Toc439854374)

[2.1.3 Bases de datos de traballo 9](#_Toc439854375)

[2.1.3.1 Base de datos tendaBD 9](#_Toc439854376)

[2.1.3.2 Base de datos practicas5 11](#_Toc439854377)

[2.1.3.3 Base de datos traballadores 12](#_Toc439854378)

[2.2 Actividade 13](#_Toc439854379)

[2.2.1 Consultas con datos de máis dunha táboa 13](#_Toc439854380)

[2.2.2 Composición de táboas 13](#_Toc439854381)

[Nomes de columna cualificados 14](#_Toc439854382)

[2.2.2.1 Composición interna 14](#_Toc439854383)

[Composición interna con INNER JOIN e condición de igualdade 15](#_Toc439854384)

[Composición interna con INNER JOIN sen condición de igualdade 16](#_Toc439854385)

[2.2.2.2 Composición externa con OUTER JOIN 16](#_Toc439854386)

[2.2.2.3 Composición con NATURAL JOIN 17](#_Toc439854387)

[2.2.2.4 Composición dunha táboa con ela mesma. Autocomposición 18](#_Toc439854388)

[Normas para a realización de composicións internas e externas 19](#_Toc439854389)

[2.2.3 Unión de consultas (UNION) 19](#_Toc439854390)

[2.3 Tarefas 20](#_Toc439854391)

[2.3.1 Tarefa 1. Realizar consultas con datos de máis dunha táboa utilizando unha composición interna 21](#_Toc439854392)

[Solución 22](#_Toc439854393)

[2.3.2 Tarefa 2. Realizar consultas con datos de máis dunha táboa utilizando unha composición externa 24](#_Toc439854394)

[Solución 24](#_Toc439854395)

[2.3.3 Tarefa 3. Realizar consultas dunha táboa consigo mesma 25](#_Toc439854396)

[Solución 25](#_Toc439854397)

[2.3.4 Tarefa 4. Combinar o resultado de varias sentenzas SELECT para obter un conxunto de resultados único utilizando UNION 26](#_Toc439854398)

[Solución 26](#_Toc439854399)

[3. Materiais 28](#_Toc439854400)

[3.1 Documentos de apoio ou referencia 28](#_Toc439854401)

[3.2 Recursos didácticos 28](#_Toc439854402)

[3.3 Material auxiliar 28](#_Toc439854403)

[4. Avaliación 29](#_Toc439854404)

[Criterios de avaliación 29](#_Toc439854405)

[Modelo de proba combinada para TO.2, TO.3 e TO.5 29](#_Toc439854406)

[Proba para CA5.3 sobre consulta multitáboa con composición interna 30](#_Toc439854407)

[Proba para CA5.4 sobre consulta multitáboa con composición externa 30](#_Toc439854408)

[Proba para CA5.6 sobre consulta multitáboa con unión 31](#_Toc439854409)

[Exemplo de solución para entregar en papel 31](#_Toc439854410)

[Exemplo de lista de valoración para TO.2 33](#_Toc439854411)

[Exemplo de lista de valoración para TO.3 34](#_Toc439854412)

[Exemplo de lista de valoración para TO.5 34](#_Toc439854413)

1. Ficha técnica

Contexto da actividade

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Módulo | Duración  horas | Unidade didáctica. | Sesións 50´ | Actividades | Sesións 50´ |
| MP0484. Bases de datos | 187 | UD01. Bases de datos e sistemas de almacenamento da información | 12 | A01. Bases de datos e sistemas de almacenamento da información | 7 |
| A02. Introdución aos sistemas xestores de bases de datos | 5 |
| UD02. Deseño conceptual de bases de datos | 38 | A01. Fases do deseño de bases de datos | 3 |
| A02.Descrición e representación gráfica dos elementos do Modelo Entidade-Interrelación (MER) | 10 |
| A03. Descrición e representación gráfica dos elementos do Modelo Entidade-Interrelación Estendido(MERE) | 9 |
| A04. Construción e validación do modelo de datos | 11 |
| A05. Notación Martin e ferramentas CASE | 5 |
| UD03. Deseño lóxico de base de datos | 32 | A01. O deseño lóxico empregando o modelo relacional | 15 |
| A02. Normalización de relacións | 10 |
| A03. Operacións do MR: álxebra e cálculo | 7 |
| UD04. Deseño físico de bases de datos | 28 | A01. Creación da estrutura de bases de datos relacionais | 13 |
| A02. Modificación da estrutura de bases de datos relacionais | 9 |
| A03. Verificación e probas | 3 |
| A04. Utilización de ferramentas gráficas para a creación da estrutura de bases de datos relacionais | 3 |
| **UD05. Consultas de datos** | **39** | A01. Consultas simples cunha táboa. | 12 |
| **A02. Consultas con datos de máis dunha táboa** | **10** |
| A03. Consultas resumo con agrupamentos | 7 |
| A04. Consultas con subconsultas | 7 |
| A05. Optimización de consultas | 3 |
| UD06. Tratamento de datos | 24 | A01. Manipulación de datos | 12 |
| A02. Vistas | 5 |
| A03. Transaccións e bloqueos | 7 |
| UD07. Programación de bases de datos | 22 | A01. Introdución á programación con SQL | 4 |
| A02. Rutinas almacenadas | 9 |
| A03. Disparadores e eventos | 5 |
| A04. Cursores | 4 |
| UD08. Administración de bases de datos | 19 | A01. Copias de seguridade e intercambio de datos entre SXBDs | 10 |
| A02. Índices e xestión de usuarios | 9 |
| UD09. Bases de datos obxecto-relacionais | 10 | A01. Uso de bases de datos obxecto-relacionais | 10 |

NOTA: Esta actividade está vinculada á programación recollida no arquivo CSIFC02\_MP0487\_V000500\_UD05\_Consultas.pdf

Título da actividade

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Título | Descrición | Duración |
| A02 | Consultas con datos de máis dunha táboa | Realizáronse consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | 10 |

Resultados de aprendizaxe do currículo

|  |  |
| --- | --- |
| Resultados de aprendizaxe do currículo | Completo |
| * RA5 - Consulta a información almacenada nunha base de datos empregando asistentes, ferramentas gráficas e a linguaxe de manipulación de datos | Non |

Obxectivos didácticos e título e descrición da actividade

| Obxectivos específicos | | Actividade | | Descrición básica | Duración |
| --- | --- | --- | --- | --- | --- |
| O2.1 | Realizar consultas con máis dunha táboa utilizando composicións internas e externas. | A02 | Consultas con datos de máis dunha táboa. | Realizáronse consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | 10 |
| O2.2 | Realizar consultas con máis dunha táboa utilizando unións. |

Criterios de avaliación

|  |
| --- |
| Criterios de avaliación |
| * CA5.3 Realizáronse consultas sobre o contido de varias táboas mediante composicións internas. * CA5.4 Realizáronse consultas sobre o contido de varias táboas mediante composicións externas. * CA5.6 Realizáronse unións de consultas. |

Contidos

| Contidos |
| --- |
| * Composicións internas e externas. * Unión de consultas. |

Actividades de ensino e aprendizaxe e de avaliación, métodos, recursos e instrumentos de avaliación

| Que e para que | Como | | | Con que | Como e con que se valora | Duración (sesións) |
| --- | --- | --- | --- | --- | --- | --- |
| Actividade (título e descrición) | Profesorado (en termos de tarefas) | Alumnado (tarefas) | Resultados  ou produtos | Recursos | Instrumentos e procedementos de avaliación |
| A02. Consultas con datos de máis dunha táboa   * Realizáronse consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | * **Tp2.1** Exposición de conceptos sobre sentenzas LMD (Linguaxe de manipulación de datos) que permiten realizar consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | **Ta2.1** Toma de notas e formulación de dúbidas. |  | * Material didáctico subministrado polo profesorado en papel e/ou formato dixital. * Proxector. * Ordenadores con conexión a Internet, que terán instalado MySQL e MySQL Workbench. * Manual de referencia de MySQL. |  | 1 |
| * **Tp2.2** Demostración práctica de consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | * **Ta2.2** Realización de consultas con datos de máis dunha táboa, utilizando composicións de táboas e unións de consultas. | * Guións de sentenzas LMD que solucionan os exercicios de consulta propostos. |  | 8 |
|  | * **Ta2.3** Tarefa de avaliación combinando os instrumentos de avaliación TO.2, TO.3 e TO.5. | * Exame en papel e en formato dixital. | * Máquina virtual para exame que terá instalado MySQL e MySQL Workbench. * Manual de referencia de MySQL. | * **TO.2** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións internas. Escala de valores (observación indirecta). * **TO.3** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións externas. Escala de valores (observación indirecta). * **TO.5** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando unións. Escala de valores (observación indirecta). | 1 |

1. A02. Consultas con datos de máis dunha táboa
   1. Introdución
      1. Obxectivos

Os obxectivos desta actividade son:

* Realizar consultas con datos de máis dunha táboa utilizando a composición de táboas e a unión de consultas.
  + 1. Software

Utilizarase a plataforma WAMP (Windows-Apache-MySQL-PHP) WampServer 2.5 (última versión estable en outubro 2015), que inclúe MySQL Community Edition 5.6.17 como SXBDR (Sistema Xestor de Bases de Datos Relacional). As razóns de utilización deste software son que:

* É software libre, polo que o alumnado poderá descargalo de forma gratuíta e utilizalo legalmente na súa casa.
* É unha forma sinxela de facer a instalación do software necesario para desenvolver aplicacións web.
* Páxina oficial de  WampServer: <http://www.wampserver.com>
* Páxina oficial de MySQL: <https://www.mysql.com/>

Utilizarase MySQL Workbench 6.3 como ferramenta cliente gráfica xa que é a recomendada por MySQL en outubro de 2015, aínda que tamén poderían utilizarse outras como phpMyAdmin, EMS MyManager, ou MySQL Query Browser.

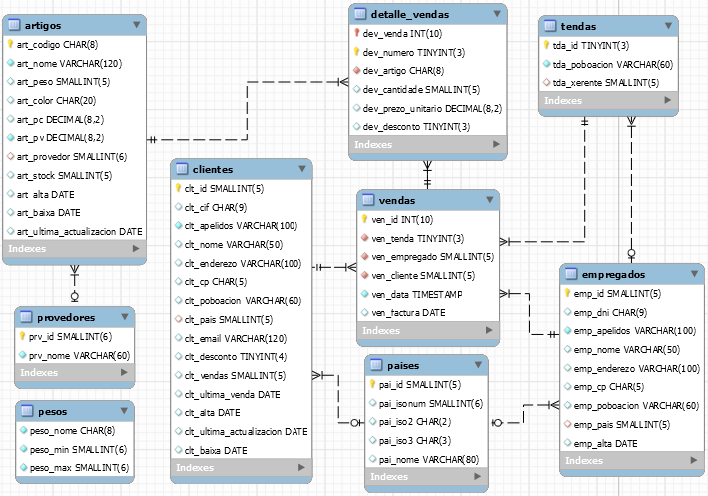
Normalmente, para a proba das consultas realizadas nesta actividade, mostrarase a zona de manipulación de datos de Workbench, coas filas que forman a táboa de resultados. Para completar a anterior información ou cando a consulta non pode mostrarse enteira xa que devolve moitas filas, mostrarase ademais a zona de saída (output) coa información do estado da execución da consulta e o número de filas que devolve.

* En [https://www.mysql.com/products/workbench/](https://www.mysql.com/products/workbench/%20) pode obterse información detallada sobre a ferramenta MySQL Workbench e descargar o software.
* En [http://dev.mysql.com/doc/index-gui.html](http://dev.mysql.com/doc/index-gui.html%20) pode descargarse o manual de MySQL Workbench.
* O material anexo a esta actividade inclúe unha guía básica de MySQL Workbench 6.3.
  + 1. Bases de datos de traballo

As bases de datos *tendaBD*, *traballadores* e *practicas5* utilizaranse para os exemplos e tarefas desta actividade. Antes de empezar a probar os exemplos ou realizar as tarefas, hai que executar os scripts de creación no servidor e poñer en uso a base de datos correspondente. Os scripts atópanse no cartafol anexo a esta actividade descrito no apartado '3.3 Material auxiliar'.

* + - 1. Base de datos tendaBD

A base de datos *tendaBD* serve para controlar as vendas dunha cadea de tendas. Gárdanse nela os datos das vendas que se realizan, das tendas nas que se fan as vendas, dos artigos vendidos, e dos clientes. As táboas desta base de datos que se van a utilizar nesta actividade móstranse no seguinte diagrama entidade relación deseñado con Workbench e descríbense a continuación.



* Táboa *empregados*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| emp\_id | smallint unsigned | Non | Primaria | Identificador do empregado. Numéranse de 1 en adiante de forma automática. |
| emp\_dni | char(9) |  |  | DNI do empregado. |
| emp\_apelidos | varchar(100) | Non | Índice | Apelidos do empregado. |
| emp\_nome | varchar(50) |  | Nome do empregado. |
| emp\_enderezo | varchar(100) |  |  | Enderezo do empregado. |
| emp\_cp | char(5) |  |  | Código postal do empregado. |
| emp\_poboacion | varchar(60) |  |  | Poboación do empregado. |
| emp\_pais | smallint unsigned |  | Foránea | Código do país segundo a táboa de países. |
| emp\_alta | date |  |  | Data na que se deu de alta o empregado. |

* Táboa *pesos*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| peso\_nome | char(8) | Non |  | Nome que describe o tipo de peso. |
| peso\_min | smallint | Non |  | Peso mínimo para ese nome. |
| peso\_max | smallint | Non |  | Peso máximo para ese nome. |

* Táboa *clientes*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| clt\_id | smallint unsigned | Non | Primaria | Identificador do cliente. Numeraranse de 1 en adiante de forma automática. |
| clt\_cif | char(9) |  | Única |  |
| clt\_apelidos | varchar(100) | Non | Índice | Apelidos ou razón social do cliente. |
| clt\_nome | varchar(50) |  | Nome ou tipo de sociedade (SL, SA, ...) do cliente. |
| clt\_enderezo | varchar(100) |  |  |  |
| clt\_cp | char(5) |  |  | Código postal do cliente. |
| clt\_poboacion | varchar(60) |  |  |  |
| clt\_pais | smallint unsigned |  | Foránea | Código do país segundo a táboa de países. |
| clt\_email | varchar(120) |  |  |  |
| clt\_desconto | tinyint |  |  | Porcentaxe de desconto aplicable ao cliente. |
| clt\_vendas | smallint unsigned |  |  | Número de vendas feitas ao cliente. |
| clt\_ultima\_venda | date |  |  | Data da última venda feita ao cliente. |
| clt\_alta | date | Non |  | Data na que se deu de alta ao cliente. |
| clt\_ultima\_actualizacion | date |  |  | Data da última vez que se fixeron cambios nos datos do cliente. |
| clt\_baixa | date |  |  | Data na que se deu de baixa ao cliente. |

* Táboa *artigos*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| art\_codigo | char(8) | Non | Primaria | Toma valores entre 1 e 200.000. |
| art\_nome | varchar(120) | Non | Índice | Nome ou descrición do artigo. |
| art\_peso | smallint unsigned |  |  | Peso en gramos. Valor numérico enteiro. |
| art\_color | char(20) |  |  | Cor do artigo |
| art\_pc | decimal(8,2) |  |  | Prezo de compra do artigo. |
| art\_pv | decimal(8,2) | Non |  | Prezo de venda do artigo. |
| art\_provedor | smallint |  | Foránea | Identificador do provedor. |
| art\_stock | smallint unsigned |  |  | Número de unidades do artigo dispoñibles no almacén. |
| art\_alta | date | Non |  | Data na que se deu de alta o artigo. |
| art\_baixa | date |  |  | Data na que se deu de baixa o artigo. |
| art\_ultima\_actualizacion | date |  |  | Data da última vez que se fixeron cambios nos datos do artigo. |

* Táboa *paises*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| pai\_id | smallint unsigned | Non | Primaria | Identificador do país. Numeraranse de 1 en adiante de forma automática. |
| pai\_isonum | smallint |  |  | Número de país segundo a norma ISO 3166-1:2013.[[1]](#footnote-3) |
| pai\_iso2 | char(2) |  |  | Código de país de 2 carácteres segundo a norma ISO 3166-1:2013. |
| pai\_iso3 | char(3) |  |  | Código de país de 3 carácteres segundo a norma ISO 3166-1:2013. |
| pai\_nome | varchar(80) |  |  | Nome do país. |

* Táboa *provedores*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| prv\_id | smallint | Non | Primaria | Identificador do provedor. |
| prv\_nome | varchar(60) | Non |  | Nome do provedor. |

* Táboa *tendas*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| tda\_id | tinyint unsigned | Non | Primaria | Identificador da tenda. Numéranse do 1 en adiante de forma automática. |
| tda\_poboacion | varchar(60) | Non |  | Poboación na que está situada a tenda. |
| tda\_xerente | smallint unsigned |  | Foránea | Identificador do empregado que é xerente da tenda. |

* Táboa *vendas*

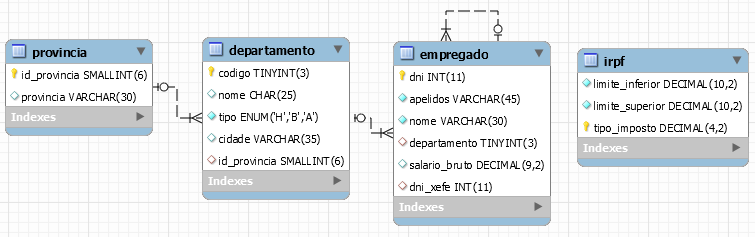
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| ven\_id | int unsigned | Non | Primaria | Identificador da venda. Numeraranse de 1 en adiante de forma automática. |
| ven\_tenda | tinyint unsigned | Non | Foránea | Identificador da tenda na que se fixo a venda. |
| ven\_empregado | smallint unsigned | Non | Foránea | Identificador do empregado que fixo a venda. |
| ven\_cliente | smallint unsigned | Non | Foránea | Identificador do cliente ao que se fixo a venda. |
| ven\_data | date | Non |  | Data e hora na que se fixo a venda. |
| ven\_factura | date |  |  | Data da factura na que se inclúe esta venda. |

* Táboa *detalle\_vendas*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | | Observacións |
| dev\_venda | int unsigned | Non | Primaria | Foránea | Identificador da venda á que corresponde a liña de detalle. |
| dev\_numero | tinyint unsigned | Non |  | Número da liña de detalle dentro da venda. |
| dev\_artigo | char(8) | Non | Foránea | | Identificador do artigo vendido. |
| dev\_cantidade | smallint unsigned | Non |  | | Número de unidades vendidas. |
| dev\_prezo\_unitario | decimal(8,2) unsigned | Non |  | | Prezo por cada unidade vendida. |
| dev\_desconto | tinyint unsigned | Non |  | | Porcentaxe de desconto aplicado. |

* + - 1. Base de datos practicas5

A base de datos *practicas5* está creada con fins didácticos para realizar os exemplos de consultas nesta unidade. Está formada por un grupo de táboas, relacionadas entre si, tal e como se mostra no seguinte diagrama entidade relación deseñado con Workbench e se describe a continuación.



* Táboa *empregado.*  A columna *departamento* é unha clave foránea que contén o código do departamento no que traballa o empregado, e fai referencia á columna *codigo* da táboa *departamento*. Os valores que toma a columna *departamento* teñen que coincidir cos que toma a columna *codigo* da táboa *departamento*, ou ser NULL no caso que o empregado non teña asignado ningún departamento. A columna *dni\_xefe* é outra clave foránea que contén o dni doutro empregado que sería o seu xefe, ou o valor NULL no caso que non tivera xefe.
* Táboa *departamento*. A columna *id\_provincia* é unha clave foránea que fai referencia á columna *id\_provincia* da táboa *provincia*.
* Táboa *irpf.* Contén a porcentaxe de imposto que hai que aplicarlle a cada empregado, en función do seu salario bruto, dependendo dos límites entre os que se atope. Esta táboa podería conter unha información similar a esta:

u5a1_taboairpf.emf

* + - 1. Base de datos traballadores

A base de datos *traballadores* serve para levar control dos empregados, departamentos e centros dunha empresa. Está formada por un grupo de táboas, relacionadas entre si, tal e como se mostra no seguinte grafo relacional e se describe a continuación. As táboas son MyIsam (non transaccionais) e por tanto non teñen definidas claves foráneas.

TRABALLADORES_GRAFO.emf

* Táboa centro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| cenNumero | int | Non | Primaria | Número co que se identifica. |
| cenNome | char(30) |  | Índice | Nome. |
| cenEnderezo | char(30) |  |  | Enderezo. |

* Táboa empregado

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| empNumero | int | Non | Primaria | Número co que se identifica. |
| empDepartamento | int | Non | Índice | Número do departamento no que traballa. |
| empExtension | smallint | Non |  | Extensión telefónica para o empregado. Pode compartirse entre empregados de diferentes departamentos. |
| empDataNacemento | date |  |  | Data de nacemento. |
| empDataIngreso | date |  |  | Data de ingreso na empresa. |
| empSalario | decimal(6,2) |  |  | Salario mensual en euros. |
| empComision | decimal(6,2) |  |  | Comisión mensual. |
| empFillos | smallint |  |  | Número de fillos. |
| empNome | char(20) | Non | Índice | Nome do empregado coa forma: primeiro apelido, nome. |

* Táboa departamento

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| depNumero | int | Non | Primaria | Número co que se identifica. |
| depNome | char(20) |  | Índice | Nome. |
| depDirector | int | Non | Índice | Número do empregado director do departamento. |
| deptipoDirector | char(1) |  |  | Tipo de directo: P (en propiedade, é dicir, titular), F (en funcións). |
| depPresuposto | decimal(9,2) |  |  | Cantidade en euros de presuposto anual. |
| depDepende | int |  | Índice | Número do departamento do que depende. |
| depCentro | int |  | Índice | Número do centro ao que pertence. |
| depEmpregados | smallint unsigned |  |  | Número de empregados que traballan no departamento. |

* 1. Actividade
     1. Consultas con datos de máis dunha táboa

Na primeira actividade desta unidade didáctica, realizáronse consultas simples con datos dunha única táboa pero a práctica diaria de consultas sobre unha base de datos implica frecuentemente a máis dunha táboa. Este tipo de consultas pódense realizar de varias formas en función do requirimento da consulta:

* Utilizar composicións de táboas nunha sentenza SELECT escribindo unha única consulta que utiliza a cláusula JOIN dentro da cláusula FROM.
* Utilizar sentenzas SELECT aniñadas dentro doutras sentenzas SELECT. As sentenzas aniñadas chámanse subconsultas e veranse máis adiante na actividade 4.
* Unindo o conxunto de resultados dunha consulta co conxunto de resultados doutras consultas, empregando o operador UNION.
  + 1. Composición de táboas

As primeiras normas ANSI SQL para combinar varias táboas nunha consulta permitían poñer a relación das táboas na cláusula FROM separadas por coma, e as condicións para facer os enlaces entre as táboas na cláusula WHERE. Esta sintaxe sigue estando permitida hoxe en día, aínda que non é a máis recomendable.

Exemplo: seleccionar apelidos, nome e cidade dos empregados que estean asignados a un departamento, tendo en conta que a cidade é a cidade na que está situado o departamento no que traballa o empregado.

/\* Enlace entre dúas táboas utilizando unha clave foránea\*/

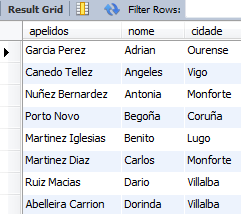
**select** apelidos**,** empregado**.**nome**,** cidade

**from** empregado**,** departamento

**where** departamento **=** codigo**;**

A primeira operación que se realiza cunha sentenza como a anterior na que se fai referencia a máis dunha táboa na cláusula FROM, é o produto cartesiano entre esas táboas, é dicir, relaciona cada fila dunha táboa con todas as filas da outra táboa. Por iso despois hai que poñer a condición de enlace na cláusula WHERE para que se seleccionen só aquelas composicións de filas que nos interesan.

O número de filas que devolve a consulta do exemplo é 19 tal e como se observa na imaxe seguinte, a pesar de que hai 20 empregados. Isto é debido a que hai un empregado que ten na columna *departamento* o valor NULL, por non estar asignado a ningún departamento, e non se pode combinar con ningunha fila da táboa *departamento* porque a columna *código* é clave primaria e non pode ter o valor NULL.





A sintaxe das composicións de táboas cambia a partir da norma ANSI92, para poñer as condicións de enlace na cláusula FROM xunto coa relación de táboas, coa finalidade de deixar a cláusula WHERE para as condicións que deben cumprir as filas que hai que mostrar.

Nesta actividade, utilizarase a sintaxe que recomenda a norma ANSI92 por ser máis eficiente e estruturar mellor o código das consultas.

|  |  |  |
| --- | --- | --- |
| Resumo de tipo de composición máis utilizados | | |
| Composición interna | INNER JOIN con condición de igualdade  INNER JOIN sen condición de igualdade |  |
| Composición externa | LEFT OUTER  RIGTH OUTER | Prioridade esquerda  Prioridade dereita |
| Composición natural | NATURAL JOIN | Pode ser interna ou externa |
| Autocomposición | Composición dunha táboa consigo mesma | Utilizando alias de táboas |

Nomes de columna cualificados

Nas combinacións de tipo interna, externa e autocomposición, hai que ter moi en conta que pode haber columnas que teñan o mesmo nome en dúas ou máis táboas. Cando sucede isto é obrigatorio cualificar os nomes das columnas, utilizando o formato *nome\_táboa.nome\_columna*, para evitar erros producidos polo uso de nomes ambiguos.

Tamén é recomendable utilizar os nomes de columna cualificados para mellorar o rendemento da consulta, porque proporcionan información a o servidor que desta maneira non ten que buscar a que táboa pertence cada columna. Para simplificar o uso de nomes cualificados e que estes sexan máis curtos, recoméndase utilizar alias para os nomes de táboa. Exemplo:

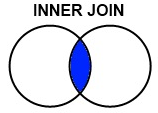
**select** em.apelidos**,** em**.**nome**,** de.cidade

**from** empregado **as** em**,** departamento **as** de

**where** em.departamento **=** de.codigo**;**

* + - 1. Composición interna

A composición interna crea unha táboa que contén todas as columnas das táboas que forman a composición e só as filas que cumpren as condicións da composición.



Composición interna con INNER JOIN e condición de igualdade

Este tipo de composición é a máis utilizada e permite relacionar dúas táboas que teñen algunha columna con datos comúns que serve de enlace entre elas. O caso máis normal é o que establecen as claves foráneas dunha táboa, que toman valores que teñen que coincidir cos valores da clave primaria doutra táboa. Esta composición dá como resultado o conxunto formado polas parellas de filas das dúas táboas que interveñen na composición, que cumpran a condición de que o valor da clave foránea dunha sexa igual ao valor da clave primaria da outra. As columnas de enlace deben ter asociados índices para optimizar o rendemento da consulta.

A sintaxe utilizada na norma ANSI92 para a reunión interna é:

FROM nome\_táboa\_1 [ INNER ] JOIN nome\_táboa\_2

ON ( condición de enlace )

A opción INNER indica o tipo de composición interna, pero non é necesario poñela explicitamente. Cando se pon só a palabra JOIN entre os nomes de dúas táboas enténdese que é unha composición interna (INNER).

Exemplo: seleccionar *apelidos*, *nome* e *cidade* dos empregados que estean asignados a un departamento, tendo en conta que *cidade* é a cidade na que está situado o departamento no que traballa o empregado. A condición de enlace escríbese na cláusula FROM, reservando a cláusula WHERE para establecer condicións que seleccionen as filas das táboas de orixe que nos interesan.

/\* Enlace entre dúas táboas utilizando unha clave foránea

con INNER JOIN segundo a norma ANSI-92\*/

**select** em.apelidos**,** em**.**nome**,** de.cidade

**from** empregado **as** em **inner** **join** departamento **as** de **on** **(em.**departamento **=** de.codigo**);**

Cando interveñen máis de dúas táboas na composición, hai que ter en conta que para cada JOIN ten que existir unha condición de enlace. Neste caso, pódese asociar unha cláusula ON a cada JOIN coa correspondente condición de enlace, ou ben poñer unha única cláusula ON cunha condición composta formada por todas as condicións de enlace relacionadas con operadores AND.

Exemplo: mostrar *apelidos* e *nome* do empregado, nome do departamento no que traballa e nome da provincia no que está o departamento.

**select** em.apelidos**,**em**.**nome**,**de**.**nome**,pr.**provincia

**from** empregado **as** em

**join** departamento **as** de **on** **(**em.departamento **=** de.codigo**)**

**join** provincia **as** pr **on** **(**de**.**id\_provincia **=** pr**.**id\_provincia**);**

Tamén se podería escribir utilizando unha única condición de enlace composta:

**select** em.apelidos**,**em**.**nome**,**de**.**nome**,pr.**provincia

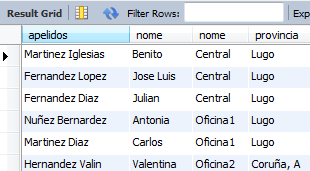
**from** empregado **as** em

**join** departamento **as** de

**join** provincia **as** pr

**on** **(em.**departamento **=** de.codigo **and** de**.**id\_provincia **=** pr**.**id\_provincia**);**

Calquera destas consultas obtén o seguinte resultado:





Composición interna con INNER JOIN sen condición de igualdade

Cando se fai unha composición entre dúas táboas utilizando unha composición interna, é posible utilizar condicións de enlace distintas da condición de igualdade; neste caso suponse que non existe ningunha columna que conteña valores idénticos nas dúas táboas.

Exemplo: seleccionar *apelidos*, *nome*, salario bruto, tipo de imposto e salario neto (salario bruto – impostos) de todos os empregados ordenados por nome.

/\* JOIN sen condición de igualdade \*/

**select** em.apelidos**,** em.nome**,** em.salario\_bruto**,** ir.tipo\_imposto**,**

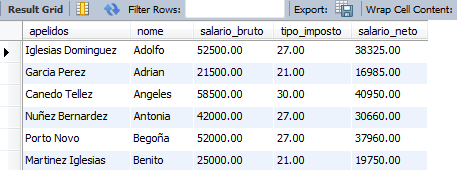
**round(**em.salario\_bruto**-**em**.**salario\_bruto**\***ir.tipo\_imposto**/**100**,**2**)** **as** salario\_neto

**from** empregado **as** em **join** irpf **as** ir

**on** em.salario\_bruto **between** ir.limite\_inferior **and** ir.limite\_superior

**order by** em.nome**;**

Neste caso, o *salario\_bruto* é a columna da táboa *empregado* que se vai a utilizar como enlace e non ten que coincidir exactamente co valor almacenado en ningunha columna da táboa *irpf*, senón que ten que estar entre un *límite\_inferior* e un *límite\_superior,* ambos inclusive*.* O resultado da execución en Workbench podería ser:





* Tarefa 1. Realizar consultas con datos de máis dunha táboa utilizando unha composición interna.
  + - 1. Composición externa con OUTER JOIN

A composición externa permite mostrar as filas dunha das táboas aínda que non cumpran a condición de enlace (toma o valor NULL para as columnas do resto de táboas), ademais das parellas de filas para as que se cumpra a condición de enlace tal e como ocorría na composición interna.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Existen dous tipos de reunión externa: pola esquerda, e pola dereita, dependendo de cal das táboas se considere táboa principal. A sintaxe é:

FROM *nome\_táboa\_1* { LEFT | RIGTH } [ OUTER ] JOIN *nome\_táboa\_2*

ON ( *condición de enlace* )

* A opción LEFT mostra todas as filas da táboa da esquerda, aínda que non estean relacionas con filas do resto das táboas.
* A opción RIGHT, mostra todas as filas da táboa da dereita, aínda que non estean relacionas con filas do resto das táboas.

Exemplo: mostrar *apelidos* e *nome* dos empregados e nome da cidade na que está o departamento no que traballa, aínda que o empregado non teña departamento asignado. O resultado ten que estar ordenado polo nome.

/\* OUTER JOIN enlace externo, neste caso, pola esquerda\*/

**select** em.apelidos**,** em**.**nome**,** de.cidade

**from** empregado **as** em **left** **join** departamento **as** de

**on** **(em.**departamento **=** de.codigo**)**

**order** **by** em.nome**;**

Na sentenza anterior, a táboa principal é a táboa *empregado* por estar situada á esquerda (*left*), e a táboa *departamento* considérase a táboa secundaria da consulta. Na execución, móstrase unha fila para cada empregado, e para os que non cumpran a condición de enlace por non ter asignado aínda un departamento, móstrase o valor NULL na columna *cidade*. O número de filas que devolve a consulta é 20 que é o número de filas que ten a táboa *empregado*.





Exemplo de LEFT JOIN con IF NULL: mostrar *apelidos* e *nome* dos empregados que non teñen departamento asignado.

**select** em**.**apelidos**,** em**.**nome

**from** empregado **as** em **left** **join** departamento **as** de

**on** **(**em**.**departamento **=** de**.**codigo**)**

**where** de**.**cidade **is** **null;** # Vale calquera columna da táboa departamento

Para as filas da táboa principal (*empregado*) que non cumpran a condición de enlace, todas as columnas da táboa secundaria (*departamento*) toman o valor NULL, polo que ao establecer a condición da cláusula WHERE só se mostran as filas da táboa principal que non están relacionadas con ningunha fila da táboa secundaria.



* Tarefa 2. Realizar consultas con datos de máis dunha táboa utilizando unha composición externa.
  + - 1. Composición con NATURAL JOIN

A composición natural permite enlazar táboas por columnas que teñen o mesmo nome. Neste tipo de composicións non é necesario introducir a condición de enlace coa opción ON, pois sobreenténdese que a condición de enlace consiste en comprobar a coincidencia dos valores das columnas que teñen o mesmo nome. Ademais, as columnas que existen nas dúas táboas só se mostran unha vez, polo que non é necesario cualificalas empregando o formato *nome\_táboa.nome columna.* Sintaxe:

FROM *nome\_táboa\_1* NATURAL [{ LEFT | RIGTH } [ OUTER ]] JOIN *nome\_táboa\_2*

A combinación con NATURAL JOIN pode dar prioridade a algunha das táboas, cando se define como enlace externo (OUTER).

Cando se utiliza NATURAL JOIN hai que ter coidado que nas dúas táboas só teñan o mesmo nome as columnas que se van a utilizar para facer o enlace, xa que en calquera outro caso os resultados obtidos poden ser imprevisibles.

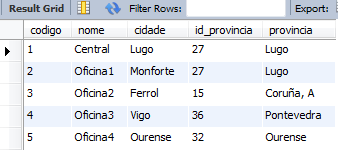
Exemplo: mostrar os datos de todas as columnas da táboa *departamento*, completados co nome da provincia da táboa *provincia*.

/\* Utilización de NATURAL JOIN \*/

**select** de**.\*,** pr.provincia

**from** departamento **as** de **natural** **join** provincia **as** pr

**order** **by** de.codigo**;**





Na mesma consulta pódense utilizar distintos tipos de composición.

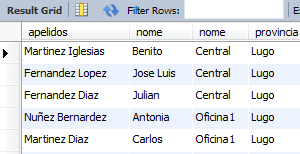
Exemplo: mostrar apelidos e nome do empregado, nome do departamento no que traballa e nome da provincia no que está o departamento.

/\* Utilización de dúas composicións, una delas NATURAL JOIN \*/

**select** em.apelidos**,**em**.**nome**,**de**.**nome**,pr.**provincia

**from** empregado as em **join** departamento as de **on** **(em.**departamento **=** de.codigo**)**

**natural join** provincia as pr**;**





* + - 1. Composición dunha táboa con ela mesma. Autocomposición

A autocomposición utilízase cando hai que comparar unha fila dunha táboa con outras filas da mesma táboa. Para evitar que ao ler unha fila da táboa se perda a información da anterior fila lida, pódese poñer dúas veces o nome da táboa na cláusula FROM, asignándolle dous alias ou sinónimos diferentes; a partir dese momento, manéxanse como se foran dúas táboas distintas. As consultas feitas cunha autocomposición poden ser resoltas tamén empregando subsentenzas, como se verá en actividades posteriores.

Exemplo: mostrar apelidos e nome dos empregados, e apelidos e nome do seu xefe.

/\* Autocomposición: Enlace dunha táboa con ela mesma \*/

**select** em1**.**apelidos**,** em1**.**nome**,** em1**.**departamento**,**

em2**.**apelidos **as** Apelido\_xefe**,** em2**.**nome **as** Nome\_xefe

**from** empregado **as** em1 **join** empregado **as** em2

**on** em1**.**dni\_xefe **=** em2**.**dni

**order** **by** Apelido\_xefe**,** Nome\_xefe**;**





* Tarefa 3. Realizar consultas dunha táboa consigo mesma.

Normas para a realización de composicións internas e externas

Non existen normas estándar para as composicións internas e externas, pero a continuación enuméranse unha serie de conclusións sobre as composicións internas e externas que axudarán a utilizalas mellor.

* As columnas que se utilizan para facer a composición deben ser do mesmo tipo. Para mellorar o rendemento da consulta deben ter asociados índices.
* Cando se fai unha composición, é como se crease unha táboa que ten as columnas de todas as táboas que se combinan, e as filas que verifican as condicións de composición. Por esta razón, na sentenza SELECT pódese utilizar calquera columna das táboas que se relacionan na cláusula FROM.
* Pódense combinar tantas táboas como desexemos, pero poñer táboas innecesarias reduce o rendemento da consulta.
* Unha mesma táboa pode aparecer máis dunha vez na composición, pero utilizando alias diferentes.
* Recoméndase utilizar nomes de columna cualificados. Ver o apartado *'Nomes de columna cualificados'* ao principio da actividade.
* Se as condicións de composición van na cláusula WHERE, hai que ter en conta que primeiro debemos poñer todas aquelas condicións que non sexan de composición. Deste xeito o produto cartesiano previo ao *join* terá menos filas que combinar.
* Se unha táboa ten unha clave primaria composta, nas condicións de composición hai que facer referencia á clave enteira. Exemplo: Temos unha táboa *t1* que ten unha clave primaria composta formada polas columnas (*cp1,cp2*), e unha táboa *t2* que ten unha clave foránea composta polas columnas (*cf1,cf2*) que fai referencia á clave primaria da táboa t1. A condición de composición podería ser:

|  |  |
| --- | --- |
| t1.cp1 = t2.cf1 and t1.cp2 = t2.cf2 | Condición composta |
| (t1.cp1, t1.cp2) = (t2.cf1,t2.cf2) | Comparación de valores tipo fila |

* + 1. Unión de consultas (UNION)

A cláusula UNION utilízase para combinar o resultado de varias consultas nun único conxunto de resultados. A sintaxe é:

Sentenza SELECT

UNION [ALL | DISTINCT]

Sentenza SELECT

[UNION [ALL | DISTINCT]

Sentenza SELECT] ...

* As opcións ALL e DISTINCT permiten indicar se hai que mostrar as filas duplicadas ou non. O valor por defecto é DISTINCT. Para que se mostren as filas duplicadas hai que poñer a opción ALL.
* As sentenzas SELECT que se utilizan na UNION teñen que ter o mesmo número de columnas na lista de selección. As columnas seleccionadas teñen que ser do mesmo tipo en todas as sentenzas SELECT. Por exemplo, se a primeira consulta selecciona tres columnas, o resto das sentenzas SELECT deben seleccionar tamén tres columnas; se a primeira columna da primeira consulta é de tipo char(50), a primeira columna do resto de sentenzas SELECT ten que ser de tipo char(50).
* Os nomes das columnas da táboa resultante da UNION son os que corresponden aos nomes das columnas da primeira sentenza SELECT.
* Restricións para as sentenzas SELECT que participan nunha unión:
* A cláusula INTO OUTFILE só a pode levar a última sentenza.
* Non se pode utilizar a opción HIGH\_PRIORITY. Non se mostra erro no caso de poñela na primeira sentenza, pero non se ten en conta. No caso de poñela en calquera das outras sentenzas, móstrase unha mensaxe de erro de sintaxe.
* Cando se queren utilizar as cláusulas ORDER BY ou LIMIT, hai que pechar as sentenzas SELECT entre parénteses.
* Non ten efecto a cláusula ORDER BY dentro dos parénteses, xa que a UNION produce un conxunto de filas desordenadas. Só ten sentido utilizar a cláusula ORDER BY para unha sentenza SELECT individual cando se combina coa cláusula LIMIT.

Exemplo extraído do manual de referencia de MySQL 5.6:

(SELECT a FROM tbl\_name WHERE a=10 AND B=1 ORDER BY a LIMIT 10)

UNION

(SELECT a FROM tbl\_name WHERE a=11 AND B=2 ORDER BY a LIMIT 10);

* No caso de querer ordenar as filas que forman o conxunto de resultados, hai que poñer a cláusula ORDER BY fóra dos parénteses, despois da última sentenza SELECT. Tamén é posible limitar o número de filas que forman o conxunto de resultados poñendo a cláusula LIMIT fóra dos parénteses, despois da última sentenza SELECT.

Exemplo extraído do manual de referencia de MySQL 5.6:

(SELECT a FROM tbl\_name WHERE a=10 AND B=1)

UNION

(SELECT a FROM tbl\_name WHERE a=11 AND B=2)

ORDER BY a LIMIT 10;

* Tarefa 4. Combinar o resultado de varias sentenzas SELECT para obter un conxunto de resultados único utilizando UNION.
  1. Tarefas

As tarefas propostas son as seguintes:

* Tarefa 1. Realizar consultas con datos de máis dunha táboa utilizando unha composición interna.
* Tarefa 2. Realizar consultas con datos de máis dunha táboa utilizando unha composición externa.
* Tarefa 3. Realizar consultas dunha táboa consigo mesma.
* Tarefa 4. Combinar o resultado de varias sentenzas SELECT para obter un conxunto de resultados único utilizando UNION.
  + 1. Tarefa 1. Realizar consultas con datos de máis dunha táboa utilizando unha composición interna

A tarefa consiste en realizar as seguintes consultas utilizando unha composición interna con condición de igualdade.

Sobre a base de datos tendaBD

* Tarefa 1.1. Seleccionar os artigos de cor negra e mostrar o seu número, nome e peso, así como o nome do provedor.
* Tarefa 1.2. Seleccionar para todos os apelidos, nome e o nome da provincia na que residen. Os dous primeiros díxitos do código postal (*clt\_cp*) corresponden ao código da provincia na que reside o cliente. Ordenar o resultado polo nome da provincia, e dentro da provincia, polos apelidos e nome, alfabeticamente.
* Tarefa 1.3. Mostrar para cada venda: nome e apelidos do cliente, día, mes, e ano da venda (cada un nunha columna).
* Tarefa 1.4. Mostrar unha lista que conteña: número de vendas, número de artigos vendidos, suma de unidades vendidas e a media dos prezos unitarios dos artigos vendidos.
* Tarefa 1.5. Seleccionar para cada artigo o seu número, nome, peso e o nome que corresponde ao peso (*peso\_nome*), tendo en conta a información contida na táboa *pesos*, que da un nome aos pesos en función do intervalo ao que pertence. Ordenar o resultado polo peso do artigo, de maior a menor.
* Tarefa 1.6. Mostrar para cada venta: nome e apelidos do cliente, a data da venta con formato dd/mm/aa e os días transcorridos dende que se fixo a venta. Ordenar o resultado polo número de días transcorridos dende a venta.
* Tarefa 1.7. Seleccionar os nomes das provincias nas que temos clientes.
* Tarefa 1.8. Seleccionar para cada venda:
* Datos da venda: identificador e data da venda.
* Datos do cliente: nome do cliente (nome e apelidos separados por coma).
* Datos do empregado: nome do empregado (nome e apelidos separados por coma).

Mostrar os datos ordenados polos apelidos e nome do cliente.

* Tarefa 1.9. Seleccionar información sobre os artigos vendidos. Para cada liña de detalle interesa:
* Datos do cliente: apelidos e nome separados por coma, nunha única columna.
* Datos do artigo: nome, cantidade, prezo unitario, desconto e o importe final para o cliente (resultado de multiplicar a cantidade polo prezo unitario e aplicar o desconto que corresponde).

Mostrar os resultados ordenados polo nome do artigo.

Sobre a base de datos traballadores

* Tarefa 1.10. Seleccionar o número e nome de departamento, xunto co nome do director, para os departamentos independentes, é dicir, que non dependen de ningún outro departamento.
* Tarefa 1.11. Mostrar nome (só nome, sen apelidos) e enderezo do *centro* ao que pertence o departamento no que traballa, dos empregados cun nome (sen ter en conta os apelidos) que empece por 'A'.
* Tarefa 1.12. Seleccionar para todos os empregados que non son directores, o nome de departamento no que traballa, o seu nome e salario, o nome e salario do director do seu departamento, e a diferenza do seu salario e o salario do director do departamento. Ordenar o resultado polo nome do departamento.

Solicítase esta información para facer un estudio da diferenza de salarios entre os directores dos departamento e os traballadores que traballan no departamento.

Solución

* Tarefa 1.1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar os artigos de cor negra e mostrar o seu número, nome e peso, así como o

nome do provedor.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** ar**.**art\_codigo **as** Numero**,**

ar**.**art\_nome **as** Articulo**,**

ar**.**art\_peso **as** Peso**,**

pr**.**prv\_nome **as** Provedor

**from** artigos **as** ar **join** provedores **as** pr **on** ar**.**art\_provedor**=**pr**.**prv\_id

**where** ar**.**art\_color**=**'negro'**;**

* Tarefa 1.2

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar para todos os apelidos, nome e o nome da provincia na que residen. Os dous

primeiros díxitos do código postal (clt\_cp) corresponden ao código da provincia na que

reside o cliente. Ordenar o resultado polo nome da provincia, e dentro da provincia,

polos apelidos e nome, alfabeticamente.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** cl**.**clt\_apelidos**,** cl**.**clt\_nome**,** pr**.**pro\_nome

**from** clientes **as** cl **join** provincias **as** pr **on** **left(trim(**cl**.**clt\_cp**),**2**)=**pr**.**pro\_id

**order** **by** pr**.**pro\_nome**,** cl**.**clt\_apelidos**,** cl**.**clt\_nome**;**

* Tarefa 1.3

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Mostrar para cada venda: nome e apelidos do cliente, día, mes, e ano da venda (cada

un nunha columna).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** cl**.**clt\_nome **as** Nome\_cliente**,**

cl**.**clt\_apelidos **as** Apelidos**,**

**day(**ve**.**ven\_data**)** **as** Dia\_venta**,**

**month(**ve**.**ven\_data**)** **as** Mes\_venta**,**

**year(**ve**.**ven\_data**)** **as** Ano\_venta

**from** clientes **as** cl **join** vendas **as** ve **on** cl**.**clt\_id**=**ve**.**ven\_cliente**;**

* Tarefa 1.4

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Mostrar unha lista que conteña: número de vendas, número de artigos vendidos, suma de

unidades vendidas e a media dos prezos unitarios dos artigos vendidos. Facer os

cálculos coa información contidas nas táboas vendas e detalle:vendas

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **count(distinct** ve**.**ven\_id**)** **as** Numero\_vendas**,**

**count(distinct** dv**.**dev\_artigo**)** **as** Numero\_artigos**,**

**sum(**dv**.**dev\_cantidade**)** **as** Suma\_unidades**,**

**round(avg(**dv**.**dev\_prezo\_unitario**),**2**)** **as** Media\_prezo

**from** vendas **as** ve **join** detalle\_vendas **as** dv **on** dv**.**dev\_venda **=** ve**.**ven\_id**;**

* Tarefa 1.5

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar para cada artigo o seu número, nome, peso e o nome que corresponde ao peso

(peso\_nome), tendo en conta a información contida na táboa pesos, que da un nome aos

pesos en función do intervalo ao que pertence. Ordenar o resultado polo peso do artigo,

de maior a menor.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** ar**.**art\_codigo**,**

ar**.**art\_nome**,**

ar**.**art\_peso**,**

pe**.**peso\_nome

**from** artigos **as** ar **join** pesos **as** pe **on** ar**.**art\_peso **between** peso\_min **and** peso\_max

**order** **by** ar**.**art\_peso**;**

* Tarefa 1.6

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Mostrar para cada venda: nome e apelidos do cliente, a data da venta con formato

dd/mm/aa e os días transcorridos dende que se fixo a venta. Ordenar o resultado polo

número de días transcorridos dende a venta.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **distinct** cl**.**clt\_apelidos **as** Apelidos**,**

cl**.**clt\_nome **as** Nome**,**

**date\_format(**ve**.**ven\_data**,** '%d/%m/%Y'**)** **as** Data\_venta**,**

**datediff(**curdate**(),**ve**.**ven\_data**)** **as** Dias\_diferencia

**from** vendas **as** ve **join** clientes **as** cl **on** ve**.**ven\_cliente**=**cl**.**clt\_id

**order** **by** Dias\_diferencia**;**

* Tarefa 1.7

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar os nomes das provincias nas que temos clientes.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **distinct** pro\_nome

**from** provincias **as** pr

**join** clientes **as** cl **on** pr**.**pro\_id **=** **left(trim(**cl**.**clt\_cp**),**2**);**

* Tarefa 1.8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar para cada venda:

Datos da venda: identificador e data da venda.

Datos do cliente: nome do cliente (nome e apelidos separados por coma).

Datos do empregado: nome do empregado (nome e apelidos separados por coma).

Mostrar os datos ordenados polos apelidos e nome do cliente

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** ve**.**ven\_id **as** Numero\_venda**,**

ve**.**ven\_data **as** Data\_da\_venda**,**

**concat(**cl**.**clt\_apelidos**,**', '**,**cl**.**clt\_nome**)** **as** Apelidos\_nome\_cliente**,**

**concat(**em**.**emp\_apelidos**,**', '**,**em**.**emp\_nome**)** **as** Apelidos\_nome\_empregado

**from** vendas **as** ve

**join** clientes **as** cl **on** ve**.**ven\_cliente **=** cl**.**clt\_id

**join** empregados **as** em **on** ve**.**ven\_empregado **=** em**.**emp\_id

**order** **by** Apelidos\_nome\_cliente**;**

* Tarefa 1.9

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar información sobre os artigos vendidos. Para cada liña de detalle interesa:

– Datos do cliente: apelidos e nome separados por coma, nunha única columna.

– Datos do artigo: nome, cantidade, prezo unitario, desconto e o importe final para

o cliente (resultado de multiplicar a cantidade polo prezo unitario e aplicar o

desconto que corresponde).

Mostrar os resultados ordenados polo nome do artigo.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **concat(**cl**.**clt\_apelidos**,**', '**,**cl**.**clt\_nome**)** **as** Apelidos\_nome\_cliente**,**

ar**.**art\_nome **as** Artigo**,**

dv**.**dev\_cantidade **as** Cantidade**,**

dv**.**dev\_prezo\_unitario **as** Prezo\_unitario**,**

dv**.**dev\_desconto **as** '% Desconto'**,**

**round((**dv**.**dev\_prezo\_unitario**\***dv**.**dev\_cantidade**)\*(**1-dv**.**dev\_desconto**/**100**),**2**)** **as** Total

**from** vendas **as** ve

**join** clientes **as** cl **on** ve**.**ven\_cliente **=** cl**.**clt\_id

**join** detalle\_vendas **as** dv **on** ve**.**ven\_id **=** dv**.**dev\_venda

**join** artigos **as** ar **on** dv**.**dev\_artigo **=** ar**.**art\_codigo

**order** **by** Artigo**;**

* Tarefa 1.10

/\* Seleccionar o número e nome de departamento, xunto co nome do director, para os

departamentos independentes, é dicir, que non dependen de ningún outro departamento.\*/

**select** de**.**depNumero **as** Numero**,**

de**.**depNome **as** Nome**,**

em**.**empNome **as** Director

**from** departamento **as** de **join** empregado **as** em **on** de**.**depDirector**=**em**.**empNumero

**where** de**.**depDepende **is** **null;**

* Tarefa 1.11

/\* Mostrar nome (só nome, sen apelidos) e enderezo do centro ao que pertence o

departamento no que traballa, dos empregados cun nome (sen ter en conta os apelidos)

que empece por 'A'.\*/

**select** **trim(right(**em**.**empNome**, length(**em**.**empNome**)-**locate**(**','**,** em**.**empNome **))) as** Nome**,**

ce**.**cenEnderezo **as** Enderezo

**from** empregado **as** em

**join** departamento **as** de **on** em**.**empDepartamento **=** de**.**depNumero

**join** centro **as** ce **on** de**.**depCentro**=**ce**.**cenNumero

**where** **trim(right(**em**.**empNome**,** **length(**em**.**empNome**)-**locate**(**','**,** em**.**empNome**))) like** 'A%'**;**

/\* Ollo: A función TRIM elimina espazos en branco ao principio e ao final da cadea.

Non é o mesmo utilizar right(em.empNome, length(em.empNome)-locate(',', em.empNome)-1)

xa que hai un empregado que non ten espazo en branco despois da coma e ten A como

segunda letra do nome: SANTOS,SANCHO \*/

* Tarefa 1.12

/\* Seleccionar para todos os empregados que non son directores, o nome de departamento

no que traballa, o seu nome e salario, o nome e salario do director do seu

departamento, e a diferenza do seu salario e o salario do director do departamento.

Ordenar o resultado polo nome do departamento.\*/

**select** de**.**depNome **as** Departamento**,**

em1**.**empNome **as** Empregado**,**

em1**.**empSalario **as** Salario\_empregado**,**

em2**.**empNome **as** Nome\_Director**,**

em2**.**empSalario **as** Salario\_director**,**

em2**.**empSalario**-**em1**.**empSalario **as** Diferenza

**from** empregado **as** em1 **join** departamento **as** de **on** em1**.**empDepartamento **=** de**.**depNumero

**join** empregado **as** em2 **on** de**.**depDirector**=**em2**.**empNumero

**where** em1**.**empNumero **<>** em2**.**empNumero

**order** **by** de**.**depNome**;**

* + 1. Tarefa 2. Realizar consultas con datos de máis dunha táboa utilizando unha composición externa

A tarefa consiste en realizar as seguintes consultas utilizando unha composición externa.

Sobre a base de datos tendaBD

* Tarefa 2.1. Para todos os clientes con identificador inferior ou igual a 10, seleccionar os datos das vendas que se lle fixeron. Hai que mostrar para cada venda, o identificador do cliente, apelidos, nome e data de venda. Se a algún deses clientes non se lle fixo ningunha venda, deberá aparecer na lista co seu identificador, nome, apelidos, e o texto 'SEN COMPRAS' na columna da data da venda.
* Tarefa 2.2. Seleccionar os nomes das provincias nas que non temos ningún cliente.
* Tarefa 2.3. Seleccionar o código (*emp\_id*), apelidos e nome de todos os empregados. Engadir unha columna na lista de selección, co alias Vendas, na que se mostre o literal 'Si' se o empregado fixo algunha venda, ou o literal 'Non' no caso de que aínda non fixera ningunha venda.

Solución

* Tarefa 2.1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Obter os datos: identificador de cliente, apelidos e data de venda para todas as

compras dos clientes con identificador inferior ou igual a 10. Se non existen compras,

deberá aparecer na lista co texto 'SEN COMPRAS'.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **distinct** cl**.**clt\_id **as** Codigo\_cliente**,**

cl**.**clt\_apelidos **as** Apelidos**,**

cl**.**clt\_nome **as** Nome**,**

**ifnull(**ve**.**ven\_data**,**'SEN COMPRAS'**)** **as** Data\_compra

**from** clientes **as** cl **left** **join** vendas **as** ve **on** cl**.**clt\_id**=**ve**.**ven\_cliente

**where** cl**.**clt\_id**<=**10**;**

* Tarefa 2.2

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar os nomes das provincias nas que non temos ningún cliente.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** pro\_nome

**from** provincias **as** pr

**left** **join** clientes **as** cl **on** pr**.**pro\_id **=** **left(trim(**cl**.**clt\_cp**),**2**)**

**where** clt\_nome **is** **null;** /# Vale calquera columna da táboa clientes

* Tarefa 2.3

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Tarefa 2.3. Seleccionar o código (*emp\_id*), apelidos e nome de todos os empregados.

Engadir unha columna na lista de selección, co alias Vendas, na que se mostre o literal

'Si' se o empregado fixo algunha venda, e o literal 'Non' no caso de que aínda non

fixera ningunha venda.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** **distinct** emp\_id**,** emp\_dni**,** emp\_apelidos**,** emp\_nome**,**

**if(**ven\_id **is** **null,**'Non'**,**'Si'**)** **as** Vendas

**from** empregados **left** **join** vendas **on** emp\_id **=** ven\_empregado**;**

* + 1. Tarefa 3. Realizar consultas dunha táboa consigo mesma

A tarefa consiste en realizar as seguintes consultas compoñendo unha táboa consigo mesma.

Sobre a base de datos tendaBD

* Tarefa 3.1. Obter unha lista de todos os artigos que teñan un prezo de compra superior ao prezo de compra do artigo con código ' 0713242'.

Sobre a base de datos traballadores

* Tarefa 3.2. Mostrar o número, nome e salario de todos os empregados que teñen un salario maior que o salario do empregado número 180. Engadir na lista de selección unha columna para mostrar o salario do empregado número 180.

Solución

* Tarefa 3.1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Obter unha lista de todos os artigos que teñan un prezo de compra superior ao prezo

de compra do artigo con código '0713242'.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** ar1**.**art\_codigo **as** Numero**,**

ar1**.**art\_nome **as** Nome**,**

ar1**.**art\_pc **as** Prezo\_compra**,**

ar2**.**art\_pc **as** Prezo\_artigo\_8

**from** artigos **as** ar1 **join** artigos **as** ar2 **on** ar1**.**art\_pc**>**ar2**.**art\_pc

**where** ar2**.**art\_codigo**=**'0713242'**;**

* Tarefa 3.2

/\* Mostrar o número, nome e salario de todos os empregados que teñen un salario maior

que o do empregado número 180. Engadir na lista de selección unha columna para mostrar

o salario do empregado número 180.\*/

**select** em1**.**empNumero**,** em1**.**empNome**,** em1**.**empSalario**,** em2**.**empSalario **as** 'Empregado 180'

**from** empregado **as** em1 **join** empregado **as** em2 **on** em1**.**empNumero **!=** em2**.**empNumero

**where** em2**.**empNumero**=**180

**and** em1**.**empSalario **>** em2**.**empSalario

* + 1. Tarefa 4. Combinar o resultado de varias sentenzas SELECT para obter un conxunto de resultados único utilizando UNION

A tarefa consiste en realizar as seguintes consultas utilizando UNION.

Sobre a base de datos tendaBD

* Tarefa 4.1. Seleccionar todos os artigos negros, xunto cos artigos que pesan máis de 5000 gramos, escribindo dúas consultas, e empregando o operador de unión de consultas.
* Tarefa 4.2. Para facer un envío de cartas con información dunha nova campaña por correo postal, seleccionar apelidos, nome, enderezo, código postal e poboación de todos os clientes e de todos os empregados. Na lista hai que diferenciar se a persoa é cliente ou empregado. Ordenar o resultado por orden alfabético de apelidos e nome.

Solución

* Tarefa 4.1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Seleccionar todos os artigos negros, xunto cos artigos que pesan máis de 5000 gramos,

escribindo dúas consultas, e empregando o operador de unión de consultas

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**select** art\_codigo **as** Codigo**,**

art\_nome **as** Nome**,**

art\_peso **as** Peso**,**

art\_color **as** Color

**from** artigos

**where** art\_peso**>**5000

**union**

**select** art\_codigo **as** Codigo**,**

art\_nome **as** Nome**,**

art\_peso **as** Peso**,**

art\_color **as** Color

**from** artigos

**where** art\_color**=**"negro"

**order** **by** Codigo**;**

* Tarefa 4.2

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Para facer un envío de cartas con información dunha nova campaña por correo postal,

seleccionar apelidos, nome, enderezo, código postal e poboación de todos os clientes

e de todos os empregados. Na lista hai que diferenciar se a persoa é cliente ou

empregado. Ordenar o resultado por orden alfabético de apelidos e nome.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**(select** 'Cliente' **as** Persoa**,**

cl**.**clt\_apelidos **as** Apelidos**,**

cl**.**clt\_nome **as** Nome**,**

cl**.**clt\_enderezo **as** Enderezo**,**

cl**.**clt\_cp **as** CP**,**

cl**.**clt\_poboacion **as** Poboacion**,**

pr**.**pro\_nome

**from** clientes **as** cl

**join** provincias **as** pr **on** **left(trim(**cl**.**clt\_cp**),**2**)=**pr**.**pro\_id

**where** clt\_baixa **is** **null)**

**union**

**(select** 'Empregado' **as** Persoa**,**

em**.**emp\_apelidos **as** Apelidos**,**

em**.**emp\_nome **as** Nome**,**

em**.**emp\_enderezo **as** Enderezo**,**

em**.**emp\_cp **as** CP**,**

em**.**emp\_poboacion **as** Poboacion**,**

pr**.**pro\_nome **as** Provincia

**from** empregados **as** em

**join** provincias **as** pr **on** **left(trim(**em**.**emp\_cp**),**2**)=**pr**.**pro\_id **)**

**order** **by** Apelidos**,**Nome**;**

1. Materiais
   1. Documentos de apoio ou referencia

* HUESO IBAÑEZ, Luis. *Bases de datos*. Ciclos Formativos Ra-Ma, 2012.
* SANCHEZ, Jorge. *Manual de Gestión e bases de datos*.

<http://www.jorgesanchez.net/bd/sgbd.html>

* CAMPS PARÉ, Rafael, CASILLAS SANTILLÁN, Luís Alberto, COSTAL COSTA, Dolors, GILBERT GINESTÁ, Marc, MARTÍN ESCOFET, Carme, PÉREZ MORA, Óscar. *Bases de datos.* UOC, 2007.

<http://ocw.uoc.edu/informatica-tecnologia-y-multimedia/bases-de-datos/Course_listing>

* *Manual de referencia de MySQL 5.6.* <http://dev.mysql.com/doc/>
  1. Recursos didácticos
* Ordenadores con conexión a Internet, que terán instalado o sistema xestor de bases de datos MySQL e o cliente MySQL Workbench.
* Proxector.
* Material didáctico subministrado polo profesorado en papel e/ou formato dixital.
* Máquina virtual para exame que terá instalado o software estritamente necesario para a realización da tarefa de avaliación mediante as probas escritas prácticas.
* Manual de referencia de MySQL.
  1. Material auxiliar

O material auxiliar anexo a esta actividade está almacenado na carpeta CSIFC02\_MP0484\_V000502\_UD05\_A02\_ConsultasMultitaboa\_Anexos que contén:

* O arquivo GuiaWorkbench.docx que é unha guía básica de MySQL Workbench 6.3.
* O arquivo V000502\_scriptsSQL.zip cos scripts necesarios para crear as bases de datos empregadas nos exemplos, tarefas e probas de avaliación.

1. Avaliación

Criterios de avaliación

|  |  |  |  |
| --- | --- | --- | --- |
| Criterios de avaliación seleccionados  para esta actividade | Tipo de  evidencia de aprendizaxe | Instrumento de avaliación | Peso na cualificación  da UD |
| * CA5.3 Realizáronse consultas sobre o contido de varias táboas mediante composicións internas. | * Exame en papel e en formato dixital. | * **TO.2** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións internas. Escala de valores (observación indirecta). | 15 |
| * CA5.4 Realizáronse consultas sobre o contido de varias táboas mediante composicións externas. | * **TO.3** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións externas. Escala de valores (observación indirecta) | 10 |
| * CA5.6 Realizáronse unións de consultas. | * **TO.5** - Documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando unións. Escala de valores (observación indirecta). | 6 |

Modelo de proba combinada para TO.2, TO.3 e TO.5

Proponse unha proba práctica para avaliar CA5.3, CA5.4 e CA5.6 mediante observación indirecta sobre un documento de rexistro da práctica. A solución da práctica deberá ser entregada polo alumnado nun documento escrito e en dixital.

* O documento escrito debe conter:
* Bloque de sentenzas de consulta solicitado.
* Imaxe capturada co resultado da execución do bloque dende o cliente Workbench.
* O arquivo dixital debe conter:
* Arquivos .sql co código correspondente ao bloque de sentenzas.

A parte práctica realizarase na base de datos *equipos*. Subminístrase o arquivo *equipos.sql* co script que permite crear esa base de datos. A base de datos está formada polas táboas que se describen a continuación e a relación entre táboas móstrase no diagrama E/R deseñado con Workbench que aparece despois da descrición.

* Táboa *equipo*

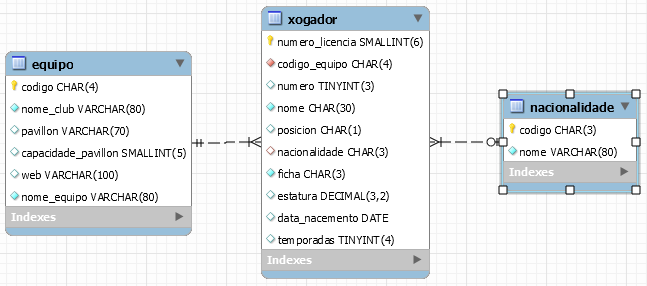
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| codigo | char(4) | non | Primaria | Código do equipo. Para enlazar con outras táboas |
| nome\_club | varchar(80) | non |  | Nome do club ao que pertence o equipo |
| pavillon | varchar(70) |  |  | Nome do pavillón no que xogan os partidos como local |
| capacidade\_pavillon | smallint |  |  | Número de espectadores que entran no seu pavillón |
| web | varchar(100) |  |  | Url da web |
| nome\_equipo | varchar(80) | non |  | Nome oficial do equipo na liga |

* Táboa *nacionalidade*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| codigo | char(3) | non | Primaria | Código do país |
| nome | varchar(80) | non |  | Nome do país |

* Táboa *xogador*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| numero\_licencia | smallint | non | Primaria | Código do equipo no que xoga o xogador. Autoincrementado |
| codigo\_equipo | char(4) | non | Foránea | Código do equipo no que xoga o xogador |
| numero | tinyint |  |  | Número de dorsal do xogador |
| nome | char(30) | non |  | Nome do xogador |
| posicion | char(1) |  |  | Posición na que xoga: B = base, P = pivot, A = alero, ...... |
| nacionalidade | char(3) | non | Foránea | Nacionalidade do xogador |
| ficha | char(3) | non |  | Tipo de ficha: EXT=estranxeiro EUR=europeo JFL=nacional |
| estatura | Decimal(3,2) |  |  | Estatura do xogador en metros |
| data\_nacemento | date |  |  | Data de nacemento |
| temporadas | tinyint |  |  | Número de tempadas que leva no equipo |



Proba para CA5.3 sobre consulta multitáboa con composición interna

Escribir as sentenzas SQL que permitan realizar as seguintes consultas mediante composición interna e capturar as imaxes que identifiquen o resultado dende o cliente Workbench. Se o resultado é moi longo, débese mostrar só un anaco do resultado e capturar a imaxe que identifique o número de liñas que devolve a consulta.

* 1.- Seleccionar nome e nacionalidade dos xogadores do equipo de nome 'Asefa Estudiantes', ordenados polo nome do xogador.
* 2.- Mostrar os nomes que corresponden ás nacionalidades dos xogadores (nomes dos países). Non mostrar nomes de países duplicados.
* 3.- Seleccionar nome, nacionalidade e nome do pavillón dos xogadores do equipo de código 'BILB', ordenados alfabeticamente polo nome do xogador.

Proba para CA5.4 sobre consulta multitáboa con composición externa

Escribir as sentenzas SQL que permitan realizar as seguintes consultas mediante composición externa e capturar as imaxes que identifiquen o resultado dende o cliente Workbench. Se o resultado é moi longo, débese mostrar só un anaco do resultado e capturar a imaxe que identifique o número de liñas que devolve a consulta.

* 4.- Mostrar os nomes dos países dos que non temos ningún xogador.
* 5.- Mostrar datos de todos os xogadores ordenados alfabeticamente polo nome do país e en caso de que coincidan, ordenar tamén polo nome do xogador. Os datos son: nome do xogador, nome do país do que ten a nacionalidade, e nome do equipo no que xoga. Se non existe a nacionalidade, ten que aparecer o texto 'Sen confirmar'.

Proba para CA5.6 sobre consulta multitáboa con unión

Escribir as sentenzas SQL que permitan realizar as seguintes consultas mediante unión e capturar as imaxes que identifiquen o resultado dende o cliente Workbench. Se o resultado é moi longo, débese mostrar só un anaco do resultado e capturar a imaxe que identifique o número de liñas que devolve a consulta.

* 6.- Escribir dúas consultas e unilas co operador UNION, para obter unha única táboa de resultados:
* Consulta 1: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos xogadores do equipo de código 'BILB'.
* Consulta 2: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos xogadores do equipo que ten como nome 'Asefa Estudiantes'.
* Crear unha única táboa de resultados, ordenados alfabeticamente polo nome do xogador.

Exemplo de solución para entregar en papel

Proponse a seguinte solución:

* 1.- Seleccionar nome e nacionalidade dos xogadores do equipo de nome 'Asefa Estudiantes', ordenados polo nome do xogador.

/\*Seleccionar nome e nacionalidade dos xogadores do equipo de nome 'Asefa Estudiantes',

ordenados polo nome do xogador.\*/

**select** xo**.**nome **as** Xogador**,**

xo**.**nacionalidade **as** Nacionalidade

**from** xogador **as** xo

**join** equipo **as** eq **on** **(**eq**.**codigo **=** xo**.**codigo\_equipo**)**

**where** eq**.**nome\_equipo**=**'Asefa Estudiantes'

**order** **by** xo**.**nome**;**





* 2.- Mostrar os nomes que corresponden ás nacionalidades dos xogadores (nomes dos países). Non mostrar nomes de países duplicados.

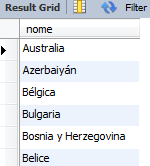
/\*Mostrar os nomes que corresponden ás nacionalidades dos xogadores (nomes dos paises).

Non mostrar nomes de países duplicados.\*/

**select** **distinct** na**.**nome

**from** nacionalidade **as** na

**join** xogador **as** xo **on** xo**.**nacionalidade **=** na**.**codigo**;**





* 3.- Seleccionar nome, nacionalidade e nome do pavillón dos xogadores do equipo de código 'BILB', ordenados alfabeticamente polo nome do xogador.

/\*Mostrar nome, nacionalidade e nome do pavillón dos xogadores do equipo de código

'BILB', ordenados alfabeticamente polo nome do xogador.\*/

**select** xo**.**nome **as** Xogador**,**

xo**.**nacionalidade **as** Nacionalidade**,**

eq**.**pavillon **as** Pavillon

**from** xogador **as** xo

**join** equipo **as** eq **on** **(**eq**.**codigo **=** xo**.**codigo\_equipo**)**

**where** eq**.**codigo**=**'BILB'

**order** **by** xo**.**nome**;**





* 4.- Mostrar os nomes dos países dos que non temos ningún xogador.

/\*Mostrar os nomes que corresponden ás nacionalidades dos xogadores (nomes dos países).

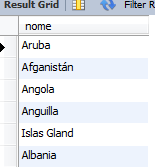
Non mostrar nomes de países duplicados.\*/

**select** **distinct** na**.**nome

**from** nacionalidade **as** na

**left** **join** xogador **as** xo **on** xo**.**nacionalidade **=** na**.**codigo

**where** xo**.**nome **is** **null;** # Vale calquera columna da táboa xogador





* 5.- Mostrar datos de todos os xogadores ordenados alfabeticamente polo nome do país e en caso de que coincidan, ordenar tamén polo nome do xogador. Os datos son: nome do xogador, nome do país do que ten a nacionalidade, e nome do equipo no que xoga. Se non existe a nacionalidade, ten que aparecer o texto 'Sen confirmar'.

/\*Mostrar datos de todos os xogadores ordenados alfabeticamente polo nome do

país e en caso de que coincidan, ordenar tamén polo nome do xogador.

Os datos son: nome do xogador, nome do país do que ten a nacionalidade,

e nome do equipo no que xoga.

Se non existe a nacionalidade, ten que aparecer o texto 'Sen confirmar'.\*/

**select** xo**.**nome **as** Xogador**,**

**ifnull(**na**.**nome**,**'Sen confirmar'**)** **as** Nacionalidade**,**

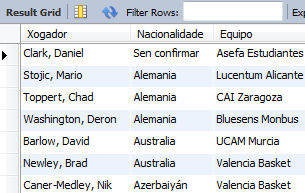
eq**.**nome\_equipo **as** Equipo

**from** xogador **as** xo

**join** equipo **as** eq **on** **(**eq**.**codigo **=** xo**.**codigo\_equipo**)**

**left** **join** nacionalidade **as** na **on** **(**xo**.**nacionalidade **=** na**.**codigo**)**

**order** **by** na**.**nome**,**xo**.**nome**;**





* 6.- Escribir dúas consultas e unilas co operador UNION, para obter unha única táboa de resultados:
* Consulta 1: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos xogadores do equipo de código 'BILB'.
* Consulta 2: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos xogadores do equipo que ten como nome 'Asefa Estudiantes'.
* Crear unha única táboa de resultados, ordenados alfabeticamente polo nome do xogador.

/\*

Escribir dúas consultas e unilas co operador UNION, para obter unha única táboa de

resultados:

- Consulta 1: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos

xogadores do equipo de código 'BILB'.

- Consulta 2: seleccionar nome, nacionalidade, nome do equipo e nome do pavillón dos

xogadores do equipo que ten como nome 'Asefa Estudiantes'.

- Crear unha única táboa de resultados, ordenados alfabeticamente polo nome do

xogador.\*/

(**select** xo**.**nome **as** Xogador**,**

xo**.**nacionalidade **as** Nacionalidade**,**

eq**.**nome\_equipo **as** Equipo**,**

eq**.**pavillon **as** Pavillon

**from** xogador **as** xo **join** equipo **as** eq **on** **(**eq**.**codigo **=** xo**.**codigo\_equipo**)**

**where** eq**.**nome\_equipo**=**'Asefa Estudiantes'

**union**

**select** xo**.**nome **as** Xogador**,**

xo**.**nacionalidade **as** Nacionalidade**,**

eq**.**nome\_equipo **as** Equipo**,**

eq**.**pavillon **as** Pavillon

**from** xogador **as** xo **join** equipo **as** eq **on** **(**eq**.**codigo **=** xo**.**codigo\_equipo**)**

**where** eq**.**codigo**=**'BILB')

**order** **by** Xogador**;**





Exemplo de lista de valoración para TO.2

Proponse a seguinte lista de valoración para o instrumento de avaliación TO.2 (EV) (observación indirecta: documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións internas).

| Nome | Data | |
| --- | --- | --- |
| Indicadores para  CA5.3 Realizáronse consultas sobre o contido de varias táboas mediante composicións internas. | Valoración máxima  15 | Cualificación |
| As sentenzas de consulta non teñen erros sintácticos e cumpren as condicións pedidas. | 14 |  |
| As sentenzas de consulta contidas no arquivo .sql correspondente execútanse correctamente e resolven a consulta solicitada. | 1 |  |
| As imaxes capturadas dende o cliente Workbench reflicten os resultado pedidos e correspóndense co arquivo .sql. | 1 |  |

Exemplo de lista de valoración para TO.3

Proponse a seguinte lista de valoración para o instrumento de avaliación TO.3 (EV) (observación indirecta: documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando composicións externas).

| Nome | Data | |
| --- | --- | --- |
| Indicadores para  CA5.4 Realizáronse consultas sobre o contido de varias táboas mediante composicións externas. | Valoración máxima  10 | Cualificación |
| As sentenzas de consulta non teñen erros sintácticos e cumpren as condicións pedidas. | 8 |  |
| As sentenzas de consulta contidas no arquivo .sql correspondente execútanse correctamente e resolven a consulta solicitada. | 1 |  |
| As imaxes capturadas dende o cliente Workbench reflicten os resultados pedidos e correspóndense co arquivo .sql. | 1 |  |

Exemplo de lista de valoración para TO.5

Proponse a seguinte lista de valoración para o instrumento de avaliación TO.5 (EV) (observación indirecta: documento de rexistro de realización de consultas con datos de máis dunha táboa utilizando unións).

| Nome | Data | |
| --- | --- | --- |
| Indicadores para  CA5.6 Realizáronse unións de consultas. | Valoración máxima  6 | Cualificación |
| As sentenzas de consulta non teñen erros sintácticos e cumpren as condicións pedidas. | 4 |  |
| As sentenza de consulta contidas no arquivo .sql correspondente execútanse correctamente e resolven a consulta solicitada. | 1 |  |
| As imaxes capturadas dende o cliente Workbench reflicten os resultados pedidos e correspóndense co arquivo .sql. | 1 |  |

1. Máis información sobre a norma ISO 3166-1:2013 en <https://es.wikipedia.org/wiki/ISO_3166-1> [↑](#footnote-ref-3)