Dirección Xeral de Educación, Formación Profesional e Innovación Educativa

Material para

a formación profesional inicial

A04. Subconsultas

|  |  |  |
| --- | --- | --- |
| 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** | **A04** | **Subconsultas** |
| 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\_V000504\_UD05\_A04\_Subconsultas.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](#_Toc441056831)

[Contexto da actividade 5](#_Toc441056832)

[Título da actividade 6](#_Toc441056833)

[Resultados de aprendizaxe do currículo 6](#_Toc441056834)

[Obxectivos didácticos e título e descrición da actividade 6](#_Toc441056835)

[Criterios de avaliación 6](#_Toc441056836)

[Contidos 6](#_Toc441056837)

[Actividades de ensino e aprendizaxe e de avaliación, métodos, recursos e instrumentos de avaliación 7](#_Toc441056838)

[2. A04. Consultas con subconsultas 8](#_Toc441056839)

[2.1 Introdución 8](#_Toc441056840)

[2.1.1 Obxectivos 8](#_Toc441056841)

[2.1.2 Software 8](#_Toc441056842)

[2.1.3 Bases de datos de traballo 9](#_Toc441056843)

[2.1.3.1 Base de datos tendaBD 9](#_Toc441056844)

[2.1.3.2 Base de datos practicas5 11](#_Toc441056845)

[2.1.3.3 Base de datos traballadores 12](#_Toc441056846)

[2.2 Actividade 13](#_Toc441056847)

[2.2.1 Subconsultas 13](#_Toc441056848)

[2.2.2 Subconsultas como parte dunha condición 14](#_Toc441056849)

[2.2.2.1 Utilizando operadores relacionais 14](#_Toc441056850)

[Subconsultas que devolven un escalar 14](#_Toc441056851)

[Subconsultas que devolven unha táboa 14](#_Toc441056852)

[2.2.2.2 Utilizando o predicado BETWEEN 16](#_Toc441056853)

[2.2.2.3 Utilizando o predicado IN 17](#_Toc441056854)

[2.2.2.4 Utilizando o predicado EXISTS 18](#_Toc441056855)

[2.2.3 Subconsultas que non forman parte dunha condición 19](#_Toc441056856)

[2.3 Tarefas 19](#_Toc441056857)

[2.3.1 Tarefa 1. Realizar consultas que utilicen os operadores relacionais combinados con subconsultas dentro dunha condición 20](#_Toc441056858)

[Solución 21](#_Toc441056859)

[2.3.2 Tarefa 2. Realizar consultas que utilicen o predicado BETWEEN combinado con subconsultas dentro dunha condición 23](#_Toc441056860)

[Solución 24](#_Toc441056861)

[2.3.3 Tarefa 3. Realizar consultas utilizando subconsultas co predicado IN 24](#_Toc441056862)

[Solución 25](#_Toc441056863)

[2.3.4 Tarefa 4. Realizar consultas utilizando subconsultas co predicado EXISTS 29](#_Toc441056864)

[Solución 29](#_Toc441056865)

[2.3.5 Tarefa 5. Realizar consultas utilizando subconsultas que non forman parte dunha condición 30](#_Toc441056866)

[Solución 30](#_Toc441056867)

[3. Materiais 32](#_Toc441056868)

[3.1 Documentos de apoio ou referencia 32](#_Toc441056869)

[3.2 Recursos didácticos 32](#_Toc441056870)

[3.3 Material auxiliar 32](#_Toc441056871)

[4. Avaliación 33](#_Toc441056872)

[Criterios de avaliación 33](#_Toc441056873)

[Modelo de proba para TO.6 33](#_Toc441056874)

[Proba para CA5.7 sobre consulta con subconsulta 34](#_Toc441056875)

[Exemplo de solución para entregar en papel 34](#_Toc441056876)

[Exemplo de lista de valoración para TO.6 36](#_Toc441056877)

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 |
| A04 | Consultas con subconsultas. | Realizáronse consultas que conteñen a outras consultas como parte da sentenza SELECT. | 7 |

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 |
| --- | --- | --- | --- | --- | --- |
| O4.1 | Realizar consultas que conteñen a outras consultas como parte da sentenza SELECT. | A04 | Consultas con subconsultas. | Realizáronse consultas que conteñen a outras consultas como parte da sentenza SELECT. | 7 |

Criterios de avaliación

|  |
| --- |
| Criterios de avaliación |
| * CA5.7 Realizáronse consultas con subconsultas. |

Contidos

| Contidos |
| --- |
| * Subconsultas |

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 |
| A04. Consultas con subconsultas.   * Realizáronse consultas que conteñen a outras consultas como parte da sentenza SELECT. | * **Tp4.1** Exposición de conceptos sobre sentenzas LMD (Linguaxe de manipulación de datos) que permiten realizar consultas con subconsultas como parte de SELECT. | **Ta4.1** 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. |  | 0.5 |
| * **Tp4.2** Demostración práctica de consultas con subconsultas. | * **Ta4.2** Realización de consultas con subconsultas. | * Guións de sentenzas LMD que solucionan os exercicios de consulta propostos. |  | 6 |
|  | * **Ta4.3** Tarefa de avaliación utilizando o instrumento de avaliación TO.6. | * Exame en papel e en formato dixital. | * Máquina virtual para exame que terá instalado MySQL e MySQL Workbench. * Manual de referencia de MySQL. | * **TO.6** - Documento de rexistro de realización de consultas que conteñen consultas. Escala de valores (observación indirecta). | 0.5 |

1. A04. Consultas con subconsultas
   1. Introdución
      1. Obxectivos

Os obxectivos desta actividade son:

* Realizar consultas que conteñen a outras consultas como parte da sentenza SELECT.
  + 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.

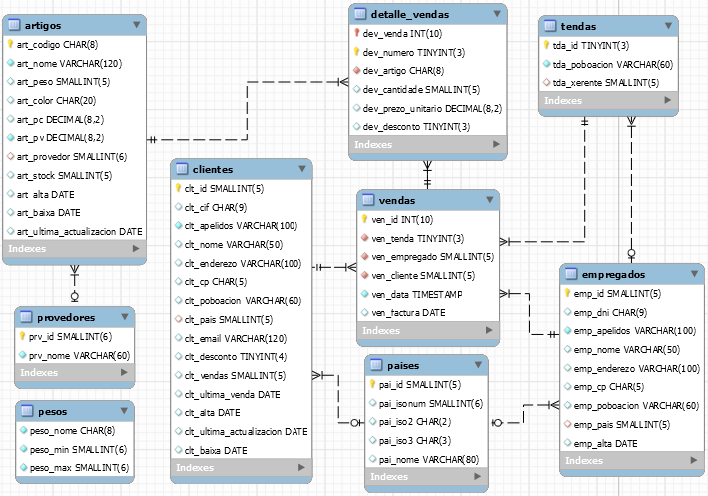
Normalmente, para a proba das consultas realizadas nesta actividade, mostrarase a zona de manipulación de datos de Workbench coas filas que forman a táboa de resultados. Para completar a anterior información ou cando a consulta non pode mostrarse enteira xa que devolve moitas filas, mostrarase ademais 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*

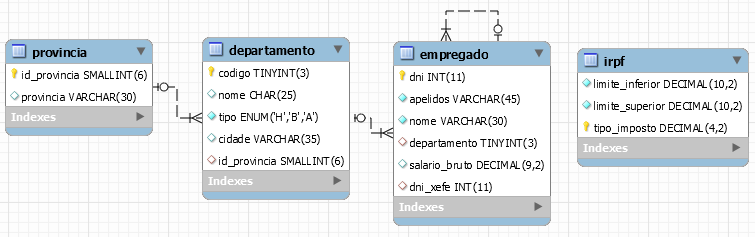
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| ven\_id | int unsigned | Non | Primaria | Identificador da venda. Numeraranse de 1 en adiante de forma automática. |
| ven\_tenda | tinyint unsigned | Non | Foránea | Identificador da tenda na que 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. Subconsultas

Unha subconsulta é unha sentenza SELECT contida noutra sentenza da linguaxe de manipulación de datos. Os resultados da subconsulta afectan ao resultado da sentenza que a contén. Pódese dicir que unha subconsulta está aniñada na consulta externa ou consulta principal e é posible aniñar subconsultas dentro doutras subconsultas ata unha profundidade considerable.

Nesta actividade utilizaranse as subconsultas como parte dunha sentenza SELECT para resolver consultas complexas; tamén reciben o nome de consultas subordinadas.

Exemplo:

SELECT \* FROM t1 WHERE column1 = (SELECT column1 FROM t2);

* A sentenza SELECT column1 FROM t2 sería a subconsulta ou consulta subordinada.
* A sentenza SELECT \* FROM t1 WHERE column1 = sería a consulta externa ou principal.

Algunhas características das subconsultas:

* Aparecen pechadas entre parénteses, dentro da consulta principal.
* Poden retornar un escalar (un valor único), unha táboa formada por unha ou varias columnas e unha ou varias filas ou o valor NULL (se non existe ningunha fila que cumpra as condicións da subconsulta).
* Os nomes de columna que aparecen nunha subconsulta poden facer referencia a táboas da consulta principal. A estas referencias chámaselles 'referencias externas' na subconsulta.
* Pódense utilizar en calquera parte da sentenza SELECT, aínda que o máis habitual é que formen parte dunha condición na cláusula WHERE ou na cláusula HAVING.

Avantaxes do uso de subconsultas:

* Permite escribir consultas estruturadas de forma que é posible illar cada parte da consulta, e descompoñer un problema grande en varios pequenos.
* Proporcionan unha maneira alternativa de resolver consultas que doutro modo necesitarían combinacións de táboas (JOIN) ou unións complexas.
* Fan consultas máis fáciles de ler.
  + 1. Subconsultas como parte dunha condición

A forma máis habitual de utilización das subconsultas é formando parte dunha condición dentro das cláusulas WHERE ou HAVING. A forma de utilizar as subconsultas e os operadores que se poden utilizar na sentenza principal para facer comparacións co resultado da subconsulta, está en función dos valores que devolve a subconsulta, que poden ser valores escalares ou táboas.

* + - 1. Utilizando operadores relacionais

Subconsultas que devolven un escalar

Na súa forma máis simple, unha subconsulta devolve un só valor, e pode ser utilizada como operando en calquera expresión (coma se fose un literal ou unha columna) ou como parámetro dunha función. Normalmente compárase o valor que devolve a subconsulta cunha expresión mediante un operador relacional. Sintaxe:

expresión operador\_relacional (sentenza SELECT escalar)

Exemplo: mostrar *dni*, *apelidos* e *nome* dos empregados do mesmo departamento que o empregado que ten o DNI '12852654'.

* Paso 1: coñecer o departamento ao que pertence ese empregado. Iso pódese saber executando a consulta:

**select** departamento

**from** empregado

**where** dni **=** '12852654'**;**

Resultado da execución:



* Paso 2: a consulta anterior devolve un valor escalar e pode ser utilizada como subconsulta na sentenza SELECT que permite mostrar os datos solicitados dos empregados que traballan nese departamento:

**select** dni**,**apelidos**,**nome

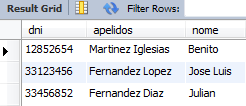
**from** empregado

**where** departamento **=** **(select** departamento

**from** empregado

**where** dni **=** '12852654'**);**

Resultado da execución:



Subconsultas que devolven unha táboa

As subconsultas poden devolver unha táboa cunha fila polo menos. Para a comparación coa subconsulta hai que utilizar os operadores relacionais combinados coas palabras ALL ou ANY. Sintaxe:

expresión operador\_relacional {ALL | ANY} (sentencia SELECT)

* Cando se utiliza a palabra ALL diante do operador relacional, a condición será verdadeira se é verdadeira para todos os valores que devolve a subconsulta.
* Cando se utiliza a palabra ANY diante do operador relacional, a condición é verdadeira se é verdadeira para polo menos un dos valores que devolve a subconsulta.

Exemplo con ALL: mostrar *dni*, *apelidos*, *nome* e *salario\_bruto* dos empregados que teñan un salario bruto superior ao de todos os empregados que traballan no departamento 4.

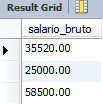
* Paso 1: seleccionar os salarios de todos os empregados do departamento 4:

**select** salario\_bruto

**from** empregado

**where** departamento **=** 4

Resultado da execución:



* Paso 2: a consulta anterior devolve unha táboa que ten varias filas, e unha columna cos salarios de todos os empregados do departamento 4. A sentenza SELECT que hai que construír ten que comparar o salario de cada empregado con todos os valores que devolve a subconsulta, e seleccionar só os que teñen un salario bruto maior que todos eses valores.

**select** dni**,**apelidos**,**nome**,**salario\_bruto

**from** empregado

**where** salario\_bruto **>** **ALL** **(select** salario\_bruto

**from** empregado

**where** departamento **=** 4**);**

Resultado da execución:



Unha solución alternativa para resolver esta consulta consiste en calcular o salario bruto máis alto dos empregados do departamento 4, que sería un valor escalar, e despois comparar o salario bruto de cada empregado con ese valor escalar utilizando un operador relacional.

* Paso 1: calcular o salario máis alto dos empregados do departamento 4.

**select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 4

Resultado da execución:



* Paso 2: o resultado da consulta é un valor escalar e pódese comparar co operador relacional 'maior que' (>) .

**select** dni**,**apelidos**,** nome**,** salario\_bruto

**from** empregado

**where** salario\_bruto **>** **(select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 4**);**

Exemplo con ANY: mostrar *dni*, *apelidos*, *nome, salario\_bruto* e *departamento* dos empregados que traballan nalgún departamento da provincia de Lugo (*id\_provincia* = 27).

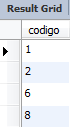
* Paso 1: Seleccionamos os códigos dos departamentos que están situados na provincia de Lugo.

**select** codigo

**from** departamento

**where** id\_provincia **=** '27'**;**

Resultado da execución:



* Paso 2: na sentenza principal hai que comparar os departamentos de cada empregado dos valores que devolve a subconsulta, e seleccionar só aqueles que teñan un departamento igual a algún dos valores que devolve a subconsulta.

**select** dni**,**apelidos**,** nome**,** salario\_bruto**,** departamento

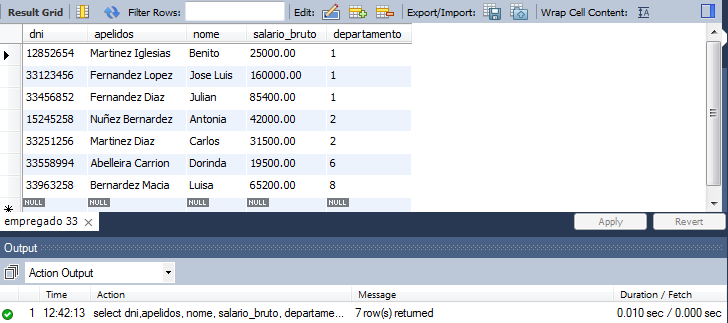
**from** empregado

**where** departamento **=** **ANY** **(select** codigo

**from** departamento

**where** id\_provincia **=** '27'**);**

Resultado da execución:



* Tarefa 1. Realizar consultas que utilicen os operadores relacionais combinados con subconsultas dentro dunha condición.
  + - 1. Utilizando o predicado BETWEEN

Pódese utilizar o predicado BETWEEN con subconsultas que devolven un escalar. Serve para comparar unha expresión cun intervalo de valores que se obteñen dunha ou dúas subconsultas.

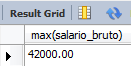
Exemplo: mostrar *dni*, *apelidos*, *nome, departamento*  e *salario\_bruto* dos empregados que teñan un salario bruto entre o salario máis alto do departamento 2 e o salario máis alto do departamento 8.

* Paso 1: calcular o salario máis alto dos empregados do departamento 2.

**select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 2**;**

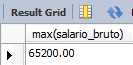


* Paso 2: calcular o salario máis alto dos empregados do departamento 8.

**select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 8**;**



* Paso 3: subconsulta final utilizando as dúas anteriores.

**select** dni**,**apelidos**,**nome**,**departamento**,**salario\_bruto

**from** empregado

**where** salario\_bruto **between** **(select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 2**)**

**and** **(select** **max(**salario\_bruto**)**

**from** empregado

**where** departamento **=** 8**);**



* Tarefa 2. Realizar consultas que utilicen o predicado BETWEEN combinado con subconsultas dentro dunha condición.
  + - 1. Utilizando o predicado IN

O predicado IN utilízase cando a subconsulta devolve unha táboa cunha ou máis filas e permite comparar o valor que corresponde a unha expresión cos valores que devolve a subconsulta. Se de antemán se sabe que a subconsulta vai devolver unha táboa dunha fila, non é recomendable utilizar este predicado e deberíanse de utilizar operadores relacionais.

Sintaxe:

expresión [ NOT ] IN (sentenza SELECT)

* A condición é verdadeira cando o valor que representa a expresión é igual a algún dos valores que devolve a subconsulta.
* Cando se utiliza o operador NOT diante, a condición é verdadeira cando o valor que representa a expresión é distinto de todos os valores que devolve a subconsulta. No caso que na lista de valores estea incluído o valor NULL hai que ter en conta que a comparación de calquera valor co valor NULL devolve NULL e non *true* ou *false*, o que pode provocar problemas e obter resultados que non son correctos.
* Cando comparamos o valor dunha expresión cos valores que devolve unha subconsulta que produce como resultado unha táboa de máis dunha fila:
* IN equivale a unha condición = ANY
* NOT IN equivale a unha condición != ALL

Exemplo: mostrar *dni*, *apelidos*, *nome, salario\_bruto* e *departamento* dos empregados que traballan nalgún departamento que non estea situado na provincia de Lugo.

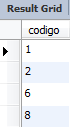
* Paso 1: seleccionar os códigos dos departamentos que están situados na provincia de Lugo.

**select** codigo

**from** departamento

**where** id\_provincia **=** '27'**;**

Resultado da execución:



* Paso 2: na sentenza principal hai que comparar o departamento de cada empregado cos valores que devolve a subconsulta, e seleccionar só aqueles que teñan un departamento distinto a algún dos valores que devolve a subconsulta.

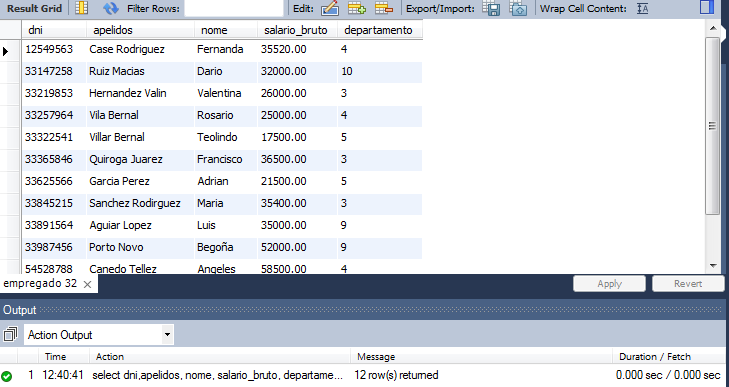
**select** dni**,**apelidos**,** nome**,** salario\_bruto**,** departamento

**from** empregado

**where** departamento **NOT IN** **(select** codigo

**from** departamento

**where** id\_provincia **=** '27'**);**



* Tarefa 3. Realizar consultas que utilicen o predicado IN combinado con subconsultas dentro dunha condición.
  + - 1. Utilizando o predicado EXISTS

Permite comprobar se a subconsulta devolve algunha fila ou non. Os valores que devolve a subconsulta non se comparan con ningún outro valor.

Sintaxe:

[ NOT ] EXISTS (sentencia SELECT)

* A condición é verdadeira se a subconsulta devolve unha o máis filas, e é falsa se a subconsulta non devolve ningunha fila.

Para que este tipo de condición teña utilidade, a subconsulta debe conter algunha condición que devolva resultados diferentes para cada fila da táboa da sentenza principal. Isto supón que nalgunha condición da subconsulta ten que intervir algunha columna das táboas da consulta principal ('referencias externas'); posto que se isto non fose así, a condición sería verdadeira ou falsa para todas as filas da consulta principal.

Exemplo: mostrar *dni*, *apelidos*, *nome, salario\_bruto* e *departamento* dos compañeiros de departamento do empregados con DNI 33456345, que teñan un salario maior có seu.

**select** dni**,**apelidos**,**nome**,**salario\_bruto,departamento

**from** empregado **as** emp1

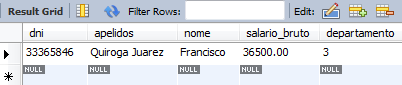
**where** **exists** **(** **select** emp2**.**departamento

**from** empregado **as** emp2

**where** emp2**.**dni **=** '33845215'

**and** emp1**.**departamento **=** emp2**.**departamento

**and** emp1**.**salario\_bruto **>** emp2**.**salario\_bruto**);**



* Tarefa 4. Realizar consultas que utilicen o predicado EXISTS combinado con subconsultas dentro dunha condición.
  + 1. Subconsultas que non forman parte dunha condición

As subconsultas non só se utilizan como parte dunha condición nas cláusulas WHERE ou HAVING, se non que poden aparecer noutras partes dunha instrución SELECT.

Exemplo: mostrar os salarios brutos de todos os empregados e a diferenza entre o seu salario e o salario medio de todos os empregados que teñan como xefe ao empregado con DNI 33252141.

**select** dni**,** apelidos**,** nome**,** salario\_bruto**,**

**round(**salario\_bruto**-(select** **avg(**salario\_bruto**)** **from** empregado**),**2**)** **as** diferenza

**from** **(select** dni**,** apelidos**,** nome**,** salario\_bruto

**from** empregado

**where** dni\_xefe **=**'33456852'**)** **as** xefe\_33456852

**order** **by** diferenza**;**



* Tarefa 5. Realizar consultas que utilicen subconsultas que non forman parte dunha condición.
  1. Tarefas

As tarefas propostas son as seguintes:

* Tarefa 1. Realizar consultas que utilicen os operadores relacionais combinados con subconsultas dentro dunha condición.
* Tarefa 2. Realizar consultas que utilicen o predicado BETWEEN combinado con subconsultas dentro dunha condición.
* Tarefa 3. Realizar consultas que utilicen o predicado IN combinado con subconsultas dentro dunha condición.
* Tarefa 4. Realizar consultas que utilicen o predicado EXISTS combinado con subconsultas dentro dunha condición.
* Tarefa 5. Realizar consultas que utilicen subconsultas que non forman parte dunha condición.
  + 1. Tarefa 1. Realizar consultas que utilicen os operadores relacionais combinados con subconsultas dentro dunha condición

A tarefa consiste en realizar as seguintes consultas utilizando operadores relacionais pra relacionar unha consulta principal con subconsultas dentro dunha condición.

Sobre a base de datos tendaBD

* Tarefa 1.1. Seleccionar código e peso dos artigos que teñen o peso máis pequeno có peso do artigo de código '077WN45'.
* Tarefa 1.2. Seleccionar código, nome, peso e cor dos artigos da mesma cor có artigo de código '242C283', e que teñan un peso maior ou igual co peso medio de todos os artigos.
* Tarefa 1.3. Seleccionar código da tenda e nome do xerente das tendas nas que se vendeu polo menos unha unidade do artigo de código ' 077WN45'.
* Tarefa 1.4. Mostrar a lista de artigos cun prezo de venta maior có prezo de venta do artigo máis barato de cor negra. Obter o mesmo resultado nos dous casos seguintes:
* Utilizando *any*.
* Utilizando a función *min()*.
* Tarefa 1.5. Seleccionar código de artigo, descrición do artigo e código do provedor para os artigos que subministra o provedor que ten o nome SEAGATE.
* Tarefa 1.6. Mostrar información do artigo máis caro, Seleccionando o seu código, nome, prezo de venta e nome do provedor que o subministra.

Sobre a base de datos traballadores

* Tarefa 1.7. Mostrar nomes e salarios dos empregados que teñan salario maior có salario de tódolos empregados do departamento 122. O resultado móstrase ordenado por nome de empregado.
* Tarefa 1.8. Mostrar os nomes e salarios dos empregados con salarios maiores ou iguais a o de Claudia Fierro, ordenados alfabeticamente. Resolver esta consulta:
* con subconsulta.
* sen subconsulta utilizando JOIN.
* Tarefa 1.9. Mostrar nome e presuposto dos departamentos que teñen o presuposto máis alto e o máis baixo.
* Tarefa 1.10. Mostrar número, nome, data de ingreso na empresa e nome do departamento no que traballa o empregado ou empregados que levan máis tempo na empresa.
* Tarefa 1.11. Mostrar nome e salario dos empregados que cumpran algunha das seguintes condicións:
* Ingresaron na empresa despois do 1-1-88.
* Ingresaron na empresa antes do 1-1-88 pero teñen un salario inferior ao salario medio de todos os empregados da empresa.
* Tarefa 1.12. Mostrar a extensión telefónica asignada a máis empregados indicando o número de empregados que a comparten. Se hai máis dunha, deben aparecer todas.

Solución

* Tarefa 1.1

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

Seleccionar código e peso dos artigos que teñen o peso máis pequeno có peso do artigo

de código '077WN45'

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

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

art\_peso Peso

**from** artigos

**where** art\_peso **<** **(select** art\_peso **from** artigos **where** art\_codigo**=**'077WN45'**)**

**order** **by** art\_peso**;**

* Tarefa 1.2

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

Seleccionar código, nome, peso e cor dos artigos da mesma cor có artigo de código

'242C283', e que teñan un peso maior ou igual co peso medio de todos os artigos

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

**select** art\_codigo**,**

art\_nome**,**

art\_peso**,**

art\_color

**from** artigos

**where** art\_color **=** **(select** art\_color **from** artigos **where** art\_codigo **=** '242C283'**)**

**and** art\_peso **>=** **(select** **avg(**art\_peso**)** **from** artigos**);**

* Tarefa 1.3

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

Seleccionar código da tenda e nome do xerente, das tendas nas que se vendeu polo menos

unha unidade do artigo de código 077WN45

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

**select** te.tda\_id **as** Codigo\_tenda**,**

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

**from** tendas **as** te **join** empregados **as** em **on** te.tda\_xerente **=** em.emp\_id

**where** te.tda\_id **=** **any** **(select distinct** ve.ven\_tenda

**from** vendas **as** ve **join** detalle\_vendas **as** dv

**on** ve.ven\_id **=** dv.dev\_venda

**where** dv.dev\_artigo **=** '077WN45'**);**

* Tarefa 1.4

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

Mostrar a lista de artigos cun prezo de venta maior có prezo de venta do artigo

máis barato de cor negra, utilizando any

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

**select** **\***

**from** artigos

**where** art\_pv **>** **any** **(select** art\_pv

**from** artigos

**where** art\_color**=**"NEGRO"**);**

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

Mostrar a lista de artigos cun prezo de venta maior có prezo de venta do artigo

máis barato de cor negra, utilizando a función min().

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

**select** **\***

**from** artigos

**where** art\_pv **>** **(select** **min(**art\_pv**)**

**from** artigos

**where** art\_color**=**"NEGRO"**);**

* Tarefa 1.5

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

Seleccionar código de artigo, descrición do artigo e código do provedor para os artigos

que subministra o provedor SEAGATE

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

**select** art\_codigo **as** Artigo**,**

art\_nome **as** Descripción**,**

art\_provedor **as** 'Código de provedor'

**from** artigos

**where** art\_provedor **=** **( select** prv\_id

**from** provedores

**where** prv\_nome **=** 'SEAGATE'**);**

/\* outra forma, combinando as táboas \*/

**select** art\_codigo**,** art\_nome**,** art\_provedor

**from** artigos **join** provedores **on** art\_provedor**=**prv\_id

**where** prv\_nome**=**"SEAGATE"**;**

* Tarefa 1.6

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

Mostrar información do artigo máis caro, Seleccionando o seu código, nome, prezo de

venda e nome do provedor que o subministra.

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

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

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

ar.art\_pv **as** Prezo\_venda**,**

pr.prv\_nome **as** Proveedor

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

**where** ar.art\_pv **=** **(select** **max(**art\_pv**)** **from** artigos**);**

* Tarefa 1.7

/\*Mostrar nomes e salarios dos empregados que teñan salario maior có salario de

tódolos empregados do departamento 122. O resultado móstrase ordenado por nome de

empregado.\*/

**select** empNome **as** Nome**,**

empSalario **as** Salario

**from** empregado

**where** empSalario **>** **(select** **max(**empSalario**)**

**from** empregado

**where** empDepartamento**=**122**)**

**order** **by** empNome**;**

/\*Outra forma\*/

**select** empNome empregadoado**,**

empSalario **as** Salario

**from** empregado

**where** empSalario **>** **all** **(select** empSalario

**from** empregado

**where** empDepartamento**=**122**)**

**order** **by** empNome**;**

* Tarefa 1.8

/\*Mostrar os nomes e salarios dos empregados con salarios maiores ou iguais a o

de Claudia Fierro, ordenados alfabeticamente.\*/

#Con subconsulta

**select** empNome **as** Eempregado**,**

empSalario **as** Salario

**from** empregado

**where** empSalario **>=**

**(select** empSalario

**from** empregado

**where** empNome **=** 'FIERRO, CLAUDIA'**)**

**order** **by** empNome**;**

#Sen subconsulta utilizando join para facer unha autocombinación

**select** emp1**.**empNome **as** Eempregado**,**

emp1**.**empSalario **as** Salario

**from** empregado **as** emp1 **join** empregado **as** emp2

**on** emp1**.**empSalario **>=** emp2**.**empSalario

**where** emp2**.**empNome **=** 'FIERRO, CLAUDIA'

**order** **by** emp1**.**empNome**;**

* Tarefa 1.9

/\*Mostrar nome e presuposto dos departamentos que teñen o presuposto máis alto e o

máis baixo.\*/

**select** depNome **as** Departamento**,**

depPresuposto **as** Presuposto

**from** departamento

**where** depPresuposto **=** **(select** **min(**depPresuposto**)** **from** departamento**)**

**or** depPresuposto **=** **(select** **max(**depPresuposto**)** **from** departamento**);**

* Tarefa 1.10

/\*Mostrar número, nome, data de ingreso na empresa e nome do departamento no que

traballa o empregado ou empregados que levan máis tempo na empresa.\*/

**select** em.empNumero **as** Numero**,**

em.empNome **as** Nome**,**

em.empDataIngreso **as** **Data,**

de.depNome **as** Departamento

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

**where** em.empDataIngreso **=** **(select** **min(**empDataIngreso**)** **from** empregado**);**

* Tarefa 1.11

/\*Mostrar nome e salario dos empregados que cumpran algunha das seguintes condicións:

– Ingresaron na empresa despois do 1-1-88.

– Ingresaron na empresa antes do 1-1-88 pero teñen un salario inferior ao salario medio

de todos os empregados da empresa.\*/

**select** empNome **as** Empregado**,**

empSalario **as** Salario

**from** empregado

**where** empDataIngreso **>**'1988/01/01'

**or** **(** empDataIngreso **<=**'1988/01/01'

**and** empSalario **< (select** **avg(**empSalario**)** **from** empregado**))**

**order** **by** empNome**;**

* Tarefa 1.12

/\*Mostrar a extensión telefónica asignada a máis empregados indicando o número de

empregados que a comparten. Se hai máis dunha, deben aparecer todas.\*/

**select** empExtension **as** Extension**,**

**count(\*)** **as** Empregados

**from** empregado

**group** **by** empExtension

**having** **count(\*)=(select** **count(\*)**

**from** empregado

**group** **by** empExtension

**order** **by** 1 **desc limit** 1**);**

* + 1. Tarefa 2. Realizar consultas que utilicen o predicado BETWEEN combinado con subconsultas dentro dunha condición

A tarefa consiste en realizar as seguintes consultas utilizando o predicado BETWEEN para relacionar unha consulta principal con subconsultas dentro dunha condición.

Sobre a base de datos tendaBD

* Tarefa 2.1. Mostrar nome e prezo de venda dos artigos que teñen un prezo de venda comprendido entre o prezo do artigo '0713242' e a media de prezos de todos os artigos. Os datos deben mostrarse ordenados alfabeticamente polo nome do artigo.

Sobre a base de datos traballadores

* Tarefa 2.2. Seleccionar o nome e número de departamento dos empregados que pertenzan a un departamento cun presuposto comprendido entre os presupostos dos departamentos 122 e 121 (incluídos). Os datos deben mostrarse ordenados de menor a maior polo número do departamento.

Solución

* Tarefa 2.1

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

Mostrar nome e prezo de venda dos artigos que teñen un prezo de venda comprendido

entre o prezo do artigo '0713242' e a media de prezos de todos os artigos.

Os datos deben mostrarse ordenados alfabeticamente polo nome do artigo.

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

**select** art\_nome **as** Artigo**,**

art\_pv **as** Prezo

**from** artigos

**where** art\_pv **between** **(select** art\_pv **from** artigos **where** art\_codigo**=**'0713242'**)**

**and** **(select** **avg(**art\_pv**)** **from** artigos**)**

**order** **by** art\_nome**;**

* Tarefa 2.2

/\* Seleccionar o nome e número de departamento dos empregados que pertenzan a un

departamento cun presuposto comprendido entre os presupostos dos departamentos 122

e 121 (incluídos). Os datos deben mostrarse ordenados de menor a maior polo número

do departamento\*/

**select** empNome**,** empDepartamento

**from** empregado

**where** empDepartamento **= any** **(select** depNumero

**from** departamento

**where** depPresuposto **between** **(select** depPresuposto

**from** departamento

**where** depNUmero **=** 122**)**

**and** **(select** depPresuposto

**from** departamento

**where** depNUmero **=** 121**))**

**order** **by** empDepartamento **desc;**

/\* Para mostrar, ademais, algunha columna da táboa departamento hai que facer un join.

Exemplo: mostrando, ademais, o nome do departamento\*/

**select** empNome**,** empDepartamento**,** depNome

**from** empregado **join** departamento **on** empDepartamento **=** depNumero

**where** depPresuposto

**between** **(select** depPresuposto

**from** departamento

**where** depNUmero **=** 122**)**

**and** **(select** depPresuposto

**from** departamento

**where** depNUmero **=** 121**)**

**order** **by** empDepartamento **desc;**

* + 1. Tarefa 3. Realizar consultas utilizando subconsultas co predicado IN

A tarefa consiste en realizar as seguintes consultas utilizando o predicado IN para relacionar unha consulta principal con subconsultas dentro dunha condición.

Sobre a base de datos tendaBD

* Tarefa 3.1. Obter a lista de provedores que subministran como mínimo un artigo de cor negra.
* Tarefa 3.2. Mostrar identificador de cliente, apelidos e nome na mesma columna separados por coma, para os clientes que só teñen unha venda. O resultado estará ordenado polo identificador do cliente.
* Tarefa 3.3. Mostrar identificador e nome dos clientes que fixeron algunha compra despois do día en que o cliente número 6 fixo a súa última compra.
* Tarefa 3.4. Mostrar os nomes dos xerentes das tendas nas que se fixo algunha venta.
* Tarefa 3.5. Importe total das vendas que se fixeron ao cliente LEANDRO FERREIRO BENITEZ.
* Tarefa 3.6. Seleccionar o id, apelidos e nome dos empregados que aínda non fixeron ningunha venda.

Sobre a base de datos traballadores

* Tarefa 3.7. Mostrar o número de departamento e o número de empregados dos departamentos que teñen un presuposto anual superior a 36000 euros.
* Tarefa 3.8. Mostrar nome de departamento e de empregado para os empregados que traballan nalgún departamento que dependa do centro 'SEDE CENTRAL'. Os datos mostraranse ordenados por departamento e nome de empregado.
* Tarefa 3.9. Mostrar número de empregados e suma dos salarios, comisións e fillos, para os departamentos nos que existe algún empregado cun salario base mensual maior de 2000 euros.
* Tarefa 3.10. Mostrar o nome de todos os directores de departamento ordenados polo número de departamento. Resolver a consulta:
* con subconsultas.
* sen subconsultas.
* Tarefa 3.11. Mostrar os nomes dos directores de departamentos que dependen dun centro de traballo que ten un nome que empeza pola letra 'S'. Resolver a consulta:
* con subconsultas.
* sen subconsultas.
* Tarefa 3.12. A empresa decide gratificar aos directores en funcións incrementando o seu salario base un 5%. Mostrar ordenados alfabeticamente, os nomes destes empregados, o seu salario, a gratificación que lle corresponde, e o salario final que resulta de sumarlle a nova gratificación ao salario. Resolver a consulta:
* con subconsultas.
* sen subconsultas.
* Tarefa 3.13. Mostrar nome e salario dos empregados co salario base maior cá media dos soldos dos directores que están en funcións.
* Tarefa 3.14. Mostrar nome e salario+comisións dos empregados do centro “RELACIÓN CON CLIENTES ” que gañan máis de 1500 euros entre salario e comisións, ordenados por departamento, salario+comisión e nome.
* Tarefa 3.15. Mostrar os nomes dos empregados que traballan no mesmo departamento que Lavinia Sanz ou Cesar Pons.

Solución

* Tarefa 3.1

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

Obter a lista de provedores que subministran como mínimo un artigo de cor negra

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

**select** **\***

**from** provedores

**where** prv\_id **in** **(select distinct** art\_provedor

**from** artigos

**where** art\_color **=** "NEGRO"**)** **;**

* Tarefa 3.2

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

Mostrar identificador de cliente, apelidos e nome na mesma columna separados por coma,

para os clientes que só teñen unha venda. O resultado estará

ordenado polo identificador do cliente

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

**select** clt\_id **as** Cliente**,**

**concat(**clt\_apelidos**,**', '**,**clt\_nome**)** **as** 'Apelidos e nome'

**from** clientes

**where** clt\_id **in** **(select** ven\_cliente

**from** vendas

**group** **by** ven\_cliente

**having** **count(\*)** **=** 1**)**

**order** **by** clt\_id**;**

/\* outra forma alternativa utilizando referencias externas na subconsulta\*/

**select** clt\_id **as** Cliente**,**

**concat(**clt\_apelidos**,**', '**,**clt\_nome**)** **as** 'Apelidos e nome'

**from** clientes

**where** 1 **=** **(select** **count(\*)**

**from** vendas

**where** ven\_cliente **=** clt\_id**)**

**order** **by** clt\_id**;**

* Tarefa 3.3

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

Mostrar identificador e nome dos clientes que fixeron algunha compra despois do día

en que o cliente número 6 fixo a súa última compra.

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

**select** clt\_id**,** clt\_nome

**from** clientes

**where** clt\_id **in** **(** **select** ven\_cliente

**from** vendas

**where** ven\_data**>(select** **max(**ven\_data**)**

**from** vendas

**where** ven\_cliente**=**6**));**

/\* Tamén valería a condición:

where ven\_data > ALL (SELECT ven\_data FROM vendas WHERE ven\_cliente=6) \*/

/\* Outra solución podería ser \*/

**select** **distinct** ve.ven\_cliente as Cliente**,**

cl.clt\_nome as Nome

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

**where** ve.ven\_data **>** **(** **select** **max(**ven\_data**)**

**from** vendas

**where** ven\_cliente**=**6 **);**

* Tarefa 3.4

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

Mostrar os nomes dos xerentes das tendas nas que se fixo algunha venta

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

**select** **count(**em.emp\_nome**)** **as** Xerente

**from** tendas **as** te **join** empregados **as** em **on** te.tda\_xerente **=** em.emp\_id

**where** te.tda\_id **in** **(select** **distinct** ven\_tenda **from** vendas**);**

* Tarefa 3.5

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

Importe total das vendas que se fixeron ao cliente LEANDRO FERREIRO BENITEZ

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

**select** **sum((**dv.dev\_cantidade**\***dv.dev\_prezo\_unitario**)\*(**1-dv.dev\_desconto**/**100**))**

**as** Importe\_total

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

**where** ve.ven\_cliente **in** **(select** clt\_id

**from** clientes

**where** clt\_apelidos**=**'FERREIRO BENITEZ' **and** clt\_nome**=**'LEANDRO'**);**

* Tarefa 3.6

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

Seleccionar o id, apelidos e nome dos empregados que aínda non fixeron ningunha venda.

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

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

**from** empregados

**where** emp\_id **not** **in** **(select** **distinct** ven\_empregado **from** vendas**);**

* Tarefa 3.7

/\*Mostrar o número de departamento e o número de empregados dos departamentos que teñen

un presuposto anual superior a 36000 euros.\*/

**select** empDepartamento **as** Departamento**,**

**count(\*)** **as** Empregados

**from** empregado

**where** empDepartamento **in** **(select** depNumero

**from** departamento

**where** depPresuposto**>**36000**)**

**group** **by** empDepartamento**;**

#sen utilizar subsentencias

**select** em.empDepartamento **as** Departamento**,**

**count(\*)** **as** Empregados

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

**where** de.depPresuposto**>**36000

**group** **by** em.empDepartamento**;**

* Tarefa 3.8

/\*Mostrar nome de departamento e de empregado para os empregados que traballan nalgún

departamento que dependa do centro 'SEDE CENTRAL'. Os datos mostraranse ordenados por

nome de departamento e nome de empregado.\*/

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

em.empNome **as** Empregado

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

**where** de.depCentro **in** **(select** cenNumero

**from** Centro

**where** cenNome **like** '%SEDE CENTRAL%'**)**

**order** **by** de.depNome**,** em.empNome**;**

* Tarefa 3.9

/\*Mostrar número de departamento, número de empregados e suma dos salarios, comisións e

fillos, para os departamentos nos que existe algún empregado cun salario base mensual

maior de 2000 euros.\*/

**select** empDepartamento **as** Departamento**,**

**count(\*)** **as** Empregados**,**

**sum(**empSalario**)** **as** Salarios**,**

**sum(ifnull(**empComision**,**0**))** **as** Comisions**,**

**sum(**empFillos**)** **as** Fillos

**from** empregado

**where** empDepartamento **in** **(select** **distinct** empDepartamento

**from** empregado

**where** empSalario**>**2000**)**

**group** **by** empDepartamento**;**

* Tarefa 3.10

/\*Mostrar o nome de todos os directores de departamento ordenados polo número de

departamento.\*/

#Con subconsultas

**select** empDepartamento **as** Departamento**,**

empNome **as** Director

**from** empregado

**where** empNumero **in** **(select** depDirector

**from** departamento**)**

**order** **by** empDepartamento**;**

#Sen subconsultas

**select** em.empDepartamento **as** Departamento**,**

em.empNome **as** Director

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

**order** **by** em.empDepartamento**;**

* Tarefa 3.11

/\*Mostrar os nomes dos directores de departamentos que dependen dun centro de traballo

que ten un nome que empeza pola letra 'S'.\*/

#Con subconsultas

**select** empDepartamento **as** Departamento**,**

empNome **as** Director

**from** empregado

**where** empNumero **in** **(select** depDirector

**from** departamento

**where** depCentro **in (select** cenNumero

**from** centro

**where** cenNome **like** 'S%'**));**

#Sen subconsultas

**select** em.empDepartamento **as** Departamento**,**

em.empNome **as** Director

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

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

**where** ce.cenNome **like** 'S%'**;**

* Tarefa 3.12

/\*A empresa decide gratificar aos directores en funcións incrementando o seu salario

base un 5%. Mostrar ordenados alfabeticamente, os nomes destes empregados, o seu

salario, a gratificación que lle corresponde, e o salario final que resulta de

sumarlle a nova gratificación ao salario.\*/

#Con subconsultas

**select** empNome **as** Empregado**,**

empSalario **as** Salario\_base**,**

empSalario**\***0.05 **as** Gratificacion**,**

empSalario**+**empSalario**\***0.05 **as** Salario\_base\_final

**from** empregado

**where** empNumero **in** **(select** depDirector

**from** departamento

**where** deptipoDirector**=**'F'**)**

**order** **by** empNome**;**

#Sen subconsultas

**select** em.empNome **as** Empregado**,**

em.empSalario **as** Salario\_base**,**

em.empSalario**\***0.05 **as** Gratificacion**,**

em.empSalario**+**em.empSalario**\***0.05 **as** Salario\_base\_final

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

**where** de.deptipoDirector**=**'F'

**order** **by** em.empNome**;**

* Tarefa 3.13

/\*Mostrar nome e salario dos empregados co salario base maior cá media dos soldos

dos directores que están en funcións.\*/

**select** empNome **as** Empregado**,**

empSalario **as** Salario

**from** empregado

**where** empSalario **>** **(select** **avg(**empSalario**)**

**from** empregado

**where** empNumero **in** **(select** depDirector

**from** departamento

**where** deptipoDirector**=**'F'**));**

* Tarefa 3.14

/\*Mostrar nome e salario+comisións dos empregados do centro 'RELACIÓN CON CLIENTES'

que gañan máis de 1500 euros entre salario e comisións.

\*/

**select** empNome **as** Empregado**,**

**(**empSalario**+ifnull(**empComision**,**0**))** **as** Salario

**from** empregado **join** departamento **on** empDepartamento**=**depNumero

**where** **(**empSalario**+ifnull(**empComision**,**0**)** **>**1500**)**

**and** depCentro **= (select** cenNumero

**from** centro

**where** cenNome**=**'RELACION CON CLIENTES'**);**

* Tarefa 3.15

/\*Mostrar os nomes dos empregados que traballan no mesmo departamento que Lavinia Sanz

ou Cesar Pons. \*/

**select** empDepartamento **as** Departamento**,**

empNome **as** Empregado

**from** empregado

**where** empDepartamento **in** **(select distinct** empDepartamento

**from** empregado

**where** empNome **in** **(**'PONS, CESAR'**,** 'SANZ, LAVINIA'**));**

* + 1. Tarefa 4. Realizar consultas utilizando subconsultas co predicado EXISTS

A tarefa consiste en realizar as seguintes consultas utilizando o predicado EXISTS para relacionar unha consulta principal con subconsultas dentro dunha condición.

Sobre a base de datos tendaBD

* Tarefa 4.1. Mostrar a lista de artigos cun prezo de venta maior có prezo de venta do artigo máis barato de cor negra utilizando unha consulta de existencia.
* Tarefa 4.2. Mostrar o nome dos artigos de cor negra que teñan algunha venda de máis de 5 unidades. Obter o mesmo resultado nos dous casos seguintes:
* sen utilizar *join*.
* utilizando *join*.
* Tarefa 4.3. Mostrar nome e apelidos dos clientes que non fixeron ningunha compra.

Solución

* Tarefa 4.1

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

Mostrar a lista de artigos cun prezo de venta maior có prezo de venta do artigo

máis barato de cor negra, utilizando unha consulta de existencia

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

**select** **\***

**from** artigos **as** ar1

**where** **exists** **(select** art\_color

**from** artigos **as** ar2

**where** art\_color**=**"negro" **and** ar1**.**art\_pv**>**ar2**.**art\_pv**);**

* Tarefa 4.2

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

Mostrar o nome dos artigos de cor negra que teñan algunha venta de máis de 5 unidades

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

#Sen utilizar join

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

art\_nome **as** Articulo**,**

art\_color **as** Color

**from** artigos

**where** art\_color **=** "NEGRO"

**and** **exists** **(select** dev\_artigo

**from** detalle\_vendas

**where** dev\_artigo **=** art\_codigo **and** dev\_cantidade **>** 5**);**

#Utilizando join

**select** **distinct** ar.art\_codigo **as** Codigo**,**

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

ar.art\_color **as** Color

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

**where** ar.art\_color**=**"NEGRO" **and** dv.dev\_cantidade **>** 5**;**

* Tarefa 4.3

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

Mostrar nome e apelidos dos clientes que non fixeron ningunha compra.

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

**select** clt\_nome **as** Nome**,**

clt\_apelidos **as** Apelidos

**from** clientes

**where** **not** **exists** **(select** **\*** **from** vendas **where** ven\_cliente**=**clt\_id**);**

/\*outra forma alternativa\*/

**select** clt\_nome **as** Nome**,**

clt\_apelidos **as** Apelidos

**from** clientes

**where** clt\_id **not** **in** **(** **select** ven\_cliente **from** vendas **);**

* + 1. Tarefa 5. Realizar consultas utilizando subconsultas que non forman parte dunha condición

A tarefa consiste en realizar as seguintes consultas utilizando subconsultas que non formen parte dunha condición.

Sobre a base de datos tendaBD

* Tarefa 5.1. Mostrar as vendas diarias para cada tenda. A información que se debe mostrar é: código e poboación da tenda, data das vendas, suma dos importes das vendas na tenda nesa data con dous decimais, e acumulado dos importes das vendas feitas pola tenda ata esa data, con dous decimais.

Sobre a base de datos traballadores

* Tarefa 5.2. Seleccionar o número de departamento, a media dos salarios do departamento, e o salario medio de todos os empregados da empresa, para os departamentos que teñen un salario medio maior que o salario medio de todos os empregados da empresa.
* Tarefa 5.3. Mostrar o nome dos empregados que non son directores, e a diferenza do salario base respecto ao do empregado que menos cobra. Ordenar o resultado de forma descendente pola diferenza e ascendente polo nome.

Solución

* Tarefa 5.1

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

Mostrar información resumo das vendas para cada tenda.

A información que se debe mostrar é:

Código da tenda, poboación da tenda, data de cada venda (só a data sen hora),

suma dos importes das vendas na data, con 2 decimais

(cantidade vendida por prezo unitario menos desconto) e

acumulado dos importes das vendas feitas na tenda ata esa data, 2 decimais.

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

**select** te**.**tda\_id **as** Tenda**,**

te**.**tda\_poboacion **as** Poboacion**,**

**date(**ve1**.**ven\_data**)** **as** **Data,**

**round(sum(**dv1**.**dev\_prezo\_unitario**\***dv1**.**dev\_cantidade**\*(**1-dv1**.**dev\_desconto**/**100**)),**2**)**

**as** Vendas**,**

**(select** **round(sum(**dv2**.**dev\_prezo\_unitario**\***dv2**.**dev\_cantidade**\***

**(**1-dv2**.**dev\_desconto**/**100**)),**2**)**

**from** detalle\_vendas **as** dv2 **join** vendas **as** ve2

**on** dv2**.**dev\_venda **=** ve2**.**ven\_id

**where** ve1**.**ven\_tenda **=** ve2**.**ven\_tenda **and** **date(**ve2**.**ven\_data**)<=date(**ve1**.**ven\_data**))**

**as** Acumulado

**from** detalle\_vendas **as** dv1 **join** vendas **as** ve1 **on** dv1**.**dev\_venda **=** ve1**.**ven\_id

**join** tendas **as** te **on** te**.**tda\_id **=** ve1**.**ven\_tenda

**group** **by** te**.**tda\_id**,** **date(**ve1**.**ven\_data**);**

* Tarefa 5.2

/\*Seleccionar o número de departamento, a media dos salarios do departamento, e o

salario medio de todos os empregados da empresa, para os departamentos que teñen un

salario medio maior que o salario medio de todos os empregados da empresa. \*/

**select** empDepartamento **as** Departamento**,**

**avg(**empSalario**)** **as** Media\_departamento**,**

**(select** **avg(**empSalario**)from** empregado**)** **as** Media\_empregados

**from** empregado

**group** **by** empDepartamento

**having** **avg(**empSalario**)>(select** **avg(**empSalario**)from** empregado**);**

* Tarefa 5.3

/\*Mostrar o nome dos empregados que non son directores, e a diferenza do soldo respecto

ao empregado que menos cobra. Ordenar o resultado de forma descendente pola diferenza e

ascendente polo

nome.\*/

**select** empNome **as** Empregado**,**

empSalario**-(select** **min(**empSalario**)** **from** empregado**)** **as** Diferenza

**from** empregado

**where** empNumero **not** **in** **(select** depDirector

**from** departamento

**where**  depDirector **is not null)**

**order** **by** Diferenza **desc,** Empregado **desc;**

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\_V000504\_ UD05\_A04\_Subconsultas\_Anexos que contén:

* O arquivo GuiaWorkbench.docx que é unha guía básica de MySQL Workbench 6.3.
* O arquivo V000504\_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 |
| * CA5.7 Realizáronse consultas con subconsultas. | * Exame en papel e en formato dixital. | * **TO.6** - Documento de rexistro de realización de consultas que conteñen consultas. Escala de valores (observación indirecta). | 18 |

Modelo de proba para TO.6

Proponse unha proba práctica para avaliar CA5.7 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 sentencias.

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*

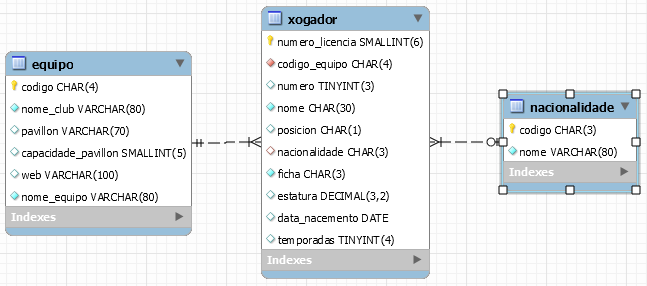
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| codigo | char(4) | non | Primaria | Código do equipo. Para enlazar con outras táboas |
| nome\_club | varchar(80) | non |  | Nome do club ao que pertence o equipo |
| pavillon | varchar(70) |  |  | 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 *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 |

* Táboa *nacionalidade*

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



Proba para CA5.7 sobre consulta con subconsulta

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

* 1.- Mostrar nome e idade do xogador máis novo. Pode haber máis de un.
* 2.- Mostrar código e nome dos equipos que teñen máis de 6 xogadores españois ('ESP').
* 3.- Mostrar os nomes dos xogadores que teñan a mesma nacionalidade co xogador que ten como nome: Hervelle, Axel. No resultado non pode saír este xogador.
* 4.- Mostrar os nomes dos equipos que non teñan ningún xogador coa nacionalidade de Estados Unidos de América ('USA').
* 5.- Mostrar o código e nome do equipo que ten o xogador máis alto da liga ACB.

Exemplo de solución para entregar en papel

Proponse a seguinte solución:

* 1.- Mostrar nome e idade do xogador máis novo.

/\* Mostrar nome e idade do xogador máis novo \*/

**select** nome **as** Nome**,**

timestampdiff**(year,** data\_nacemento**,**curdate**())** **as** Idade

**from** xogador

**where** data\_nacemento **=** **(select** **max(**data\_nacemento**)** **from** xogador**);**



* 2.- Mostrar código e nome dos equipos que teñen máis de 6 xogadores españois ('ESP').

/\* Mostrar código e nome dos equipos que teñen máis de 6 xogadores españois ('ESP')\*/

**select** codigo **as** Codigo**,**

nome\_equipo **as** Equipo

**from** equipo

**where** codigo **in** **(select** codigo\_equipo

**from** xogador

**where** nacionalidade **=**'ESP'

**group** **by** codigo\_equipo

**having** **count(\*)** **>** 6**);**



* 3.- Mostrar os nomes dos xogadores que teñan a mesma nacionalidade co xogador que ten como nome: Hervelle, Axel. No resultado non pode saír este xogador.

/\* Mostrar os nomes dos xogadores que teñan a mesma nacionalidade co xogador

que ten como nome: Hervelle, Axel. No resultado non pode saír este xogador \*/

**select** nome **as** Xogador**,**

nacionalidade **as** Nacionalidade

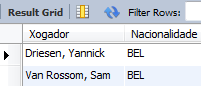
**from** xogador

**where** nacionalidade **=** **(select** nacionalidade

**from** xogador

**where** nome **=** 'Hervelle, Axel'**)**

**and** **not** nome **=** 'Hervelle, Axel'**;**



* 4.- Mostrar os nomes dos equipos que non teñan ningún xogador coa nacionalidade de Estados Unidos de América ('USA').

/\* Mostrar os nomes dos equipos que non teñan ningún xogador coa

nacionalidade de Estados Unidos de América ('USA') \*/

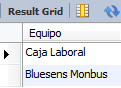
**select** nome\_equipo **as** Equipo

**from** equipo

**where** codigo **not** **in** **(select** **distinct** codigo\_equipo

**from** xogador

**where** nacionalidade **=** 'usa'**);**



5.- Mostrar o código e nome do equipo que ten o xogador máis alto da liga ACB.

/\* Mostrar o código e nome do equipo que ten o xogador máis alto da liga ACB \*/

**select** codigo **as** Código**,**

nome\_equipo **as** Equipo

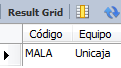
**from** equipo

**where** codigo **in** **(select** codigo\_equipo

**from** xogador

**where** estatura **=** **(select** **max(**estatura**)**

**from** xogador**));**



Exemplo de lista de valoración para TO.6

Proponse a seguinte lista de valoración para o instrumento de avaliación TO.6 (EV) (observación indirecta: documento de rexistro de realización de consultas que conteñen consultas).

| Nome | Data | |
| --- | --- | --- |
| Indicadores para  CA5.7 Realizáronse consultas con subconsultas | Valoración máxima  18 | Cualificación |
| As sentenzas de consulta non teñen erros sintácticos e cumpren as condicións pedidas. | 16 |  |
| As sentenzas de consulta contidas no arquivo .sql correspondent,e 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)