Dirección Xeral de Educación, Formación Profesional e Innovación Educativa

Material para

a formación profesional inicial

A03. Consultas resumo con agrupamentos

|  |  |  |
| --- | --- | --- |
| 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** | **A03** | **Consultas resumo con agrupamentos** |
| 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\_V000503\_UD05\_A03\_Agrupamentos.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 4](#_Toc439855070)

[Contexto da actividade 4](#_Toc439855071)

[Título da actividade 5](#_Toc439855072)

[Resultados de aprendizaxe do currículo 5](#_Toc439855073)

[Obxectivos didácticos e título e descrición da actividade 5](#_Toc439855074)

[Criterios de avaliación 5](#_Toc439855075)

[Contidos 5](#_Toc439855076)

[Actividades de ensino e aprendizaxe e de avaliación, métodos, recursos e instrumentos de avaliación 6](#_Toc439855077)

[2. A03. Consultas resumo con agrupamentos 7](#_Toc439855078)

[2.1 Introdución 7](#_Toc439855079)

[2.1.1 Obxectivos 7](#_Toc439855080)

[2.1.2 Software 7](#_Toc439855081)

[2.1.3 Bases de datos de traballo 8](#_Toc439855082)

[2.1.3.1 Base de datos tendaBD 8](#_Toc439855083)

[2.1.3.2 Base de datos practicas5 10](#_Toc439855084)

[2.1.3.3 Base de datos traballadores 11](#_Toc439855085)

[2.2 Actividade 12](#_Toc439855086)

[2.2.1 Consultas con agrupamento de filas 12](#_Toc439855087)

[2.2.2 Cláusula GROUP BY 12](#_Toc439855088)

[2.2.2.1 Agrupamento por máis dunha columna 15](#_Toc439855089)

[2.2.2.2 Modificador WITH ROLLUP 15](#_Toc439855090)

[2.2.3 Cláusula HAVING 17](#_Toc439855091)

[2.3 Tarefas 18](#_Toc439855092)

[2.3.1 Tarefa 1. Realizar consultas agrupando as filas en táboas resumo 18](#_Toc439855093)

[Solución 19](#_Toc439855094)

[2.3.2 Tarefa 2. Realizar consultas establecendo condicións para os grupos da táboa resumo 21](#_Toc439855095)

[Solución 21](#_Toc439855096)

[3. Materiais 23](#_Toc439855097)

[3.1 Documentos de apoio ou referencia 23](#_Toc439855098)

[3.2 Recursos didácticos 23](#_Toc439855099)

[3.3 Material auxiliar 23](#_Toc439855100)

[4. Avaliación 24](#_Toc439855101)

[Criterios de avaliación 24](#_Toc439855102)

[Modelo de proba para TO.4 24](#_Toc439855103)

[Proba para CA5.5 sobre consulta con agrupamentos 25](#_Toc439855104)

[Exemplo de solución para entregar en papel 25](#_Toc439855105)

[Exemplo de lista de valoración para TO.4 26](#_Toc439855106)

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 |
| A03 | Consultas resumo con agrupamentos | Realizáronse consultas mostrando información resumo coas funcións de agrupamento e establecendo condicións para os grupos. | 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 |
| --- | --- | --- | --- | --- | --- |
| O3.1 | Realizar consultas agrupando filas do conxunto de resultados e facer cálculos coas función de agrupamento. | A03 | Consultas resumo con agrupamentos | Realizáronse consultas mostrando información resumo coas funcións de agrupamento e establecendo condicións para os grupos. | 7 |
| O3.2 | Realizar consultas establecendo condicións para os grupos. |

Criterios de avaliación

|  |
| --- |
| Criterios de avaliación |
| * CA5.5 - Realizáronse consultas que xeren valores de resumo. |

Contidos

| Contidos |
| --- |
| * Consultas de resumo. Agrupamento de rexistros. |

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 |
| A03. Consultas resumo con agrupamentos  Realizáronse consultas mostrando información resumo coas funcións de agrupamento e establecendo condicións para os grupos. | * **Tp3.1** Exposición de conceptos sobre sentenzas LMD (Linguaxe de manipulación de datos) que permiten realizar consultas con agrupamentos. | **Ta3.1** 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 |
| * **Tp3.2** Demostración práctica de consultas que mostran información resumo utilizando funcións de agrupamento e establecendo condicións para os grupos. | * **Ta3.2** Realización de consultas que mostran información resumo utilizando funcións de agrupamento e establecendo condicións para os grupos. | * Guións de sentenzas LMD que solucionan os exercicios de consulta propostos. |  | 5.75 |
|  | * **Ta3.3** Tarefa de avaliación utilizando o instrumento de avaliación TO.4. | * Exame en papel e en formato dixital. | * Máquina virtual para exame que terá instalado MySQL e MySQL Workbench. * Manual de referencia de MySQL. | * **TO.4** - Documento de rexistro de realización de consultas utilizando funcións de agrupamento e xerando valores de resumo. Escala de valores (observación indirecta). | 0.25 |

1. A03. Consultas resumo con agrupamentos
   1. Introdución
      1. Obxectivos

Os obxectivos desta actividade son:

* Realizar consultas agrupando filas do conxunto de resultados e facer cálculos coas función de agrupamento.
* Realizar consultas establecendo condicións para os grupos.
  + 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.

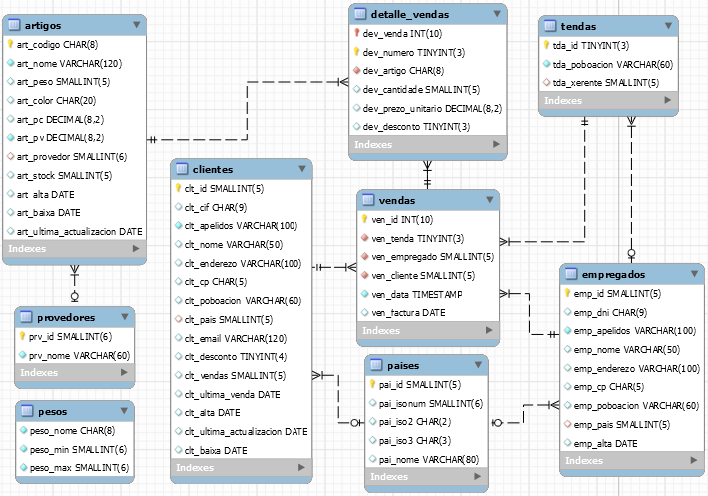
Normalmente, para a proba das consultas realizadas nesta actividade, mostrarase a zona de manipulación de datos de Workbench, coas filas que forman a táboa de resultados. Para completar a anterior información ou cando a consulta non pode mostrarse enteira xa que devolve moitas filas, mostrarase ademais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

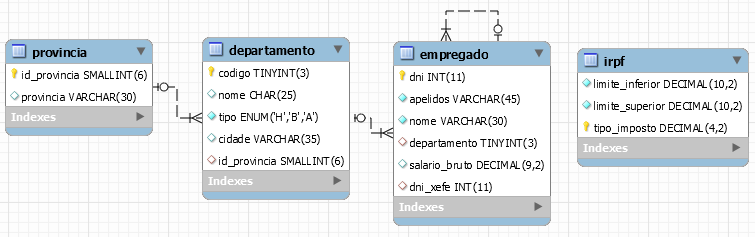
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| ven\_id | int unsigned | Non | Primaria | Identificador da venda. Numeraranse de 1 en adiante de forma automática. |
| ven\_tenda | tinyint unsigned | Non | Foránea | Identificador da tenda na que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

Chámase grupo á información correspondente a un conxunto de filas que teñen o mesmo valor nunha ou varias columnas dunha táboa denominadas columnas de agrupamento.

Existe un tipo de consultas con agrupamento de filas que dá como resultado unha táboa resumo que ten como columnas, as columnas de agrupamento e os cálculos, e ten unha fila por cada valor distinto que toman as columnas de agrupamento. Os cálculos realízanse coas funcións de agrupamento ou agregado que actúan sobre as filas de cada grupo en lugar de actuar sobre todas as filas seleccionadas como se fixo ata este momento.

A cláusula GROUP BY utilízase para facer consultas con agrupamento de filas.

* + 1. Cláusula GROUP BY

Permite especificar as columnas de agrupamento. Sintaxe:

[GROUP BY {nome\_columna | expresión | posición} [ASC|DESC], ... [WITH ROLLUP]]

* As columnas de agrupamento pódense representar por unha lista dun ou máis nomes de columnas, expresións, ou ben o número de orde que ocupa a columna dentro da lista de selección, igual que na cláusula ORDER BY. Non é recomendable utilizar o número de orde da columna porque no caso de cambiar a orde das columnas na lista de selección, ou engadir ou eliminar algunha columna, pódense producir resultados inesperados se non se cambia a cláusula GROUP BY.
* O modificador WITH ROLLUP, permite engadir filas na táboa resumo. Estas filas corresponden a cálculos feitos creando un grupo novo que contén todas as filas, e creando grupos a varios niveis no caso de agrupar por máis dunha columna.

En MySQL, cando se utiliza esta cláusula, ordénanse as filas polas columnas de agrupamento como se fose unha cláusula ORDER BY, antes de formar os grupos. As opcións [ASC | DESC] son unha extensión de MySQL ao estándar, e indican a orde na que se ordenan as filas antes de formar os grupos (ASC = ascendente, DESC = descendente); se non se indica nada, tómase ASC como valor por defecto.

Exemplo: calcular o número de departamentos que hai en cada provincia tendo en conta os datos contidos na táboa *departamento* da base de datos *practicas5*.

**select** id\_provincia**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia**;**

O SXBDR selecciona todas as filas da táboa *departamento* e as ordena pola columna *id\_provincia,* que é a columna de agrupamento, para que estean xuntos todos os departamentos que pertencen á mesma provincia; despois vai collendo fila a fila e cada vez que cambia o valor da columna *id\_provincia* prodúcese un ruptura e fórmase un novo grupo. Esta explicación do funcionamento ten unha finalidade didáctica, pero a forma en que se executa internamente a sentenza a decide o optimizador de consultas, e pode variar dun xestor a outro.

Representación gráfica do resultado da execución:

u503_exemplo1GROUPBY.emf

Na parte esquerda da imaxe móstranse as filas e columnas que forman a táboa *departamento*, remarcando as filas que teñen o mesmo valor na columna *id\_provincia.* Cada un dos marcos representa un grupo formado por todas as filas que teñen o mesmo valor na columna de agrupamento.

Na parte dereita da imaxe, móstrase a táboa resumo resultante que ten unha fila por cada grupo. As columnas que ten a táboa resumo son, en primeiro lugar, a columna de agrupamento (*id\_provincia*) para mostrar o valor que identifica ás filas de cada grupo, e en segundo lugar, os cálculos que se queren facer coas filas do grupo, utilizando neste caso, a función de agregado COUNT(\*) que conta o número de filas do grupo. Pódese observar que non ten sentido poñer calquera outra columna da táboa *departamento* na táboa resumo porque pode que teña valores distintos en distintas filas do grupo e o valor que se mostraría na columna sería inconsistente.

Algunhas consideracións a ter en conta cando se utiliza a cláusula GROUP BY:

* Para indicar as columnas de agrupamento, pódense utilizar os alias de columna.
* Pode haber diferenzas no tratamento dos valores NULL á hora de formar grupos. Algúns SXBDR consideran que non se agrupan as filas que conteñan o valor NULL na columna de agrupamento e outros consideran que todas as filas que teñan o valor NULL na columna de agrupamento forman un grupo. MySQL utiliza a segunda opción.
* O estándar SQL establece que na lista de selección da cláusula SELECT só se pode facer referencia ás columnas polas que se agrupa, funcións de agregado, constantes, ou expresións que combinen ás anteriores. Na maioría dos SXBDR, no caso de incluír calquera outra columna daría lugar a un erro. MySQL estende o uso de GROUP BY permitindo facer referencia na lista de selección de columnas distintas das columnas de agrupamento, e deixa en mans do usuario controlar que as columnas tomen un valor único para cada grupo. Pódese cambiar este comportamento, e desactivar as extensións de MySQL para GROUP BY, habilitando o modo SQL 'ONLY\_FULL\_GROUP\_BY'.
* Para cada grupo (cada valor distinto que tomen as columnas de agrupamento) créase unha fila na táboa resumo. Todos os elementos da lista de selección deben ter un valor único para cada grupo.

Os seguintes exemplos mostran dúas situacións diferentes producidas ao incluír na selección columnas distintas das columnas de agrupamento.

* Exemplo incluíndo a columna *cidade.*

**select** id\_provincia**,** cidade **,** **count(\*)**

**from** departamento

**group** **by** id\_provincia**;**

Representación gráfica do resultado da execución:

u503_exemplo5GROUPBY.emf

O contido da columna *cidade* na táboa *resumo* é inconsistente. A columna pode tomar máis dun valor para cada grupo. Na táboa resumo móstrase na columna un valor correspondente a unha fila do grupo, pero non representa unha información resumo do grupo para os casos de grupos que teñen máis dunha fila. Por exemplo, a primeira fila da táboa resumo parece que nos indica que na cidade de Ferrol hai dous departamentos, cando na cidade só hai un departamento, o 2 corresponde ao número de departamentos que hai na provincia que ten o *id\_provincia* (columna de agrupamento) 15.

MySQL non mostra ningunha mensaxe de erro no momento de executar a consulta pero a sentenza non é correcta e os datos son inconsistentes. A maioría dos SXBDR, ao executar esta sentenza mostrarían un erro indicando que na lista de selección hai unha columna que non está en GROUP BY.

* Exemplo incluíndo a columna *provincia* que contén o nome da provincia.

**select** id\_provincia**,** provincia**,** **count(\*)**

**from** departamento **natural** **join** provincia

**group** **by** id\_provincia**;**

Representación gráfica do resultado da execución:

u503_exemplo4GROUPBY.emf

A columna *provincia* da táboa resumo toma valores consistentes, porque para cada valor de *id\_provincia* (columna de agrupamento) só hai un valor posible para a columna *provincia*. Neste caso para cada grupo a columna *provincia* só toma un valor independentemente do número de filas que teña o grupo, polo que a sentenza é correcta.

* + - 1. Agrupamento por máis dunha columna

É posible agrupar por máis dunha columna, escribindo a lista das columnas polas que se quere agrupar, separadas por comas na cláusula GROUP BY.

Exemplo: contar o número de departamentos que hai en cada cidade.

* Solución 1 agrupando só pola columna *cidade*:

**select** id\_provincia**,** cidade**,** **count(\*)**

**from** departamento

**group** **by** cidade**;**

Representación gráfica do resultado da execución:

u503_exemplo3GROUPBY.emf

Créase unha única fila para a cidade 'Villalba' na táboa resumo, sen ter en conta que o mesmo nome de cidade se repite en dúas provincias distintas (27 e 28). Esta solución non sería correcta porque mostra que na cidade 'Villalba' da provincia 27 hai dous departamentos cando só hai un; non ten en conta que se trata de cidades diferentes.

* Solución 2 agrupando polas columnas *id\_provincia* e *cidade*.

**select** id\_provincia**,** cidade**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia**,** cidade**;**

Representación gráfica do resultado da execución:

u503_exemplo2GROUPBY.emf

Esta solución sería a correcta porque fai primeiro o agrupamento por provincias, e dentro de cada provincia, o agrupamento por cidade. Neste caso, a cidade 'Villalba' móstrase dúas veces, unha asociada ao id\_provincia 27 e outra ao id\_provincia 28.

* + - 1. Modificador WITH ROLLUP

Permite mostrar novas filas na táboa de resumo. Estas filas representan operacións de resumo para distintos niveis de análises. Resumo do comportamento do modificador nun servidor MySQL:

* No caso de agrupar só por unha columna, o modificador engade unha nova fila na táboa resumo na que se mostran os cálculos para un novo grupo formado por todas filas seleccionadas. Nesta fila móstrase o valor NULL na columna de agrupamento.

Exemplo: contar o número de departamentos que hai en cada provincia, e o número total de departamentos da empresa, tendo en conta os datos contidos na táboa *departamento* da base de datos *practicas5*.

/\* agrupar por unha columna co modificador WITH ROLLUP\*/

**select** id\_provincia**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia **with** **rollup;**

u503_exemploROLLUP1.emf

No resultado da execución pódese ver que ademais de mostrar unha fila por cada grupo, se engade unha fila máis co cálculo feito considerando un novo grupo formado por todas as filas da táboa *departamento*.

* No caso de agrupar por máis dunha columna, o modificador engade unha nova fila na táboa resumo cada vez que hai un cambio de valor nalgunha das columnas de agrupamento, ademais da fila correspondente ao cálculo para todas as filas.

Exemplo: contar o número de departamentos que hai en cada cidade, os que hai por cada provincia e o número total de departamentos da empresa.

/\* agrupar por máis dunha columna co modificador WITH ROLLUP\*/

**select** id\_provincia**,** cidade**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia**,** cidade **with** **rollup;**

Resultado a execución:

u503_exemploROLLUP2.emf

No resultado da execución pódese ver que ademais de mostrar unha fila por cada grupo, se engade unha fila máis por cada provincia, que é a primeira columna da lista de columnas de agrupamento, e ao final engade unha fila máis co cálculo feito considerando un novo grupo formado por todas as filas da táboa *departamento*.

Algunhas consideracións sobre o uso de ROLLUP en MySQL:

* Cando se usa este modificador non se pode utilizar a cláusula ORDER BY.
* Utilizar a cláusula LIMIT co modificador pode producir resultados difíciles de interpretar, porque a cláusula aplícase despois do modificador así que no límite cóntanse as filas extra engadidas polo modificador.

O resultado obtido cando se agrupa por unha soa columna e se utiliza ROLLUP é o mesmo que utilizando unha UNION de consultas.

/\* Agrupar por unha columna co modificador WITH ROLLUP\*/

**select** id\_provincia**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia **with** **rollup;**

/\* Alternativa utilizando union\*/

**select** id\_provincia**,** **count(\*)**

**from** departamento

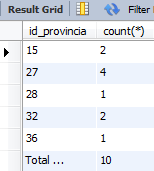
**group** **by** id\_provincia

**union**

**select** 'Total ...'**,** **count(\*)**

**from** departamento**;**

Ambas opcións obteñen o mesmo resultado:



* Tarefa 1. Realizar consultas agrupando as filas en táboas resumo.
  + 1. Cláusula HAVING

Esta cláusula está asociada coa cláusula GROUP BY. Permite establecer condicións para descartar aqueles grupos que non cumpran esas condicións. É similar á cláusula WHERE, pero a diferenza entre elas é que mentres WHERE analiza a condición nas filas das táboas de orixe da consulta (as especificadas na cláusula FROM), a cláusula HAVING o fai sobre a táboa resumo, é dicir, sobre os grupos que se formaron despois de agrupar.

Exemplo: contar os departamentos que hai en cada cidade, e mostrar só as cidades que teñen máis dun departamento.

**select** id\_provincia**,** cidade**,** **count(\*)**

**from** departamento

**group** **by** id\_provincia**,** cidade

**having count(\*) > 1;**

Representación gráfica do resultado da execución:

u503_exemploHAVING.emf

Ao aplicar a condición establecida ha cláusula HAVING, elimináronse da táboa resumo os grupos que non cumpren a condición.

Algunhas consideracións a ter en conta cando se utiliza a cláusula HAVING:

* O estándar ANSI SQL establece que as condicións desta cláusula só poden referirse ás columnas de agrupamento (as que figuran en GROUP BY), ou ás funcións de agregado. A partir da versión 5.0.2 MySQL estende este comportamento, e admite utilizar nas condicións do HAVING outras columnas diferentes das relacionadas en GROUP BY, e subconsultas. Pódese cambiar este comportamento, e desactivar estas extensións de MySQL para GROUP BY, habilitando o modo SQL 'ONLY\_FULL\_GROUP\_BY'.
* Non se debe utilizar HAVING para condicións que deban estar na cláusula WHERE.
* Pódense utilizar os alias para escribir as condicións.
* Tarefa 2. Realizar consultas establecendo condicións para os grupos da táboa resumo.
  1. Tarefas

As tarefas propostas son as seguintes:

* Tarefa 1. Realizar consultas agrupando as filas en táboas resumo.
* Tarefa 2. Realizar consultas establecendo condicións para os grupos da táboa resumo.
  + 1. Tarefa 1. Realizar consultas agrupando as filas en táboas resumo

A tarefa consiste en realizar as seguintes consultas agrupando filas.

Sobre a base de datos tendaBD

* Tarefa 1.1. Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor (con dous decimais).
* Tarefa 1.2. Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor (con dous decimais), excluíndo aos artigos que teñan un prezo de compra superior a 50 euros.
* Tarefa 1.3. Mostrar as cidades nas que existen clientes e o número de clientes que hai en cada unha de elas. Clasificar a saída en orden decrecente polo número de clientes.
* Tarefa 1.4. Mostrar código, nome, suma dos importes das vendas sen aplicar o desconto, suma dos importes das vendas despois de aplicar o desconto, e desconto efectuado, para os artigos vendidos entre 1-1-2015 e 25-5-2015.
* Tarefa 1.5. Mostrar as estatísticas de vendas do ano 2015 por tenda. A información a mostrar é: identificador da tenda, número de vendas, número de artigos vendidos, suma de unidades vendidas e a media dos prezos unitarios dos artigos vendidos.

Sobre a base de datos traballadores

* Tarefa 1.6. Mostrar o número de empregados que utiliza cada extensión telefónica.
* Tarefa 1.7. Mostrar o número de empregados que teñen 0, 1, 2, 3, ... fillos.
* Tarefa 1.8. Mostrar, para cada departamento, o número de empregados que teñen 0, 1, 2, ... fillos.
* Tarefa 1.9. Mostrar o número de departamentos que dependen de cada centro.
* Tarefa 1.10. Seleccionar, para cada departamento, o maior salario, o menor salario e a diferenza que hai entre o salario máis alto e o salario máis baixo.
* Tarefa 1.11. Mostrar, para cada departamento, a cantidade que queda do presuposto despois de restar o importe dos salarios e comisións a pagar aos empregados.

Solución

* Tarefa 1.1

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

Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor

(con dous decimais).

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

**select** art\_color**,**

**round(avg(**art\_pv**),**2**)**

**from** artigos

**where** art\_color **is** **not** **null**

**group** **by** art\_color**;**

* Tarefa 1.2

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

Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor

(con dous decimais), excluíndo aos artigos que teñan un prezo de compra superior a 50

euros

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

**select** art\_color **as** Color**,**

**avg(**art\_pv**)** **as** Prezo\_medio

**from** artigos

**where** art\_color **is** **not** **null** **and** art\_pc **<=** 50

**group** **by** art\_color**;**

* Tarefa 1.3

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

Mostrar as cidades nas que existen clientes, e o número de clientes que hai en cada

unha de elas. Clasificar a saída en orden decrecente polo número de clientes

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

**select** clt\_poboacion **as** Poblacion**,**

**count(\*)** **as** Numero\_clientes

**from** clientes

**group** **by** clt\_poboacion

**order** **by** Numero\_clientes **desc;**

* Tarefa 1.4

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

Mostrar código, nome, suma dos importes das vendas sen aplicar o desconto, suma dos

importes das vendas despois de aplicar o desconto, e desconto efectuado, para os

artigos vendidos entre 1-1-2015 e 25-5-2015.

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

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

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

**sum(**dv**.**dev\_prezo\_unitario**\***dv**.**dev\_cantidade**)** **as** Total**,**

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

**as** Cobrado**,**

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

**as** Desconto

**from** detalle\_vendas **as** dv

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

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

**where** **date(**ve**.**ven\_data**)** **between** "2015-01-01" **and** "2015-05-25"

**group** **by** ar**.**art\_codigo**,** ar**.**art\_nome**;**

* Tarefa 1.5

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

Mostrar as estatísticas de vendas do ano 2015 por tenda. A información a mostrar é:

identificador da tenda, número de vendas, número de artigos vendidos, suma de unidades

vendidas e a media dos prezos unitarios dos artigos vendidos

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

**select** ve.ven\_tenda **as** Tenda**,**

**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

**where** **year(ve.**ven\_data**)** **=** 2015

**group** **by** ve.ven\_tenda**;**

* Tarefa 1.6

/\*Mostrar o número de empregados que utiliza cada extensión telefónica\*/

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

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

**from** empregado

**group** **by** empExtension**;**

* Tarefa 1.7

/\*Mostrar o número de empregados que teñen 0, 1, 2, 3, ... fillos\*/

**select** empFillos **as** Fillos**,**

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

**from** empregado

**group** **by** empFillos**;**

* Tarefa 1.8

/\*Mostrar para cada departamento, o número de empregados que teñen 0, 1, 2, ... fillos, ordenados por número de departamento e número de fillos.\*/

**select** de.depNumero **as** Departamento**,**

em.empFillos **as** Fillos**,**

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

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

**group** **by** de.depNumero**,em.**empFillos**;**

* Tarefa 1.9

/\*Mostrar o número de departamentos que dependen de cada centro.\*/

**select** depCentro **as** Centro**,**

**count(\*)** **as** Departamentos

**from** departamento

**group** **by** depCentro**;**

# Solución mostrando o nome do centro

**select** ce.cenNome **as** Centro**,**

**count(\*)** **as** Departamentos

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

**group** **by** de.depCentro**,** ce.cenNome**;**

* Tarefa 1.10

/\* Seleccionar, para cada departamento, o maior salario, o menor salario e a diferenza

que hai entre o salario máis alto e o salario máis baixo.\*/

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

**max(**em.empSalario**)-** **min(**em.empSalario**)** **as** Diferenza**,**

**max(**em**.**empSalario**)** **as** Maior**,**

**min(**em.empSalario**)** **as** Menor

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

**group** **by** de.depNome**;**

* Tarefa 1.11

/\*Mostrar por departamento, a cantidade que queda do presuposto despois de restar o

importe dos salarios e comisións a pagar aos empregados.\*/

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

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

de**.**depPresuposto**-sum(**em**.**empSalario**)-ifnull(sum(**em**.**empComision**),**0**)** **as** Dispoñible

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

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

**group** **by** de**.**depNumero**;**

* + 1. Tarefa 2. Realizar consultas establecendo condicións para os grupos da táboa resumo

A tarefa consiste en realizar as seguintes consultas agrupando filas e establecendo condicións para os grupos.

Sobre a base de datos tendaBD

* Tarefa 2.1. Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor, para as cores que teñan o prezo medio maior que 100 euros.
* Tarefa 2.2. Mostrar as tendas que fixeron máis de 2 vendas no mes de maio de 2015. Para cada tenda débese mostrar: numero de tenda, número de vendas, número de artigos diferentes vendidos e a suma de unidades vendidas nese período de tempo.
* Tarefa 2.3Mostrar o identificador do cliente, data de venda, a cantidade de artigos vendidos, a suma dos importes das vendas na data e o desconto practicado nesas vendas, para os clientes aos que se vendeu máis de 1200 euros nun só día.
* Tarefa 2.4. Mostrar identificador de cliente, apelidos e nome na mesma columna separados por coma, e data e hora da venda, para os clientes que só teñen unha venda.

Sobre a base de datos traballadores

* Tarefa 2.5. Mostrar número e nome dos departamentos que teñan 5 empregados.
* Tarefa 2.6. Para as extensións telefónicas que son utilizadas por máis dun empregado, mostrar o número de empregados que a comparten.

Solución

* Tarefa 2.1

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

Seleccionar da táboa artigos as cores e o prezo medio de venda dos artigos de cada cor,

para as cores que teñan o prezo medio maior que 100 euros

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

**select** art\_color**,**

**round(avg(**art\_pv**),**2**)**

**from** artigos

**where** art\_color **is** **not** **null**

**group** **by** art\_color

**having** **avg(**art\_pv**)>**100**;**

/\* utilizando os alias en having \*/

**select** art\_color **as** Color**,**

**round(avg(**art\_pv**),**2**)** **as** Prezo\_medio

**from** artigos

**where** art\_color **is** **not** **null**

**group** **by** art\_color

**having** Prezo\_medio**>**100**;**

* Tarefa 2.2

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

Mostrar as tendas que fixeron máis de 2 vendas no mes de maio de 2015.

Para cada tenda débese mostrar: numero de tenda, número de vendas, número de

artigos vendidos diferentes e a suma de unidades vendidas nese período de tempo.

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

**select** ve.ven\_tenda **as** Tenda**,**

**count(\*)** **as** Numero\_vendas**,**

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

**sum(**dv.dev\_cantidade**)** **as** Numero\_unidades\_vendidas

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

**where** **month(date(**ve.ven\_data**))** **=** 5 **and** **year(date(**ve.ven\_data**))** **=** 2015

**group** **by** ve.ven\_tenda

**having** Numero\_vendas**>**2**;**

* Tarefa 2.3

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

Mostrar o identificador do cliente, data de venda, a cantidade de artigos vendidos,

a suma dos importes das vendas na data e o desconto practicado nesas vendas, para os

clientes aos que se vendeu máis de 1200 euros nun só día.

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

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

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

**sum(**dv**.**dev\_cantidade**)** **as** Cantidade**,**

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

**as** Importe**,**

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

**as** Desconto

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

**group** **by** ve**.**ven\_cliente**,** ve**.**ven\_data

**having** **sum((**dv**.**dev\_cantidade**\***dv**.**dev\_prezo\_unitario**)\*(**1-dv**.**dev\_desconto**/**100**))>=**1200**;**

* Tarefa 2.4

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

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

coma, e data e hora da venda, para os clientes que só teñen unha venda.

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

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

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

ve.ven\_data **as** **Data**

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

**group** **by** cl.clt\_id

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

* Tarefa 2.5

/\*Mostrar número e nome dos departamentos que teña 5 empregados.\*/

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

de.depNome **as** Nome

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

**group** **by** de.depNumero

**having** **count(\*)** **=** 5**;**

* Tarefa 2.6

/\* Para as extensións telefónicas que son utilizadas por máis dun empregado, mostrar o

número de empregados que a comparten.\*/

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

**count(\*)** **as** Comparten

**from** empregado

**group** **by** empExtension

**having** **count(\*)** **>** 1**;**

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\_V000503\_UD05\_A03\_Agrupamentos\_Anexos que contén:

* O arquivo GuiaWorkbench.docx que é unha guía básica de MySQL Workbench 6.3.
* O arquivo V000503\_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 |
| * CA5.5 - Realizáronse consultas que xeren valores de resumo. | * Exame en papel e en formato dixital. | * **TO.4** - Documento de rexistro de realización de consultas utilizando funcións de agrupamento e xerando valores de resumo. Escala de valores (observación indirecta). | 15 |

Modelo de proba para TO.4

Proponse unha proba práctica para avaliar CA5.5 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*

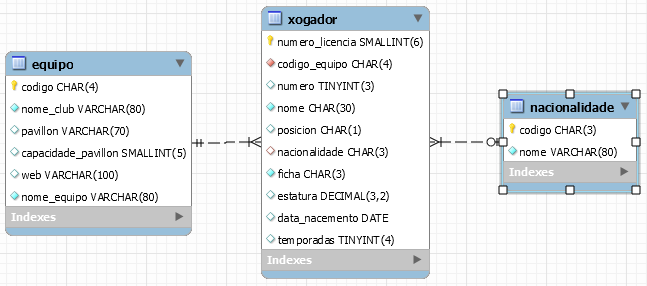
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome columna | Tipo | Null | Clave | Observacións |
| codigo | char(4) | non | Primaria | Código do equipo. Para enlazar con outras táboas |
| nome\_club | varchar(80) | non |  | Nome do club ao que pertence o equipo |
| pavillon | varchar(70) |  |  | 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 CA5.5 sobre consulta con agrupamentos

Escribir as sentenzas SQL que permitan realizar as seguintes consultas utilizando agrupamentos 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 código, nome do equipo e media das estaturas dos xogadores dos equipos que teñan os xogadores cunha estatura media superior aos 2 metros.
* 2.- Mostrar código de equipo e número de xogadores para os equipos que teñen máis de 6 xogadores, ordenado por número de xogadores.
* 3.- Seleccionar o nome dos países dos que hai xogadores, a idade do xogador con máis idade, e a estatura do xogador máis baixo de cada país

Exemplo de solución para entregar en papel

Proponse a seguinte solución:

* 1.- Mostrar código de equipo e número de xogadores españois (nacionalidade 'ESP') para os equipos que teñen máis de 6 xogadores españois. Ordenar o resultado polo número de xogadores, de maior a menor.

/\* Mostrar código de equipo e número de xogadores españois (nacionalidade 'ESP') para os

equipos que teñen máis de 6 xogadores españois. Ordenar o resultado polo número de

xogadores, de maior a menor \*/

**select** codigo\_equipo **as** Equipo**,**

**count(\*)** **as** Xogadores

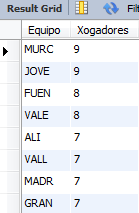
**from** xogador

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

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

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

**order** **by** Xogadores **desc;**



* 2.- Mostrar código, nome do equipo e media das estaturas dos xogadores dos equipos que teñan os xogadores cunha estatura media superior aos 2 metros.

/\* Mostrar código, nome do equipo e media das estaturas dos xogadores dos equipos que

teñan os xogadores cunha estatura media superior aos 2 metros \*/

**select** eq**.**codigo **as** Codigo**,**

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

**round(avg(**estatura**),**2**)** **as** Media\_estaturas

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

**group** **by** xo**.**codigo\_equipo

**having** **avg(**xo**.**estatura**)>=**2**;**



* 3.- Seleccionar o nome dos países dos que hai xogadores, a idade do xogador con máis idade, e a estatura do xogador máis baixo de cada país

/\*Seleccionar o nome dos países dos que hai xogadores, a idade do xogador con máis

idade, e estatura do xogador máis baixo de cada país\*/

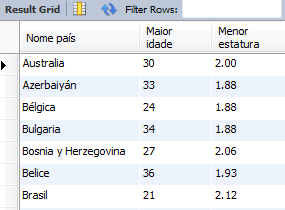
**select** na**.**nome **as** 'Nome país'**,**

**timestampdiff(year,**xo**.**data\_nacemento**,**curdate**())** **as** 'Maior idade'**,**

**min(**estatura**)** **as** 'Menor estatura'

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

**group** **by** na**.**codigo**;**





Exemplo de lista de valoración para TO.4

Proponse a seguinte lista de valoración para o instrumento de avaliación TO.4 (EV) (observación indirecta: documento de rexistro de realización de consultas utilizando funcións de agrupamento e xerando valores de resumo).

| Nome | Data | |
| --- | --- | --- |
| Indicadores para  CA5.5 - Realizáronse consultas que xeren valores de resumo. | Valoración máxima  15 | Cualificación |
| As sentenzas de consulta non teñen erros sintácticos e cumpren as condicións pedidas. | 13 |  |
| 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 |  |

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