SQL Summary – Grade 10. Version One. Exercise

Memo

(40)

SELECT <field(s)>

FROM <table(s)>

WHERE condition(s)

GROUP BY expression

    HAVING condition

ORDER BY expression

Arithmetic function: INT( ), ROUND( ), TRUNCATE( ). Formats the single parameter within the brackets. RND - a single random number.

Random numbers: RND(Upper bound – Lower bound) + Lower bound. Gives a random number from lower bound to upper bound eg 1 to 50.

Date and time: DATE, YEAR, MONTH, DAY, TIME, HOUR, MINUTE. Used with a field of the data type "date". NOW( ) yields today from the PC.

Aggregate functions: MAX( ), MIN( ), AVG( ), SUM( ), COUNT( ). Returns a single result based on a group of rows.

Compound conditions: NOT, AND, OR

More conditions: BETWEEN .. AND, IN and NOT IN, LIKE, NULL

Quotes. Regular quotes for string data. Hash symbols for date/time . #2019/05/23 9:33:00#. Boolean (True) – no quotes.

String manipulation: LEFT(x), RIGHT(x), MID(x, y), LEN, and "&" which is the concatenation operator in Ms Access (joins strings)

NOTE: Ms Access is case insensitive (unlike Java). We use upper case and lower case to make our queries more readable.

Using a table of Coffee types as an example . . . . tblCoffee

tblProducts

ProductID	ProductName	CountryOfOrigin	Roast	SizeInGrams	CostPrice	Clic
1	Gold	Ethiopia	Dark	250	R38.00	
2	Gold	Ethiopia	Dark	1000	R140.00	
3	Blue	Limu	Medium	250	R31.00	
4	Blue	Limu	Medium	500	R58.00	
5	Blue	Limu	Medium	1000	R114.00	
6	Blue	Limu	Medium	2000	R200.00	
7	Green	Kenya	Dark	500	R60.00	
8	Green	Kenya	Dark	1000	R112.00	
9	Red	Malawi	Light	250	R40.00	
10	Purple	Rwanda	Medium	250	R36.00	
11	Purple	Rwanda	Medium	1000	R108.00	
12	Purple	Rwanda	Medium	2000	R196.00	
13	Yellow	Kenya	Dark	250	R42.00	
14	Yellow	Kenya	Dark	500	R79.00	
15	Yellow	Kenya	Dark	1000	R150.00	
16	Yellow	Kenya	Dark	2000	R240.00	
(New)				0	R0.00	

SELECT . . . FROM . . .

1 Select all the fields and all the records from the table above

Select * From tblCoffee

2 All the fields and records sorted by country A to Z

Select * From tblCoffee Order By CountryOfOrigen

3 All the fields and records where the roast is dark.

Select * From tblCoffee Where Roast = "Dark"

2

THE BEST … THE 5 BEST … THE WORST … THE HIGHEST … THE MOST (EXPENSIVE) (CHEAPEST)

4 Find the coffee package with the lowest cost price   Select min(costPrice) As [min Cost] From tbl OR

Select Top 1 costPrice From tblCoffee Order By costPrice OR

5 Find the coffee package with the highest cost price   Select max(costPrice) As [max Cost] From tbl OR

Select Top1 costPrice From tblCoffee Order By costPrice Desc;

6 Write a query to give us the 5 coffee packages with the highest prices

Select Top 5 costPrice From tblCoffee OrderBy costPrice Desc;

SELECT ONLY CERTAIN FIELDS FROM THE TABLE.

7 Write a query to extract only the product name and roast type

Select productName, roast From tblProducts;

SELECT ONLY CERTAIN FIELDS and SORTING BY ONE FIELD.

8 Write a query to give us the product names sorted by country of origin

Select productName, CountryOfOrigin From tblCoffee Order By CountryOf Origin;

FIND ITEMS IN A GIVEN LIST WITHOUT USING COMPOUND OPERATORS (AND , OR)

10 Write a query that gives us only those coffees from Kenya and Rwanda

Select * From tblCoffee Where countryOfOrigin in ("Kenya", "Rwanda");

3

Using a table of players as an example . . .

ID	Full Name	Nationality	Gender	DOB	Professiona
1	Connie Rossetti	Greece	Female	17 Nov 2001	☐
2	Boycey Wingar	China	Male	15 Jan 1978	☑
3	Bert Laughrey	France	Female	22 Oct 1983	☐
4	Flossi Couroy	China	Female	14 Dec 1982	☑
5	Dian Ogborne	China	Female	30 Apr 1981	☑
6	Tonia Gilardi	Indonesia	Female	09 Sep 1973	☐
7	Consalve Lithcow	Russia	Male	04 Aug 1979	☑
8	Sonnie Pessolt	Indonesia	Female	15 Oct 1986	☐
9	Angelita Bertram	Indonesia	Female	14 Feb 1982	☐
10	Reuben Oxbrow	Peru	Male	18 Oct 2000	☐
11	Sayres Hansbury	Nigeria	Male	23 Nov 1974	☐

EXTRACT A PORTION OF A DATE - YEAR, MONTH, DAY FROM A FIELD THAT HAS THE DATA TYPE OF "Date"

" [ ] "brackets around Full Name because of the space in the middle of the field name

"AS" gives this extracted year value its own label ie "Year"

10 We need a list of all player names showing the year in which they were born. Create a label field called "Year"

Select [Full Name], Year(dob) As Year From tblPlayers

11 We need a list of all player names showing the month in which they were born. Create a label field called "Month"

Select [Full Name], month(dob) As Month From tblPlayers

12 We need a list of all player names showing the day on which they were born. Create a label field called "Day"

Select [Full Name] day(dob) As Day From tblPlayers

4

WE NEVER STORE DATA THAT WE CAN DETERMINE FROM OTHER FIELDS. Examples: ages, totals, averages, participation

DETERMINING AGE FROM DATE OF BIRTH AND SORTING FROM YOUNGEST TO OLDEST

13 Write a query to determine the ages of the players sorted from youngest to oldest.

Select [full Name], year (now()) - year (dob) As Age From tbl Orderby now() - dob;

EXTRACTING BIRTHDAYS THAT MATCH A CONDITION e.g. larger than January but smaller than June

14 Write a query to determine those players born in October, November and December.

Select [Full Name], dob From tbl Where month(dob) in (10, 11, 12) OR

Where month(dob) >=10 AND month(dob) <= 12;

COMPARING AGES TO A PARTICULAR CONDITION e.g. smaller and equal to 20

15 Write a query showing the players names who are older than 20 years.

Select [full Name] From tbl Where year (now()) - year(dob) >= 20

DETERMINGING THE AVERAGE AGE IN YEARS, FROM DATE OF BIRTH

16 Write a query to determine the average age of all the players.

Select avg (year (now()) - year(dob)) As [Ave Age] From tbl

DETERMINING THOSE NAMES THAT SO LONG THAT THEY WILL CREATE PROBLEMS WITH THE PRINTED MATERIAL.

17 We need to know the names of the players whose names are longer than 20 characters in length

Select [full Name] From tbl Where len([Full Name]) > 20;

Using a teacher's markbook as an example . . .

Note: We do not type in totals as they can be calculated from the figures already in the table. We say that they are derived fields i.e. derived from data already in the table.

tblStudents

ID	Maths	Eng	Science	IT	Art
1	46	63.5	78	50	87
2	98	56	87	95	45
3	45.5	76	87.5	87	87
4	87	67	87	67.5	95
5	93.5	59	54.5	62.5	

FINDING VALUES THAT ARE MISSING FROM OUR TABLE USING NULL – WE USE "IS NULL" OR "IS NOT NULL".

20 Write a query to make sure that every student has a mark in every field.

Select * from tbl where maths is null OR eng is null OR science is null OR IT is null OR art is null

CALCULATING TOTALS AND AVERAGES FROM GIVEN SCORES – note comma after the wildcard

21 Write a query to give the student's average mark for the five subjects.

Select id, (maths + eng + science + it + art)/5 As Average from tbl

EXTRACTING THOSE LEARNERS WHOSE AVERAGE IS HIGHER THAN 70%

22 Write a query to return all those learners who have an average mark greater than 70%;

Select id from tbl where ((maths + eng + science + it + art)/5) > 70