**Segment A. Mathematics Test**

1. Which of the following states contributed approximately 25% of the combined production of all the 4 country in the year 2002?

**Answer: a. Indonesia**

Total: 22+19+26+11+13= 91

Indonesia: 22/91 = **24.18%**

Singapore: 19/91 = 20.88%

Malaysia: 26/91 = 28.57%

Thailand: 11/91 = 12.1%

Philippines: 13/91 = 14.29%

2. What was the difference in volumes exported in 1997 and 1998?

**Answer: d. 10,000,000 kg**

1997 = 150,000,000 kg

1998 = 160,000,000 kg

Difference = 160-150 = **10,000,000 kg**

1. In which year was the value per kg the least?

1995: 150/100 = **1.5**

1996: 150/75 = 2.0

1997: 330/150 = 2.2

1998: 400/160 = 2.5

1999: 500/200 = 2.5

**Answer: a. 1995**

1. Which of the following statements is/are true regarding the consumption of chocobar?  
   (in ‘000 bars)

1993 to 1994: 118-124 = -6,000

1994 to 1995: 128-118 = 10,000

**1995 to 1996: 92-128 = -36,000**

**1996 to 1997: 134 -92 = 42,000**

1997 to 1998: 126-134 = -8,000

1998 to 1999: 122 – 126 = -4,000

**Answer: c. The steepest fall in the consumption of chocobar follows the steepest increase in consumption.**

5. Over the period 1993-94 to 1997-98, there has been a/an \_\_\_\_\_ in fertiliser subsidy.

Data from 93 to 98:

Indigenous fertiliser: 3000, 3400, 3350, 3300, 4800

Imported fertiliser: 200, 1142, 1039, 1499, 1000

**Answer: d. Inconsistent Expenditure.**

6. In which year was the bank credit per sick unit the maximum?

Bank credit/sick unit on each year ($ billions/’000s sick unit):

1996: 0.75

1997: 0.666

1998: 0.4167

1999: 0.3875

2000: 0.4

**Answer: a. 1996**

7. For strategy genre, in which year was the ratio of rejection to production the highest among the given years?

To find: ratio of **rejection** to **production**

1995: 0.06

1996: 0.0444

**1997: 0.0625**

1998: 0.05476

1999: 0.0521

2000: 0.05122

**Answer: c. 1997**

8. What is the ratio of the distribution of proteins in the muscles to that of proteins in the bones?

Distribution of proteins in muscles: 1/3

Distribution of proteins in the bones: 1/6

1/3:1/6 = 2:1

**Answer: b. 2:1**

**Segment B. SQL Query Test**

\*Note: My answer will be attached as “agate2.sql” but the overall input & output will be written here.

\*\*The output result is based on the dummy data provided.

1. Create an SQL query that shows the TOP 3 authors who sold the least books in total.

**Answer:**

**SELECT author\_name, SUM(sold\_copies) as total\_sold FROM authors a**

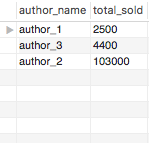
**LEFT JOIN books b on a.book\_name = b.book\_name**

**GROUP BY author\_name**

**ORDER BY total\_sold ASC**

**LIMIT 3;**

**Output:**

****

2. Write an SQL query to find out how many users inserted more than 2000 but less than 4000 images in their presentations!

**Answer:**

SELECT

COUNT(user\_id) AS no\_of\_unique\_users

FROM

(SELECT

user\_id, COUNT(event\_date\_time) AS times\_inserted

FROM

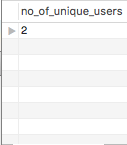
event\_log

GROUP BY user\_id

HAVING (COUNT(event\_date\_time) < 4000

AND COUNT(event\_date\_time) > 2000)) AS T;

**Output:**

****

3. Print every **department** where the **average salary** per **employee** is **over than** **500$!**

**Answer:**

1. To display the department\_name that fulfill the requirements (avg(salary) > 500) **only** and sorted by the average salary the query used will be**:**

**Query input a:**

SELECT

department\_name

FROM

(SELECT

department\_name,

CAST(AVG(salary) AS DECIMAL (10 , 2 )) AS average\_sal

FROM

employees e

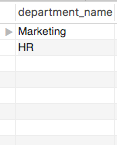
LEFT JOIN salaries s ON e.employee\_id = s.employee\_id

GROUP BY department\_name

HAVING (average\_sal > 500)) AS T

ORDER BY average\_sal;

**Output a:**

****

1. To display the department\_name and the average salary sorted by average salary the query used will be:

**Query input b:**

SELECT

department\_name,

CAST(AVG(salary) AS DECIMAL (10 , 2 )) AS average\_sal

FROM

employees e

LEFT JOIN

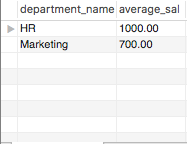
salaries s ON e.employee\_id = s.employee\_id

GROUP BY department\_name

HAVING (average\_sal > 500)

ORDER BY average\_sal DESC;

**Output b:**

****

1. Create SQL Query that **show Person Data** with each their **Deposito Amount**. Data sorted by **PERSON\_ID.**

**Answer**

**Query Input:**

SELECT

d.deposito\_id, d.amount, p.person\_id, p.name AS person\_name

FROM

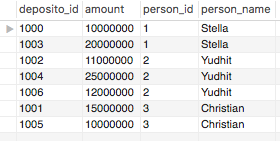
agate\_person p

LEFT JOIN

agate\_deposito d ON d.person\_id = p.person\_id

ORDER BY person\_id;

**Output:**

****