Задание:

1. Написать запрос, считающий суммарное количество имеющихся на сайте новостей и обзоров.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **SUM** |
| 7 |

**Ответ:**

SELECT

(select count(\*) from news)

+

(select count(\*) from reviews) AS 'SUM';

1. Написать запрос, показывающий список категорий новостей и количество новостей в каждой категории.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **nc\_name** | **COUNT(`n\_id`)** |
| Финансы | 1 |
| Законодательство | 0 |
| Логистика | 4 |
| Строительство | 0 |

**Ответ:**

SELECT nc\_name, count(n\_id)

FROM news\_categories

LEFT JOIN news

ON news.n\_category = news\_categories.nc\_id

GROUP BY nc\_name DESC;

1. Написать запрос, показывающий список категорий обзоров и количество обзоров в каждой категории.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **rc\_name** | **COUNT(`r\_id`)** |
| Технологии | 2 |
| Товары и услуги | 0 |

**Ответ:**

SELECT rc\_name, count(r\_id) FROM reviews\_categories

LEFT JOIN reviews

ON reviews.r\_category = reviews\_categories.rc\_id

GROUP BY rc\_name;

1. Написать запрос, показывающий список категорий новостей, категорий обзоров и дату самой свежей публикации в каждой категории.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **category\_name** | **last\_date** |
| Технологии | 2012-12-12 06:31:13 |
| Финансы | 2012-12-03 04:15:27 |
| Логистика | 2012-12-20 06:11:42 |

**Ответ:**

SELECT rc\_name, MAX(r\_dt) FROM reviews\_categories, reviews

WHERE reviews.r\_category = reviews\_categories.rc\_id

GROUP BY rc\_name

UNION

SELECT nc\_name, MAX(n\_dt) FROM news\_categories, news

WHERE news.n\_category = news\_categories.nc\_id

GROUP BY nc\_name DESC;

1. Написать запрос, показывающий список страниц сайта верхнего уровня (у таких страниц нет родительской страницы) и список баннеров для каждой такой страницы.

В результате выполнения запроса должно получиться:

|  |  |  |
| --- | --- | --- |
| **p\_name** | **b\_id** | **b\_url** |
| Юридическим лицам | 1 | http://tut.by |
| Юридическим лицам | 2 | http://tut.by |
| Юридическим лицам | 7 | http://habrahabr.ru |
| Физическим лицам | 1 | http://tut.by |

**Ответ:**

SELECT pages.p\_name, banners.b\_id, banners.b\_url

FROM pages

JOIN m2m\_banners\_pages ON m2m\_banners\_pages.p\_id = pages.p\_id

JOIN banners ON banners.b\_id = m2m\_banners\_pages.b\_id

where p\_parent IS NULL;

1. Написать запрос, показывающий список страниц сайта, на которых есть баннеры.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **p\_name** |
| Юридическим лицам |
| Физическим лицам |
| Образцы договоров |
| Банковские реквизиты |
| Схема проезда к офису |

**Ответ:**

SELECT DISTINCT pages.p\_name

FROM pages

JOIN m2m\_banners\_pages

ON m2m\_banners\_pages.p\_id = pages.p\_id;

1. Написать запрос, показывающий список страниц сайта, на которых нет баннеров.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **p\_name** |
| Почта и телефон |
| Договоры оптовых закупок |

**Ответ:**

SELECT pages.p\_name

FROM pages

WHERE pages.p\_id NOT IN

(SELECT p\_id FROM m2m\_banners\_pages);

1. Написать запрос, показывающий список баннеров, размещённых хотя бы на одной странице сайта.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **b\_id** | **b\_url** |
| 1 | http://tut.by |
| 2 | http://tut.by |
| 7 | http://habrahabr.ru |
| 4 | http://onliner.by |
| 3 | http://onliner.by |

**Ответ:**

SELECT DISTINCT banners.b\_id, banners.b\_url

FROM banners

JOIN m2m\_banners\_pages

ON banners.b\_id = m2m\_banners\_pages.b\_id

1. Написать запрос, показывающий список баннеров, не размещённых ни на одной странице сайта.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **b\_id** | **b\_url** |
| 5 | http://google.by |
| 6 | http://google.com |
| 8 | http://habrahabr.ru |
| 9 | http://gismeteo.by |
| 10 | http://gismeteo.ru |

**Ответ:**

SELECT DISTINCT banners.b\_id, banners.b\_url

FROM banners

WHERE banners.b\_id NOT IN

(SELECT m2m\_banners\_pages.b\_id FROM m2m\_banners\_pages);

1. Написать запрос, показывающий баннеры, для которых отношение кликов к показам >= 80% (при условии, что баннер был показан хотя бы один раз).

В результате выполнения запроса должно получиться:

|  |  |  |  |
| --- | --- | --- | --- |
| b\_id | b\_url | rate |  |
| 2 | http://tut.by | 150.0000 | Да-да, в исходных данных ошибка. Потому тут – 150%. |
| 3 | http://onliner.by | 90.0000 |  |
| 5 | http://google.by | 100.0000 |  |
| 6 | http://google.com | 100.0000 |  |
| 7 | http://habrahabr.ru | 99.7998 |  |
| 8 | http://habrahabr.ru | 98.0000 |  |

**Ответ:**

SELECT b\_id, b\_url, (b\_click / b\_show) \* 100 AS rate

FROM banners

WHERE (b\_click / b\_show) >= 0.8;

1. Написать запрос, показывающий список страниц сайта, на которых показаны баннеры с текстом (в поле `b\_text` не NULL).

В результате выполнения запроса должно получиться:

|  |
| --- |
| **p\_name** |
| Юридическим лицам |
| Физическим лицам |
| Банковские реквизиты |
| Схема проезда к офису |

**Ответ:**

SELECT DISTINCT p\_name

FROM pages

JOIN m2m\_banners\_pages ON pages.p\_id = m2m\_banners\_pages.p\_id

JOIN banners ON m2m\_banners\_pages.b\_id = banners.b\_id

WHERE banners.b\_text IS NOT NULL;

1. Написать запрос, показывающий список страниц сайта, на которых показаны баннеры с картинкой (в поле `b\_pic` не NULL).

В результате выполнения запроса должно получиться:

|  |
| --- |
| **p\_name** |
| Юридическим лицам |
| Образцы договоров |
| Банковские реквизиты |
| Схема проезда к офису |

**Ответ:**

SELECT DISTINCT p\_name

FROM pages

JOIN m2m\_banners\_pages ON pages.p\_id = m2m\_banners\_pages.p\_id

JOIN banners ON m2m\_banners\_pages.b\_id = banners.b\_id

WHERE banners.b\_text IS NOT NULL;

1. Написать запрос, показывающий список публикаций (новостей и обзоров) за 2011-й год.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **header** | **date** |
| Контрабанда железобетонных плит | 2011-09-14 06:19:08 |
| Почта России: вчера, сегодня и снова вчера | 2011-08-17 09:06:30 |
| Роботы на страже строек | 2011-10-03 05:17:37 |

**Ответ:**

SELECT n\_header AS header, n\_dt AS date

FROM news

WHERE year(n\_dt) = 2011

UNION

SELECT r\_header AS header, r\_dt AS date

FROM reviews

WHERE year(r\_dt) = 2011;

1. Написать запрос, показывающий список категорий публикаций (новостей и обзоров), в которых нет публикаций.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **category** |
| Законодательство |
| Строительство |
| Товары и услуги |

**Ответ:**

SELECT news\_categories.nc\_name AS category

FROM news\_categories

WHERE news\_categories.nc\_id NOT IN

(SELECT news.n\_category FROM news)

UNION

SELECT reviews\_categories.rc\_name AS category

FROM reviews\_categories

WHERE reviews\_categories.rc\_id NOT IN

(SELECT reviews.r\_category FROM reviews);

1. Написать запрос, показывающий список новостей из категории «Логистика» за 2012-й год.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **n\_header** | **n\_dt** |
| Самолётом или поездом? | 2012-12-20 06:11:42 |
| Куда всё катится? | 2012-12-11 04:36:17 |

**Ответ:**

SELECT DISTINCT news.n\_header, news.n\_dt

FROM news

JOIN news\_categories

ON news.n\_category = 3

WHERE year(n\_dt) = 2012;

1. Написать запрос, показывающий список годов, за которые есть новости, а также количество новостей за каждый из годов.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **year** | **COUNT(\*)** |
| 2011 | 2 |
| 2012 | 3 |

**Ответ:**

SELECT year(n\_dt) AS year, count(\*)

FROM news

GROUP BY year(n\_dt);

1. Написать запрос, показывающий URL и id таких баннеров, где для одного и того же URL есть несколько баннеров.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **b\_url** | **b\_id** |
| http://tut.by | 1 |
| http://tut.by | 2 |
| http://onliner.by | 3 |
| http://onliner.by | 4 |
| http://habrahabr.ru | 7 |
| http://habrahabr.ru | 8 |

**Ответ:**

SELECT b\_url, b\_id

FROM banners

WHERE b\_url IN (

SELECT b\_url FROM banners

GROUP BY b\_url

HAVING COUNT(b\_url) > 1);

1. Написать запрос, показывающий список непосредственных подстраниц страницы «Юридическим лицам» со списком баннеров этих подстраниц.

В результате выполнения запроса должно получиться:

|  |  |  |
| --- | --- | --- |
| **p\_name** | **b\_id** | **b\_url** |
| Образцы договоров | 4 | http://onliner.by |
| Банковские реквизиты | 1 | http://tut.by |
| Банковские реквизиты | 2 | http://tut.by |

**Ответ:**

SELECT pages.p\_name, banners.b\_id, banners.b\_url

FROM pages

JOIN m2m\_banners\_pages ON pages.p\_id = m2m\_banners\_pages.p\_id

JOIN banners ON m2m\_banners\_pages.b\_id = banners.b\_id

WHERE pages.p\_parent = 1;

1. Написать запрос, показывающий список всех баннеров с картинками (поле `b\_pic` не NULL), отсортированный по убыванию отношения кликов по баннеру к показам баннера.

В результате выполнения запроса должно получиться:

|  |  |  |  |
| --- | --- | --- | --- |
| **b\_id** | **b\_url** | **rate** |  |
| 2 | http://tut.by | 1.5000 | Да-да, в исходных данных ошибка. Потому тут – 150%. |
| 5 | http://google.by | 1.0000 |  |
| 6 | http://google.com | 1.0000 |  |
| 7 | http://habrahabr.ru | 0.9980 |  |
| 4 | http://onliner.by | 0.0200 |  |

**Ответ:**

SELECT DISTINCT banners.b\_id, banners.b\_url, (b\_click / b\_show) AS rate

FROM banners

WHERE banners.b\_pic IS NOT NULL

ORDER BY rate DESC;

1. Написать запрос, показывающий самую старую публикацию на сайте (не важно – новость это или обзор).

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **header** | **date** |
| Почта России: вчера, сегодня и снова вчера | 2011-08-17 09:06:30 |

**Ответ:**

SELECT news.n\_header AS header, news.n\_dt AS date

FROM news

UNION

SELECT reviews.r\_header AS header, reviews.r\_dt AS date

FROM reviews

ORDER BY date

LIMIT 1;

1. Написать запрос, показывающий список баннеров, URL которых встречается в таблице один раз.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **b\_url** | **b\_id** |
| http://google.by | 5 |
| http://google.com | 6 |
| http://gismeteo.by | 9 |
| http://gismeteo.ru | 10 |

**Ответ:**

SELECT b\_url, b\_id

FROM banners

WHERE b\_url IN (

SELECT b\_url FROM banners

GROUP BY b\_url

HAVING COUNT(b\_url) = 1);

1. Написать запрос, показывающий список страниц сайта в порядке убывания количества баннеров, расположенных на странице. Для случаев, когда на нескольких страницах расположено одинаковое количество баннеров, этот список страниц должен быть отсортирован по возрастанию имён страниц.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **p\_name** | **banners\_count** |
| Схема проезда к офису | 3 |
| Юридическим лицам | 3 |
| Банковские реквизиты | 2 |
| Образцы договоров | 1 |
| Физическим лицам | 1 |

**Ответ:**

SELECT pages.p\_name, COUNT(m2m\_banners\_pages.b\_id) AS banners\_count

FROM pages

JOIN m2m\_banners\_pages ON pages.p\_id = m2m\_banners\_pages.p\_id

GROUP BY pages.p\_name

ORDER BY banners\_count DESC, p\_name ASC;

1. Написать запрос, показывающий самую «свежую» новость и самый «свежий» обзор.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **header** | **date** |
| Самолётом или поездом? | 2012-12-20 06:11:42 |
| Когда всё это кончится?! | 2012-12-12 06:31:13 |

**Ответ:**

(SELECT n\_header, n\_dt

FROM news

ORDER BY n\_dt DESC

LIMIT 1)

UNION

(SELECT r\_header, r\_dt

FROM reviews

ORDER BY r\_dt DESC

LIMIT 1);

1. Написать запрос, показывающий баннеры, в тексте которых встречается часть URL, на который ссылается баннер.

В результате выполнения запроса должно получиться:

|  |  |  |
| --- | --- | --- |
| **b\_id** | **b\_url** | **b\_text** |
| 1 | http://tut.by | TUT.BY |
| 3 | http://onliner.by | ONLINER.BY |
| 5 | http://google.by | GOOGLE.BY |
| 8 | http://habrahabr.ru | HABRAHABR.RU |

Ответ:

SELECT DISTINCT b\_id, b\_url, b\_text

FROM banners

WHERE b\_text != '' AND b\_url LIKE CONCAT('%', b\_text, '%')

1. Написать запрос, показывающий страницу, на которой размещён баннер с самым высоким отношением кликов к показам.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **p\_name** |
| Юридическим лицам |

Ответ:

SELECT pages.p\_name

FROM pages

JOIN m2m\_banners\_pages USING (p\_id)

JOIN banners USING (b\_id)

HAVING MAX(banners.b\_click / banners.b\_show);

1. Написать запрос, считающий среднее отношение кликов к показам по всем баннерам, которые были показаны хотя бы один раз.

В результате выполнения запроса должно получиться:

|  |
| --- |
| **AVG(`b\_click`/`b\_show`)** |
| 0.81224975 |

Ответ:

SELECT AVG(b\_click/b\_show)

FROM banners

WHERE b\_show > 0;

1. Написать запрос, считающий среднее отношение кликов к показам по баннерам, у которых нет графической части (поле `b\_pic` равно NULL).

В результате выполнения запроса должно получиться:

|  |
| --- |
| **AVG(`b\_click`/`b\_show`)** |
| 0.66000000 |

Ответ:

SELECT AVG(b\_click/b\_show)

FROM banners

WHERE b\_show > 0 AND b\_pic IS NULL;

1. Написать запрос, показывающий количество баннеров, размещённых на страницах сайта верхнего уровня (у таких страниц нет родительских страниц).

В результате выполнения запроса должно получиться:

|  |
| --- |
| COUNT |
| 4 |

Ответ:

SELECT COUNT(b\_id) AS count

FROM pages

JOIN m2m\_banners\_pages USING(p\_id)

JOIN banners USING(b\_id)

WHERE pages.p\_parent IS NULL;

1. Написать запрос, показывающий баннер(ы), который(ые) показаны на самом большом количестве страниц.

В результате выполнения запроса должно получиться:

|  |  |  |
| --- | --- | --- |
| **b\_id** | **b\_url** | **COUNT** |
| 1 | http://tut.by | 4 |

Ответ:

SELECT banners.b\_id, banners.b\_url, COUNT(b\_id) AS COUNT

FROM banners

JOIN m2m\_banners\_pages USING(b\_id)

JOIN pages USING(p\_id)

GROUP BY b\_id

LIMIT 1;

1. Написать запрос, показывающий страницу(ы), на которой(ых) показано больше всего баннеров.

В результате выполнения запроса должно получиться:

|  |  |
| --- | --- |
| **p\_name** | **COUNT** |
| Юридическим лицам | 3 |
| Схема проезда к офису | 3 |