

САНКТ – ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ – ПРОЦЕССОВ УПРАВЛЕНИЯ

БАЗА ДАННЫХ “СОЗВЕЗДИЯ”

Жигалов Валентин Сергеевич

НАПРАВЛЕНИЕ 010400
ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА

ПРЕПОДАВАТЕЛЬ

ФИЛИППОВ Р.О.

1 ДЕКАБРЯ 2017 Г.

Структура базы данных

Consellations:

- Id;
- Latin_name;
- genitive_latin_name;
- shortening;
- quantity_of_visible_stars;
- brightest_star;
- square;
- id_area.

Founders:

- id;
- name;
- country;
- date_of_birth;
- date_of_death.

Area:

- id;
- name;
- semisphere;
- max_right_ascension;
- min_right_ascension;
- max_declination;
- min_declination.

Families:

- id;
- name;
- max_right_ascension;
- min_right_ascension;
- max_declination;
- min_declination.

Consellations__founders:

- id_consellation;
- id_founder;
- id.

Areas__families:

- id_area;
- id_family;
- id.

Схема базы данных

Constellations	
Id_созвездия	integer primary key
Latin_name	Varchar(30)
Genitive_latin_name	Varchar(35)
Shortening	Varchar(3)
Quantity_of_visible_stars	Integer
Brightest_star	Varchar(30)
square	integer
Id_area	Integer

area	
id	Integer primary key
name	Varchar(30)
semisphere	Varchar(5)
Max_right_ascension	Varchar
Min_right_ascension	Varchar
Max_declination	Varchar
Min_declination	varchar

Areas_families	
Id_area	Integer
Id_family	integer
id	Serial primary key

families	
Id	Integer primary key
name	Varchar(30)
Max_right_ascension	real
Min_rigth_ascension	real
Max_declination	real
Min_declinaion	real

Constellations_founders	
Id_constellation	integer
Id_founder	Integer
id	Serial primary key

founders	
id	Integer primary key
name	Varchar(30)
country	Varchar(15)
Date_of_birth	Integer
Date_of_death	Integer

Ссылка на репозиторий с БД: https://github.com/VZhigalov/stars_db

Описание базы данных

Данная БД представляет из себя список созвездий. В БД представлено 31 созвездие, признающееся Международным астрономическим союзом. Помимо созвездий, в таблице представлена информация об их первооткрывателях, а также семействах созвездий и областях звездного неба, в которых расположены те или иные созвездия.

Взаимоотношения между объектами

Звездное небо поделено на 8 областей (квадратов), по 4 в каждом полушарии. В каждой области расположено несколько созвездий, при этом одно созвездие не может находиться в нескольких областях звездного неба.

Семейства созвездий схожи с областями звездного неба. В них также входит несколько созвездий, при этом одно созвездие не может входить в несколько семейств. Но в отличие от областей, семейства возникали по мере открытия новых созвездий. Также созвездия из одних и тех же семейств могут быть в разных областях, и наоборот.

Что касается ученых, открывавших созвездия, то одно созвездие могло открываться несколькими учеными, если они работали вместе, или один продолжал дело другого.

Consellations:

В этой таблице представлены наиболее интересные характеристики созвездий.

Поля таблицы:

- **id primary key**

Для поля взят тип integer

- **latin_name**

(Латинское название). Названия созвездий не могут повторяться (уникальные), и есть ограничение на размер названия (не более 30 символов). Тип varchar(30)

- **genitive_latin_name**

(Латинское название в родительном падеже). Названия созвездий в родительном падеже используются в профессиональной среде для указания, в каком созвездии расположена звезда. Также не могут повторяться и имеют ограничение на размер. Тип varchar(35)

- **shortening**

(Сокращение). Сокращения названий созвездий также используются в профессиональной среде. Больше 3 букв в сокращении быть не может. Тип varchar(3)

- **quantity_of_visible_stars**

(Количество видимых звезд). Не все звезды из созвездий можно увидеть на Земле из-за большого расстояния, а точное число всех звезд созвездия достаточно сложно высчитать. Тип integer.

- **brightest_star**

(Самая яркая звезда). Тип данных varchar(30)

- **square**

(Площадь). Площадь созвездий представлена в квадратных градусах. Квадратный угол – часть пространства, которая является объединением всех лучей, выходящих из данной точки (центра Земли) и пересекающих некоторую поверхность (звездное небо), измеряется отношением площади части сферы с центром в точке, вырезаемой квадратным углом, к квадрату радиуса сферы. Тип integer.

- **id_area**

(id области). Показывает id области, в которой расположено созвездие. Тип integer.

Founders:

Здесь представлена информация о первооткрывателях созвездий.

- **id primary key**

Тип integer.

- **name**

(Имя). Имена ученых не могут повторяться. Тип varchar(30).

- **country**

(Страна). Страна, в которой ученый жил и работал. Тип varchar(15).

- **date_of_birth**

(Дата рождения). Год рождения ученого. Взят тип integer, так как у некоторых ученых неизвестна точная дата рождения.

- **date_of_death**

(Дата смерти). Год смерти ученого. Взят тип integer, так как у некоторых ученых неизвестна точная дата смерти.

Area:

Здесь представлена информация об областях звездного неба.

- **id primary key**

Взят тип данных integer

- **name**

(Название). Тип данных varchar(30)

- **semisphere**

(Полушарие). Поле показывает, в каком полушарии расположена данная область. Тип данных varchar(5)

- **max_right_ascension, min_right_ascension**

(Максимальное и минимальное прямое восхождение). Прямое восхождение – координата второй экваториальной системы координат, астрономический эквивалент земной долготы, но отсчитывается от точки весеннего равноденствия. Данные в таблице являются приближительными. Измеряется в часах, принимает значения от 0 до 24 не включительно. Тип данных real.

- **max_declination, min_declination**

(Максимальное и минимальное склонение). Склонение – координата второй экваториальной системы координат, астрономический эквивалент земной широты. Данные в таблице являются приблизительными. Измеряется в градусах, принимает значения от -90 до 90. Тип данных real.

Families:

Здесь представлена информация о семействах созвездий

- **id primary key**

Тип данных integer.

- **name**

(Название). Тип данных varchar(30)

- **max_right_ascension, min_right_ascension**

Аналогично областям

- **max_declination, min_declination**

Аналогично областям

Consellations__founders:

Таблица создана для реализации отношения «многие ко многим» между таблицами consellations и founders

- **Id_consellation** – id созвездий, тип integer
- **Id_founder** – id ученых, тип integer
- **Id primary key** – тип serial

Areas__families:

Таблица создана для реализации отношения «многие ко многим» между таблицами area и families

- **id_area** – id области, тип integer
- **id_founder** – id первооткрывателя, тип integer
- **id primary key** – тип serial

Легкие запросы

- 1) Вывести информацию о созвездиях, отсортировав их по площади
`SELECT * FROM consellations ORDER BY square;`
Оптимизация: `CREATE INDEX ON consellations(square);`
Создан индекс consellaions_square_idx для фильтрации созвездий по площади.
- 2) Вывести имена ученых родом из Дании
`SELECT name FROM founders WHERE country = 'Denmark';`
Оптимизация: `CREATE INDEX ON founders(country);`
Создан индекс founders_country_idx для фильтрации ученых по стране.
- 3) Сгруппировать строки таблицы «area» по полушариям
`SELECT semisphere FROM area GROUP BY semisphere;`
Оптимизация: `CREATE INDEX ON area(semisphere);`
Создан индекс area_semisphere_idx для фильтрации строк по полушариям.
- 4) Вывести названия семейств и диапазоны их прямых восхождений и склонений

```
SELECT name, (max_right_ascension – min_right_ascension) AS right_ascension_diam, (max_declination – min_declination) AS declination_diam FROM families;
```

Оптимизация: `CREATE INDEX ON families(max_right_ascension);`

Создан индекс `families_max_right_ascension_idx` для фильтрации строк по одной из координат.

Средние запросы

- 1) Вывести названия и площади созвездий и области, в которых они расположены

```
SELECT latin_name, square, name FROM consellations INNER JOIN area ON (id_area = area.id);
```

Оптимизация: `CREATE INDEX ON area(name);`

Создан индекс `area_name_idx` для оптимизации объединения таблиц.

- 2) Вывести id семейств и соответствующие названия областей звездного неба и отсортировать по id_family

```
SELECT area.name, id_family FROM areas__families LEFT OUTER JOIN area ON (area.id = id_area) ORDER BY id_family;
```

Оптимизация: `CREATE INDEX ON area(name);`

Создан индекс `area_name_idx` для оптимизации объединения таблиц.

- 3) Вывести id областей и соответствующие названия семейств созвездий

```
SELECT families.name, id_area FROM families RIGHT OUTER JOIN areas__families ON (families.id = id_family);
```

Оптимизация: `CREATE INDEX ON areas__families(id_area);`

Создан индекс `areas__families_id_area_idx` для оптимизации объединения таблиц.

Сложные запросы

- 1) Вывести области с количеством созвездий больше 2 и среднее арифметическое площадей созвездий области

```
SELECT name, avg_square FROM
```

```
(SELECT a.name, count(a.id) AS quantity, avg(c.square) AS avg_square
```

```
FROM consellations AS c
```

```
RIGHT JOIN area AS a
```

```
ON (a.id = c.id_area)
```

```
GROUP BY a.id
```

```
ORDER BY avg_square) AS areas
```

```
WHERE quantity > 2;
```

- 2) Вывести ученого, открывшего больше всего созвездий

```
SELECT name FROM
```

```
(SELECT f.name, count(f.id) AS quantity
```

```
FROM founders AS f
```

```
LEFT JOIN consellations__founders AS c_f
```

```
ON (f.id = c_f.id_founder)
```

```
GROUP BY f.id) AS counts
```

```
WHERE quantity = (SELECT max(quantity) FROM  
(SELECT f.name, count(f.id) AS quantity  
FROM founders AS f  
LEFT JOIN consellations__founders AS c_f  
ON (f.id = c_f.id_founder)  
GROUP BY f.id) AS counts);
```

- 3) Вывести названия областей с неминимальным количеством созвездий, в которых количество звезд больше 50

```
SELECT name, quantity FROM  
(SELECT a.name, count(a.id) AS quantity  
FROM area AS a  
INNER JOIN consellations AS c  
ON (a.id = c.id_area AND c.quantity_of_visible_stars > 50)  
GROUP BY a.id) AS counts  
WHERE quantity != (SELECT min(quantity) FROM  
(SELECT a.name, count(a.id) AS quantity  
FROM area AS a  
INNER JOIN consellations AS c  
ON (a.id = c.id_area AND c.quantity_of_visible_stars > 50)  
GROUP BY a.id) AS counts);
```