

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ – ПРОЦЕССОВ УПРАВЛЕНИЯ

Каменева Дарья Сергеевна

STARS

Преподаватель:

Филиппов Р.О.

Санкт-Петербург 2017

## 1. Схема

Здесь представлена структура базы данных «Звезды».

### *Stars*

- star\_id
- star\_name
- spectral\_type
- variable\_star
- star\_type\_id
- constellation\_id

### *Brightness*

- brightness\_id
- visible\_mag

### *Catalogues*

- catalogue\_id
- catalogue\_name
- year\_of\_publication
- Numofobj
- Abbreviation

### *Constellations*

- constellation\_id
- constellation\_name

### *Star\_types*

- star\_type\_id
- type\_name

### *Stars\_brightness\_relationships*

- star\_id
- brightness\_id

### *Stars\_catalogues\_relationships*

- star\_id
- catalogue\_id
- numincat

## 2. Описание базы данных

В данной БД представлены некоторые сведения о 25 звездах, имеющих самую большую видимую звездную величину(блеск). Абсолютная звездная величина в данном случае в расчет не берется. Иными словами, в данной БД содержится информация о 25 самых ярких звездах, наблюдаемых с земли. В дальнейшем база может быть дополнена информацией о других звездах и астрономических объектах.

**Stars** (содержит основную информацию о звездах)

*star\_id* (PRIMARY KEY)

*star\_name*

Название звезды уникально и не может отсутствовать в таблице. Оно может содержать слова, начинающиеся как с заглавных, так и со строчных букв (ограничения на написание отсутствуют, кроме размера: не более 30 символов).

*spectral\_type*

Спектральный тип звезды

*variable\_star*

Относится ли звезда к классу переменных звезд

*star\_type\_id*

В данном поле записано id класса (типа) звезды и прикреплена ссылка (внешний ключ) к отдельной таблице с типами звезд. Тип звезды может быть неизвестен.

*constellation\_id*

В данном поле записано id созвездия и прикреплена ссылка (внешний ключ) к отдельной таблице с созвездиями. Созвездие может быть неизвестно.

**Brightness** (содержит информацию о видимой звездной величине (блеск) звезды )

*brightness\_id* (PRIMARY KEY)

*visible\_mag*

Видимая звездная величина - мера яркости небесного тела. Уникальна, не может отсутствовать в таблице и должна быть меньше 5.

**Catalogues** (содержит информацию о звездных каталогах )

*catalogue\_id* (PRIMARY KEY)

*catalogue\_name*

Название каталога уникально и не может отсутствовать в таблице. Оно может содержать слова, начинающиеся как с заглавных, так и со строчных букв (ограничения на написание отсутствуют, кроме размера: не более 80 символов)

*year\_of\_publ*

Год последней публикации каталога (не может быть раньше 01.01.1900).

*Numofobj*

Количество объектов в каталоге (рассматриваются только каталоги в которых более 1000 объектов). Аббревиатура названия каталога

*Abbreviation*

Аббревиатура названия каталога (ограничения на написание отсутствуют, кроме размера: не более 30 символов)

## *Constellations* (содержит информацию о созвездиях )

*constellation\_id* (PRIMARY KEY)

*constellation\_name*

Название созвездия уникально и не может отсутствовать в таблице. Оно может содержать слова, начинающиеся как с заглавных, так и со строчных букв (ограничения на написание отсутствуют, кроме размера: не более 30 символов).

## *Star\_types* (содержит информацию о звездных типах )

*star\_type\_id* (PRIMARY KEY)

*type\_name*

Название созвездия уникально и не может отсутствовать в таблице. Оно может содержать слова, начинающиеся как с заглавных, так и со строчных букв (ограничения на написание отсутствуют, кроме размера: не более 30 символов).

## *Stars\_brightness\_relationships*

*star\_id* - id звезды

*brightness\_id* - id блеска

Данная таблица связывает звезды и видимую звездную величину между собой. Одна звезда может иметь несколько видимых звездных величин и разные звезды могут иметь одинаковый блеск.

## *Stars\_catalogues\_relationships*

*star\_id* - id звезды

*catalogue\_id* - id каталога

*Numincat* - номер звезды в каталоге

Данная таблица связывает каталоги и звезды между собой. Показывает звезду, каталог, в котором она содержится и номер звезды в этом каталоге. Естественно, одна звезда может содержаться в нескольких каталогах сразу.

### 3. Запросы

#### Легкие запросы

1. Выбирает по одной звезде из каждого созвездия

```
SELECT DISTINCT ON (Constellation_ID) * FROM Stars ORDER BY  
Constellation_ID
```

Оптимизация:

Был добавлен индекс для осуществления фильтрации по id созвездия

```
CREATE INDEX ON Stars(Constellation_ID);
```

2. Выбирает положительный блеск, сортирует по возрастанию

```
SELECT * FROM Brightness WHERE Visible_Mag>0  
  
ORDER BY Visible_Mag
```

Оптимизация: был добавлен индекс для осуществления фильтрации по блеску с условием по величине блеска

```
CREATE INDEX ON Brightness(Visible_Mag) WHERE Visible_Mag>0;
```

3. Выбирает все созвездия, начинающиеся на "C"

```
SELECT * FROM Constellations WHERE Constellation_name LIKE 'C%'
```

Оптимизация: был добавлен индекс для осуществления фильтрации по названию созвездия с условием на первую букву созвездия

```
CREATE INDEX ON Constellations(Constellation_name text_pattern_ops)  
WHERE Constellation_name LIKE 'C%';
```

4. Подсчитывает количество переменных звезд

```
SELECT count(s.Star_ID) AS "number of stars",  
s.Variable_star AS "Variable_star"  
FROM Stars AS s WHERE s.Variable_star='no'  
GROUP BY s.Variable_star
```

Оптимизация: был добавлен индекс для осуществления фильтрации по блеску с условием на неперенность

```
CREATE INDEX ON Stars(Variable_star) WHERE Variable_star='no';
```

### Средние запросы

1. Возвращает таблицу Stars, заменяя в ней ID созвездия и типа звезды на соответствующие названия из таблиц Constellations и Star\_types

```
SELECT Star_ID, Star_name, Spectral_type, Variable_star, Type_name AS  
Star_Type, Constellation_name AS Constellation  
FROM Stars AS s LEFT OUTER JOIN Constellations AS c  
ON s.Constellation_ID=c.Constellation_ID  
LEFT OUTER JOIN Star_types AS t ON s.Star_type_ID=t.Star_type_ID
```

Оптимизация: т.к. индекс по id созвездия уже существует, был добавлен только индекс для осуществления фильтрации по id типа звезды

```
CREATE INDEX ON Stars (Star_type_ID ) ;
```



2. Выводит по одной звезде для каждого значения блеска, сортируя записи по возрастанию блеска.

```
SELECT DISTINCT ON (Visible_Mag) s.Star_ID,s.Star_name,b.Visible_Mag AS  
Brightness  
FROM Stars AS s INNER JOIN Stars_Brightness_relationships AS sb  
ON s.Star_ID=sb.Star_ID  
INNER JOIN Brightness AS b ON sb.Brightness_ID=b.Brightness_ID  
ORDER BY Visible_Mag
```

Оптимизация: было добавлено два индекса для осуществления фильтрации по id звезды и по блеску

```
CREATE INDEX ON Stars(Star_ID) ;  
CREATE INDEX ON Brightness(Visible_Mag) ;
```

3. Выбирает все звезды отсортированные по алфавиту, для которых имеются записи о номере в каталоге(ах), номер звезды в каталоге, название и аббревиатуру соответствующего каталога.

```
SELECT s.Star_ID,s.Star_name,sc.NumInCat,c.Catalogue_name,c.Abbreviation  
FROM Stars AS s INNER JOIN Stars_Catalogues_relationships AS sc  
ON s.Star_ID=sc.Star_ID  
INNER JOIN Catalogues AS c ON sc.Catalogue_ID=c.Catalogue_ID  
ORDER BY Star_name
```

Оптимизация: был использован индекс для осуществления фильтрации по id звезды и добавлен индекс для осуществления фильтрации по id каталога

```
CREATE INDEX ON Catalogue (Catalogue_ID);
```

## Сложные запросы

1. Определяет самый распространенный тип звёзд, выводит его название и кол-во звезд этого типа. (если несколько типов имеют равное кол-во звёзд - выводит их все)

```
WITH Toptype (Star_type, stars_in_type) AS (SELECT st.type_name AS
star_type, count(s.Star_ID)
      FROM Stars AS s FULL OUTER JOIN Star_types AS st
      ON s.Star_type_ID=st.Star_type_ID
      GROUP BY type_name)
SELECT * FROM Toptype
      WHERE (stars_in_type = (SELECT MAX(stars_in_type) FROM Toptype))
```

2. Возвращает данные таблицы Stars для всех бинарных звёзд, принадлежащих созвездиям, начинающимся с буквы С, заменяя ID созвездия и типа звезды на соответствующие названия из таблиц Constellations и Star\_types

```
SELECT s.Star_ID,s.Star_name,s.Spectral_type,s.Variable_star,t.Type_name AS
Star_Type, c.Constellation_name AS Constellation
FROM ( Stars AS s INNER JOIN Constellations AS c
      ON (s.Constellation_ID=c.Constellation_ID AND Constellation_name LIKE 'C%')
      INNER JOIN Star_types AS t ON (s.Star_type_ID=t.Star_type_ID AND
      Type_name='star'))
GROUP BY
c.Constellation_name,s.Star_ID,s.Star_name,s.Spectral_type,s.Variable_star,t.
Type_name
```

3. Определяет самую яркую из имеющих блеск меньше 1 звезд звезду в созвездии, содержащем больше всего звезд. Возвращает ID и название созвездия, кол-во звезд в этом созвездии, ID, имя и яркость звезды.

```
WITH fin AS (WITH fir AS (WITH Topconst AS (SELECT c.Constellation_ID AS
const_id, Constellation_name AS Constellation, count(Star_ID) AS st_in_const
FROM Stars AS s FULL OUTER JOIN Constellations AS c
ON s.Constellation_ID=c.Constellation_ID
GROUP BY Constellation, c.Constellation_ID)
SELECT * FROM Topconst
WHERE (st_in_const = (SELECT max(st_in_const) FROM Topconst))),
sec AS (WITH maxbrig AS (SELECT s.Star_ID, s.Star_name, s.Constellation_ID as
sid, b.Visible_Mag AS Brightness
FROM Stars AS s INNER JOIN Stars_Brightness_relationships AS sb
ON s.Star_ID=sb.Star_ID
INNER JOIN Brightness AS b ON (sb.Brightness_ID=b.Brightness_ID
AND (b.Visible_Mag<=1)))
SELECT * FROM maxbrig)
SELECT const_id, Constellation, st_in_const, Star_ID, Star_name, Brightness
FROM fir INNER JOIN sec ON const_id =sid )
SELECT * FROM fin WHERE (Brightness = (SELECT max(Brightness) FROM fin))
```

