

Имя таблицы	Имя поля	Тип поля	Примечание
FAKULTET	KOD_F	Integer	<u>PRIMARY KEY</u>
	NAZV_F	Char, 30	
SPEC	KOD_S	Integer	PRIMARY KEY
	KOD_F	Integer	
STUDENT	NAZV_S	Char, 50	
	KOD_STUD	Integer	PRIMARY KEY
	KOD_S	Integer	
	FAM	Char, 30	
	IM	Char, 15	
	OT	Char, 15	
	STIP	Decimal, 3	
	BALL	Decimal, 3	

Примеры запросов SELECT на языке SQL:

SQL Пример SELECT №1

Выбрать студентов, получающих стипендию, равную 150.

SELECT fname **FROM** STUDENT **WHERE** STIP=150;

С помощью данного **SQL** запроса **SELECT** выбираются все значения из таблицы STUDENT, поле STIP которых строго равно 150.

SQL Пример SELECT №2

Выбрать студентов, имеющих балл от 82 до 90. Студенты должны быть отсортированы в порядке убывания балла.

SELECT FAM **FROM** STUDENT **WHERE** BALL **BETWEEN** 81 **AND** 91 **ORDER BY** BALL **DESC**;

Как видно из **SQL** примера, чтобы выбрать студентов, которые имеют балл от 82 до 90, мы используем условие **BETWEEN**. Чтобы отсортировать в убывающем порядке **DESC**.

SQL Пример SELECT №3

Выбрать студентов, фамилии которых начинаются с буквы «А».

SELECT FAM **FROM** STUDENT **WHERE** FAM **LIKE** 'А%';

Для того, чтобы выбрать фамилии, начинающиеся с буквы «А», мы используем оператор **SQL LIKE** для поиска значений по образцу.

SQL SELECT Пример №4

Подсчитать средний балл на каждом факультете.

SELECT NAZV_F As Название, **ROUND**(**AVG**(BALL), 2) As СредБалл **FROM** FAKULTET, SPEC, STUDENT **WHERE** STUDENT.KOD_S=SPEC.KOD_S **AND** SPEC.KOD_F=FAKULTET.KOD_F **GROUP BY** NAZV_F;

Пример запроса **SQL SELECT** показывает нам использование функции **SQL AVG** для вычисления среднего значения, **ROUND** для округления значения, раздела **GROUP BY** для группировки столбцов.

SQL SELECT. Пример №5.

Подсчитать количество студентов, обучающихся на каждом факультете. Вывести в запросе название факультета, комментарий – «обучается», количество студентов, комментарий «человек».

```
SELECT NAZV_F||' обучается '||COUNT(STUDENT.BALL)||'  
человек' AsCountStudOnFakultet FROM FAKULTET, SPEC, STUDENT WHERE  
STUDENT.KOD_S=SPEC.KOD_S AND SPEC.KOD_F=FAKULTET.KOD_F GROUP BY  
NAZV_F;
```

SQL SELECT. Пример №6.

Упорядочить студентов по факультетам, специальностям, фамилиям.

```
SELECT NAZV_F, NAZV_S, FAM FROM FAKULTET, SPEC, STUDENT WHERE  
STUDENT.KOD_S=SPEC.KOD_S AND SPEC.KOD_F=FAKULTET.KOD_F ORDER  
BY NAZV_F, NAZV_S, FAM;
```

SQL SELECT. Пример №7.

Определить, кто учится на специальности, к которой относится студент «Асанов».

```
SELECT FAM FROM STUDENT WHERE STUDENT.KOD_S=(SELECT KOD_S  
FROM STUDENT WHERE FAM='Асанов');
```

В данном **SQL** примере мы используем подзапрос **SQL SELECT**, который возвращает код специальности, на которой учится студент по фамилии Асанов.

SQL SELECT. Пример №8.

*Показать, какие специальности встречаются в таблице **STUDENT**. Дубликаты исключить. Вывести в запросе названия специальностей.*

```
SELECT DISTINCT NAZV_S FROM SPEC, STUDENT WHERE  
STUDENT.KOD_S=SPEC.KOD_S;
```

Здесь мы с помощью **SQL** ограничения **DISTINCT** выводим только различные значения.

SQL SELECT. Пример №9.

Извлечь из базы данных все данные по сотрудникам, принятым на работу после 01.01.1980 г. в формате “Сотрудник < фамилия сотрудника и его инициалы> принят на работу < дата принятия на работу>”.

```
SELECT CONCAT(CONCAT(CONCAT('Сотрудник ', sname),  
CONCAT(SUBSTR(fname, 0, 1), SUBSTR(otch, 0, 1))), CONCAT('принят на работу ',  
acceptdate)) FROM employees WHERE acceptdate > to_date('01.01.80','dd.mm.yyyy');
```

В данном **SQL SELECT**, используя **SQL** функцию **CONCAT** мы выводим все поля таблицы в одну строку. **SQL** функция **to_date** возвращает привычное для СУБД значение даты.

SQL SELECT. Пример №10.

Извлечь из базы данных перечень должностей, которые имеют сотрудники следующих отделов: ‘БИОТЕХНОЛОГИЙ’, ‘ИНЖЕНЕРНОЙ ЭКОЛОГИИ’. В запросе использовать названия отделов.

```
SELECT pname FROM posts, departments, employees WHERE posts.pid =  
employees.pid AND employees.did = departments.did AND (departments.dname =  
'БИОТЕХНОЛОГИЙ' OR departments.dname = 'ИНЖЕНЕРНОЙ ЭКОЛОГИИ');
```

Пояснение: posts — таблица должностей, departments — таблица отделов, employees — таблица сотрудников, pname — название должности.

SQL SELECT. Пример №11.

Извлечь из базы данных значение максимального личного шифра и фамилию сотрудника с этим номером в формате “Максимальный личный шифр < значение шифра> имеет сотрудник < фамилия сотрудника и его инициалы>”.

SELECT ‘Максимальный личный шифр ‘||eid||’ имеет сотрудник ‘||sname||’ ‘||SUBSTR(fname, 0, 1)||’. ‘||SUBSTR(otch, 0, 1)||’.’ As Максимальный_личный_шифр
FROM employees **WHERE** eid = (**SELECT** MAX(eid) from employees);

Для вывода максимального личного шифра мы устанавливаем условие в **WHERE** так, чтобы шифр был равен полученному максимальному шифру из подзапроса **SELECT**, используя функцию **MAX**.

SQL SELECT. Пример №12.

Получить из базы данных значение числа записей в таблице данных о сотрудниках в формате “Таблица данных о сотрудниках содержит < число записей> записей”.

SELECT ‘Таблица данных о сотрудниках содержит ‘||COUNT(*)||’ записей’
FROM employees;

Используя **SQL** функцию **COUNT**, выводим количество записей таблицы employees.

SQL SELECT. Пример №13.

Получить единым запросом список отделов и должностей предприятия.

SELECT pname **FROM** posts **UNION SELECT** dname **FROM** departments;

С помощью **UNION** мы объединяем два запроса **SQL SELECT** и выводим их как один.

SQL SELECT. Пример №14.

Вывести 30 комментариев начиная с 5 комментария из таблицы replies, кроме комментариев автора ‘Вася’. Данные сортируются по дате добавления комментария в убывающем порядке.

SELECT * **FROM** replies **WHERE** author!=‘Вася’ **ORDER BY** date **DESC** **LIMIT** 5, 30;

SQL SELECT. Пример №15.

*Получить из **SQL** таблицы news одну новость с пометкой «Важные новости».*

SELECT * **FROM** news **WHERE** status=‘Важные новости’ **LIMIT** 1;

В данном **SQL** примере мы выбираем все столбцы поля, у которого столбец status равен ‘Важные новости’. **SQL** ограничение **LIMIT 1** означает, что выбираем только одну запись.

SQL SELECT. Пример №16.

Получить имя письма с идентификатором 1565.

SELECT name **FROM** mail_inbox **WHERE** id=‘1565’;

SQL SELECT. Пример №17.

Получить название рекламного пакета с идентификатором 24.

SELECT title **FROM** ad_packages **WHERE** id=‘24’;

SQL SELECT. Пример №18.

*Вывести столбцы **id**, **title**, **price**, **c_type** с сортировкой по идентификатору в возрастающем порядке.*

```
SELECT id, title, price, c_type FROM ad_packages ORDER BY id ASC;
```

SQL SELECT. Пример №19.

*Вывести все записи из **SQL** таблицы **actions** с идентификатором 1234567890.*

```
SELECT * actions WHERE uid='1234567890';
```