

Node.JS

Параметры в хранимых процедурах



Преподаватель

Портрет

Имя Фамилия

Текущая должность

Количество лет опыта

Какой у Вас опыт - ключевые кейсы

Самые яркие проекты

Дополнительная информация по вашему усмотрению

Корпоративный e-mail

Социальные сети (по желанию)

Важно



Камера должна быть включена на протяжении всего занятия



В течение занятия вопросы задавать в чате или когда преподаватель спрашивает, есть ли у Вас вопросы



Вести себя уважительно и этично по отношению к остальным участникам занятия



Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях



Во время занятия будут интерактивные задания, будьте готовы включить камеру или демонстрацию экрана по просьбе преподавателя

Повторение



Хранимые процедуры



Создание хранимой процедуры



Пример создания простой хранимой процедуры



Вызов хранимой процедуры

План занятия

- IN-параметры
- OUT-параметры
- INOUT-параметры
- Сравнение типов параметров

База данных для работы на занятии

База данных с доступом на запись

hostname: ich-edit.edu.itcareerhub.de

MYSQL_USER: ich1

MYSQL_PASSWORD: ich1_password_ilovedbs

ОСНОВНОЙ БЛОК



IN-параметры





IN

Это входные параметры, которые передаются в процедуру при ее вызове, и их значения используются в процессе выполнения процедуры.

Характеристики IN-параметров



Значения передаются в процедуру



Значения параметров нельзя изменять внутри процедуры



Если значение IN-параметра изменено, то это изменение не передается обратно вызывающей программе.

Пример IN-параметра

Код

```
DELIMITER $$

CREATE PROCEDURE get_employee_name(IN
emp_id INT)
BEGIN
    SELECT name FROM employees WHERE id
= emp_id;
END $$

DELIMITER ;
```



Пояснения

В этом примере процедура `get_employee_name` принимает входной параметр `emp_id` и использует его для поиска имени сотрудника.

Вызов процедуры

```
CALL get_employee_name(1);
```

ВОПРОСЫ



OUT-параметры





OUT

Это выходные параметры, которые используются для того, чтобы возвращать значение из процедуры.

Характеристики OUT-параметров



Значение назначается в процедуре и возвращается после ее выполнения



Параметр не используется для входных данных



Вызывающая программа может получить значение OUT-параметра после завершения процедуры

Пример OUT-параметра

Код

```
DELIMITER $$

CREATE PROCEDURE get_employee_salary(IN
emp_id INT, OUT emp_salary INT)
BEGIN
    SELECT salary INTO emp_salary FROM
employees WHERE id = emp_id;
END $$

DELIMITER ;
```



Пояснения

В этом примере процедура `get_employee_salary` возвращает зарплату сотрудника через параметр `emp_salary`.

Вызов процедуры

```
SET @salary = 0; -- Инициализируем переменную
CALL get_employee_salary(1, @salary); -- Вызываем процедуру и передаем OUT-параметр
SELECT @salary; -- Просматриваем возвращенное значение
```

ВОПРОСЫ



INOUT-параметры





INOUT

Это двунаправленные параметры, которые могут быть использованы как для передачи входных данных в процедуру, так и для возврата данных из нее.

Характеристики INOUT-параметров

-  Параметр используется как для входных, так и для выходных данных
-  Начальное значение передается в процедуру
-  Внутри процедуры параметр может быть изменен, и его новое значение возвращается вызывающей программе

Пример OUT-параметра

Код

```
DELIMITER $$

CREATE PROCEDURE update_employee_salary(INOUT emp_salary INT)
BEGIN
    SET emp_salary = emp_salary * 1.2;
    -- Увеличиваем зарплату на 20%
END $$

DELIMITER ;
```



Пояснения

В этом примере процедура `update_employee_salary` принимает параметр `emp_salary`, увеличивает его значение на 10%, и возвращает измененное значение.

Вызов процедуры

```
SET @salary = 5000; -- Инициализируем переменную
CALL update_employee_salary(@salary); -- Передаем значение и получаем обновленное
SELECT @salary; -- Просматриваем измененное значение
```

ВОПРОСЫ



Сравнение параметров

типов

Сравнение типов параметров

Тип параметра	Передает в процедуру	Возвращает из процедуры	Может изменяться внутри процедуры
IN	Да	Нет	Нет
OUT	Нет	Да	Да
INOUT	Да	Да	Да

ВОПРОСЫ

ПРАКТИЧЕСКАЯ РАБОТА

1. IN параметр

- 
1. Создайте таблицу products с колонками `id` (INT) и `product_name` (VARCHAR), `price`.
 2. Вставьте несколько записей в таблицу.
 3. Создайте хранимую процедуру с IN-параметром для поиска имени товара по его идентификатору.
 4. Вызовите эту процедуру и проверьте результат.

2. OUT параметр

1. Создайте таблицу employees с колонками id (INT), name (VARCHAR), monthly_salary (INT).
2. Вставьте несколько записей в таблицу.
3. Создайте хранимую процедуру, которая возвращает годовую зарплату (ежемесячная зарплата * 12) через OUT-параметр.
4. Вызовите процедуру и проверьте результат, используя переменную для OUT-параметра.

3. INOUT параметр

Создайте хранимую процедуру, которая принимает значение бонуса для сотрудника, увеличивает его на 15% и возвращает новое значение через INOUT-параметр.

1. Создайте переменную для хранения значения бонуса.
2. Создайте хранимую процедуру с INOUT-параметром, которая увеличивает бонус на 15%.
3. Вызовите процедуру с начальными значениями и проверьте измененные значения.

ВОПРОСЫ

Домашнее задание

1. Выведите `id` департамента , в котором работает сотрудник, в зависимости от `Id` сотрудника
2. Создайте хранимую процедуру `get_employee_age`, которая принимает `id` сотрудника (`IN`-параметр) и возвращает его возраст через `OUT`-параметр.
3. Создайте хранимую процедуру `decrease_salary`, которая принимает зарплату сотрудника (`INOUT`-параметр) и уменьшает ее на 10%.

Заключение

