

Лабораторийн ажил 6. ӨС-гийн боловсруулалт-Сторед процедур

Лабораторийн ажлын зорилго

- Өгөгдөл боловсруулахад хувьсагч, давталт, нөхцөл шалгах үйлдлүүд ашиглах
- сторед процедур бичиж сурах
- сторед процедур руу параметр дамжуулах, буцах утгыг параметрээр дамжуулах

Даалгавар-1

Оюутан бүр Бие даалтын ажлаар сонгосон сэдвийн дагуу байгуулсан ӨС схемийг ашиглаж ӨС үүсэх скрипыг ажиллууж , ӨС-г шинээр үүсгэх.

- Бие Даалт №1 дээр үүсгэсэн (Лавлах хүснэгтүүдэд бүх бичлэгийг оруулсан байх -Бодит утгууд оруулна) скрипт файлыг хуулж Лаборатори №6 файл үүсгэ
- Лаборатори №6 скрипт файлыг ажиллуулж ӨС-г үүсгэ

Даалгавар-2

БД-н Функционал шаардлага дээр тодорхойлсон **ОБЪЕКТЫН НЭР БҮРТГЭХ/нэмэх, засах, устгах/** функцүүдийн Кверийг **СТОРЕД ПРОЦЕДУР** ашиглаж бүтээх

- Нэмэх сторед процедур бүтээх. Ж: ОбъектынНэр_ADD(param1 type, param2 type)
- Засах сторед процедур бүтээх. Ж: ОбъектынНэр_EDIT(param1 type, param2 type)
- Устгах сторед процедур бүтээх. Ж: ОбъектынНэр_DEL(param1 type)
- **Оюутан тус бүр дор хаяж ХОЁР ширхэг объектын сторед процедуруудыг бүтээнэ**

Даалгавар-3

Сторед процедурын параметрийг **IN, OUT, INOUT** ашиглаж өгөгдөл оруулах

- **Дор хаяж хоёр ширхэг сторед процедур бүтээнэ**

Даалгавар-4

Хувьсагч зарлаж, параметрийн өгөгдөл зөв эсхийг шалгах

- Хувьсагч ашигласан хоёр ширхэг сторед процедур
- IF ашигласан хоёр ширхэг сторед процедур
- CASE ашигласан хоёр ширхэг сторед процедур

Даалгавар-5

Бүтээсэн сторед процедураа дуудаж ашиглах

- 3 ширхэг нэмэх команд
- 3 ширхэг засах команд
- 3 ширхэг утгах команд

Хугацаа

Лабораторийн цаг дээр үзүүлнэ.

Гүйцэтгэж хамгаалах шаардлага

- Лабораторийн ажил №6 скрипт файлаас ӨС-ийн үүсгэх
- Даалгавар №1,2,3,4 -г ашиглаж бүтээсэн сторед процеруудыг ажиллуулж, бүтээсэн кверийг тайлбарлах
- Бүтээсэн сторед процедураа ашиглаж ӨС-д мэдээлэл нэмэх, засах, устгах командууд ажиллуулах

Нэмэлт материал

Онолын хэсэг, жишээ

Хувьсагч зарлах

Процедур дотор хувьсагчид ашиглах боломжтой.

```
DECLARE хувсагчийн_нэр өгөгдлийн_төрөл;  
DECLARE variable_name datatype(size);
```

Хувьсагчийн нэр хүснэгтийн, баганы нэртэй ижил байж болохгүй.
Зарлахдаа NULL болон DEFAULT утгатай үүсгэж болно:

```
DECLARE variable_name datatype(size) DEFAULT default_value;
```

Жишээ:

```
DECLARE total_sale INT DEFAULT 0;  
DECLARE x, y INT DEFAULT 0;
```

Хувьсагчид утга олгох

SET ашиглаж утга олгох

```
SET FirstNameVar ='Bat';  
SET DuuregVar = 'Bayanzurkh';  
SET total_count = 10;
```

SELECT командыг ашиглан утга олгох

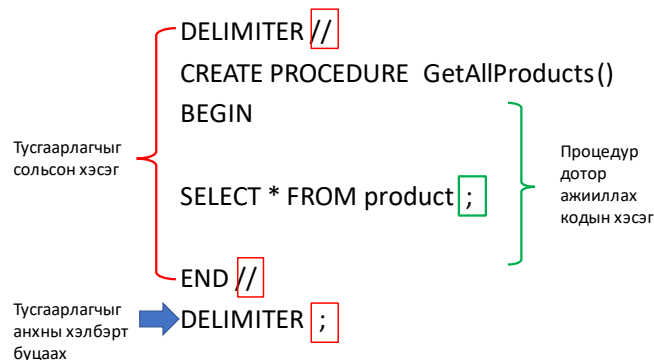
```
DECLARE total_products INT DEFAULT 0;  
SELECT COUNT(*) INTO total_products  
FROM products;
```

Үр дүнд products хүснэгтийн мөрийг тоолж гарах тоог total_products хувьсагчдад утга болгон оноож байна.

Сторед процедур үүсгэх

Бүтэц:

```
DELIMITER //
CREATE PROCEDURE ProcedureName()
    BEGIN
        Code
    END //
DELIMITER ;
```



Процедур дуудах:

```
CALL ProcedureName();
CALL GetAllProduct();
```

Процедур устгах:

```
DROP PROCEDURE ProcedureName;
DROP PROCEDURE GetAllProduct;
```

Процедур – Оролтын параметр ашиглах жишээ:

Тодорхой нэг улсад байрлах оффисуудын хот болон утасны мэдээллийг харуулах процедур :

```
DELIMITER //
CREATE PROCEDURE GetOfficeByCntry(IN countryName VARCHAR(55))
BEGIN
    SELECT city, phone FROM offices
        WHERE country = countryName;
END //
DELIMITER ;
```

Хэрэв USA-д байрлах оффисуудын мэдээллийг харах бол сторед процедур руу дараах байдлаар утгаа дамжуулна:

```
CALL GetOfficeByCountry('USA') ;
```

Гаралтын параметр ашиглах жишээ

Өгөгдсөн төлөвт(received, sent) байгаа захиалгуудыг тоолох сторец процедур бичье :

```
DELIMITER //  
CREATE PROCEDURE CountOrderByStatus(IN orderStatus VARCHAR(25),  
                                     OUT total INT)  
BEGIN  
    SELECT count(ord_num) INTO total FROM orders  
        WHERE ord_status = orderStatus;  
END//  
DELIMITER ;
```

Процедурыг дуудахдаа:

```
SET @total = 0;  
CALL CountOrderByStatus('sent', @total);  
SELECT @total;
```

INOUT параметрийн жишээ

Тэмдэгт мөрийг өөрчлөөд үр дүнг буцаадаг процедур бичье:

```
DELIMITER //  
CREATE PROCEDURE ChangeString(INOUT str VARCHAR(1024))  
BEGIN  
    SET str = concat('example ', UPPER(str));  
END//  
DELIMITER ;
```

Процедураа дуудахдаа:

```
SET @str = 'of a parameter';  
CALL ChangeString(@str);  
SELECT @str;
```

Нөхцөл шалгах үйлдэл

Синтакс- IF

```
IF илэрхийлэл THEN командууд  
    [ELSEIF илэрхийлэл THEN командууд]  
    [ELSE командууд]  
END IF;
```

Нөхцөл шалгах IF үйлдэл ашиглах жишээ:

Тэмдэгт мөрийн төрлөөс нь хамаараад тэмдэгт мөрийг өөрчилдөг процедур бичье:

```
DELIMITER //  
CREATE PROCEDURE ChangeString2(INOUT str VARCHAR(1024),  
                               IN type INT)  
BEGIN  
    IF type = 1  
        THEN SET str = concat('example ', UPPER(str));  
    ELSEIF type = 2  
        THEN SET str = concat('example ', LOWER(str));  
    ELSE
```

```
        SET str='error';  
    END IF;  
END//  
DELIMITER ;
```

Дээрх процедурыг дуудаж ажиллуулахдаа:

```
SET @str = 'of a parameter '  
CALL ChangeString2(@str, 1);  
SELECT @str;
```

Синтакс- CASE

```
CASE  
WHEN илэрхийлэл THEN командууд  
...  
WHEN илэрхийлэл THEN командууд  
ELSE командууд  
END CASE;
```

Нөхцөл шалгах CASE үйлдэл ашиглах жишээ:

Тэмдэгт мөрийн төрлөөс нь хамаараад тэмдэгт мөрийг өөрчилдөг процедур бичье:

```
DELIMITER //  
CREATE PROCEDURE ChangeString4(  
    INOUT str VARCHAR(1024), IN type INT)  
BEGIN  
    CASE type  
        WHEN 1 THEN SET str = concat('example', UPPER(str));  
        WHEN 2 THEN SET str = concat('example', LOWER(str));  
        ELSE SET str='error';  
    END CASE;  
END//  
DELIMITER ;
```

Жишээ: Тогтмол утгаар шалгах

```
SELECT GETDATA(),  
CASE DATEPART(dw, GETDATE())  
    WHEN 1 THEN "Даваа"  
    WHEN 2 THEN "Мягмар"  
    WHEN 3 THEN "Лхагва"  
    WHEN 4 THEN "Пүрэв"  
    WHEN 5 THEN "Баасан"  
    WHEN 6 THEN "Бямба"  
    WHEN 7 THEN "Ням"  
ELSE "Таны оруулсан утга буруу байна"  
END
```

Жишээ: Харьцуулах үйлдлээр шалгах. Насыг олж харгалзах текстийг хэвлэнэ.

```
DECLARE @Var1 date, @ Var2 int  
SET @Var1='2010.09.09'  
SET @Var2=DATEDIFF (yy, @Var1, GetDate())/ 365  
SELECT @Var2,  
CASE
```

```
WHEN @Var2<1 THEN 'нярай'
WHEN @Var2<10 THEN 'хүүхэд нас'
WHEN @Var2<15 THEN 'өсвөр нас'
WHEN @Var2<27 THEN 'залуу нас'
ELSE 'том хүн'
END
```

Жишээ: Оюутан бүрд “123456” гэсэн default нууц үг олгох

```
DELIMITER //
CREATE PROCEDURE SetDefaultPassword ()
BEGIN
UPDATE Student
SET Password = "123456";
END//
DELIMITER ;
```

Дээрх процедурыг дуудаж ажиллуулахдаа:

CALL SetDefaultPassword;

Жишээ: Процедур дотор функц ашиглах.

Оюутны бүтэн нэр, нас(Огнооны ялгавар олдог DATEDIFF функцийг ашиглан оюутны насыг тооцоолж байна), хүйсийг харуулна.

Хүснэгтийн бүтэц:

	Field	Type	Null	Key	Default
▶	studCode	varchar(10)	NO	PRI	NULL
	firstName	varchar(50)	NO		NULL
	lastName	varchar(50)	YES		NULL
	gender	varchar(1)	YES		NULL
	schoolGradYear	varchar(4)	YES		2016
	DateOfBirth	datetime	YES		NULL
	regNumber	varchar(10)	YES		NULL
	mobilePhone	char(8)	YES		NULL

Оруулсан өгөгдөл :

	studCode	firstName	lastName	gender	schoolGradYear	DateOfBirth	regNumber	mobilePhone
▶	SW97B123	Bat	Dorj	f	2021	1997-12-13 00:00:00	AB12345678	999999

```
drop procedure if exists GetStudentsAges;
DELIMITER //
CREATE PROCEDURE GetStudentsAges ()
BEGIN
SELECT concat(FirstName , ' ', LastName) as FullName
, TIMESTAMPDIFF (YEAR, DateOfBirth, now()) AS Age, Gender
FROM students;
END //
DELIMITER ;
```

call GetStudentsAges; /now()-> 23.03.30 /

Үр дүнд:

	FullName	Age	Gender
▶	Bat Dorj	26	f

COUNT IF – Өгөгдсөн нөхцлийг хангаж байгаа бичлэгийг тоолох /MS SQL/

```
SELECT      COUNT(IF(brand = 'Adidas',1,NULL)) Adidas,
            COUNT(IF(brand = 'Nike',1,NULL)) Nike,
            COUNT(IF(brand = 'Reebok',1,NULL)) Reebok
FROM Products;
```

MySQL - COUNT IF – Өгөгдсөн нөхцлийг хангаж байгаа бичлэгийг тоолох

```
SELECT COUNT(IF(condition, 1, NULL)) AS 'alias_name';
```

Жишээ өгөдөл

	studCode	firstName	lastName	gender	schoolGradYear	DateOfBirth	regNumber	mobilePhone
	SW97B005	Dulam	Tushig	f	2021	1997-09-01 00:00:00	AB12345677	77777777
	SW97B123	Bat	Dorj	f	2021	1997-12-13 00:00:00	AB12345679	99999999
▶	SW98B002	Dolgor	Enkh	m	2020	1998-03-30 00:00:00	AB12345678	88888888

SUM IF – Өгөгдсөн нөхцлийг хангаж байгаа бичлэгийн нийлбэрийг олох /MS SQL/

```
SELECT      SUM(IF(prod_type = 'Sneakers',1,0)) AS Sneakers,
            SUM(IF(prod_type <> 'Sneakers',1,0)) AS "Others"
FROM Products;
```

MySQL - Өгөгдсөн нөхцлийг хангаж байгаа бичлэгийн нийлбэрийг олох

```
SELECT SUM(IF(condition, 1, NULL)) AS 'alias_name';
2020 онд, 2020 оноос бусад онд төгссөн оюутны тоог гаргах
SELECT SUM(IF(`schoolGradYear` = 2020,1,0)) AS 'Grad20',
       SUM(IF(`schoolGradYear` <> 2020,1,0)) AS 'GradOther'
FROM students;
```

	studCode	firstName	lastName	gender	schoolGradYear	DateOfBirth	regNumber	mobilePhone
▶	SW97B005	Dulam	Tushig	f	2021	1997-09-01 00:00:00	AB12345677	77777777
	SW97B123	Bat	Dorj	f	2021	1997-12-13 00:00:00	AB12345679	99999999
	SW98B002	Dolgor	Enkh	m	2020	1998-03-30 00:00:00	AB12345678	88888888
	SW99B006	Hulan	Onon	f	2022	1999-05-04 00:00:00	NULL	66666666

Үр дүнд 2020, бусад онд хэдэн оюутан төгссөн тоог гаргана

	Grad20	GradOther
▶	1	3

SUM- CASE хосолсон хэрэглээ

```
SELECT
    SUM( CASE WHEN myColumn=1 THEN 1      ELSE 0 END)
FROM AD_CurrentView
```

NULL /Хоосон / утгыг шалгах

IFNULL (Илэрхийлэл, хоосон утгатай бол авах утга) AS 'alias_name';

IFNULL(илэрхийлэл1, илэрхийлэл2) - Эхний хоосон биш утгыг буцаана

SELECT IFNULL (1, 2) ; // Үр дүн: 1

SELECT IFNULL (NULL, 2) ; //Үр дүн: 2

Өгөгдөл:

	studCode	firstName	lastName	gender	schoolGradYear	DateOfBirth	regNumber	mobilePhone
▶	SW97B005	Dulam	Tushig	f	2021	1997-09-01 00:00:00	AB12345677	77777777
	SW97B123	Bat	Dorj	f	2021	1997-12-13 00:00:00	AB12345679	99999999
	SW98B002	Dolgor	Enkh	m	2020	1998-03-30 00:00:00	AB12345678	88888888
	SW99B006	Hulan	Onon	f	2022	1999-05-04 00:00:00	NULL	66666666

Регистрийн дугааргүй бүртгэлийг “AA0000000” болгох

```
select firstName, lastName,gender, schoolGradYear, DateOfBirth,
       ifnull(regNumber,'AA0000000') as regNumber, mobilePhone
from students;
```

Үр дүнд:

	firstName	lastName	gender	schoolGradYear	DateOfBirth	regNumber	mobilePhone
▶	Dulam	Tushig	f	2021	1997-09-01 00:00:00	AB12345677	77777777
	Bat	Dorj	f	2021	1997-12-13 00:00:00	AB12345679	99999999
	Dolgor	Enkh	m	2020	1998-03-30 00:00:00	AB12345678	88888888
	Hulan	Onon	f	2022	1999-05-04 00:00:00	AA00000000	66666666