1 задание

use UNIVER;

set nocount on;

GO

CREATE PROCEDURE PSUBJECT AS

BEGIN

DECLARE @count int;

SELECT \* FROM dbo.SUBJECT\_T;

SET @count = (SELECT COUNT(\*) FROM SUBJECT\_T);

RETURN @count;

END;

DECLARE @c int;

EXEC @c = PSUBJECT;

PRINT 'Count of rows in SUBJECT table: ' + CAST(@c as varchar(5));

2 задание

USE [UNIVER]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[PSUBJECT] Script Date: 06.04.2022 16:18:28 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER PROCEDURE [dbo].[PSUBJECT] @param varchar(20), @cout int output AS

BEGIN

DECLARE @count int;

SELECT \* FROM dbo.SUBJECT\_T WHERE SUBJECT\_T.PULPIT = @param;

SET @count = (SELECT COUNT(\*) FROM SUBJECT\_T WHERE SUBJECT\_T.PULPIT = @param);

PRINT 'Params: @p = ' + @param + ' @c = ' + CAST(@cout as varchar(5));

RETURN @count;

END;

GO

use UNIVER;

set nocount on;

DECLARE @k int, @r int, @param varchar(20);

EXEC @k = PSUBJECT @param = 'ИСиТ', @cout = @r output;

PRINT 'Count of rows in SUBJECT table: ' + CAST(@r as varchar(5));

PRINT 'Count of rows in SUBJECT table: ' + CAST(@k as varchar(5));

3 задание

GO

use UNIVER;

GO

CREATE procedure PSUBJECT\_3 @param varchar(20) AS

BEGIN

DECLARE @count int;

SET @count = (SELECT COUNT(\*) FROM SUBJECT\_T WHERE SUBJECT\_T.PULPIT = @param);

PRINT 'Params: @p = ' + @param + ' @c = ' + CAST(@count as varchar(5));

SELECT \* FROM dbo.SUBJECT\_T WHERE SUBJECT\_T.PULPIT = @param;

END;

CREATE TABLE #SUBJECT (

SUBJ varchar(10) NOT NULL,

SUBJ\_NAME varchar(100),

PULP varchar(10) NOT NULL

);

INSERT #SUBJECT EXEC PSUBJECT\_3 @param = 'ИСиТ';

SELECT \* FROM #SUBJECT;

4 задание

GO

use UNIVER;

SET nocount on;

GO

CREATE procedure PAUDITORIUM\_INSERT @a char(20), @n varchar(50), @c int = 0, @t char(10) AS

BEGIN

BEGIN try

INSERT into AUDITORIUM(AUDITORIUM,AUDITORIUM\_NAME,AUDITORIUM\_CAPACITY,AUDITORIUM\_TYPE)

values(@a, @n, @c, @t);

RETURN 1;

END try

BEGIN catch

print 'Error number: ' + cast(ERROR\_NUMBER() as varchar(6));

print 'Error message: ' + ERROR\_MESSAGE();

print 'Error line: ' + cast(ERROR\_LINE()as varchar(8));

if ERROR\_PROCEDURE() is not null

print 'Error procedure: ' + ERROR\_PROCEDURE();

print 'Error secerity: ' + cast(ERROR\_SEVERITY()as varchar(6));

print 'Error state: ' + cast(ERROR\_STATE()as varchar(8));

RETURN -1;

END CATCH

END;

DECLARE @paud int = 0;

EXEC @paud = PAUDITORIUM\_INSERT @a='423-1',@n='423-1',@c='90',@t='ЛК-К';

DELETE AUDITORIUM where AUDITORIUM\_NAME='423-1';

Select \* from AUDITORIUM;

5 задание

GO

create procedure SUBJECT\_REPORT @p CHAR(10)

as

declare @rc int = 0;

begin try

declare @tv char(20), @t char(300) = ' ';

IF CURSOR\_STATUS('global','ZkTov')>=-1

BEGIN

DEALLOCATE ZkTov

END

declare ZkTov CURSOR for

select SUBJECT\_T from SUBJECT\_T where PULPIT = @p;

if not exists ( select SUBJECT\_T from SUBJECT\_T where PULPIT = @p)

raiserror('ошибка', 11, 1);

else

open ZkTov;

fetch ZkTov into @tv;

print 'Предметы на кафедре';

while @@fetch\_status = 0

begin

set @t = rtrim(@tv) + ', ' + @t;

set @rc = @rc + 1;

fetch ZkTov into @tv;

end;

print @t;

close ZkTov;

return @rc;

end try

begin catch

print 'Ошибка в параметрах'

if error\_procedure() is not null

print 'Имя процедуры : ' + error\_procedure();

print 'Код ошибки : '+ cast(error\_number() as varchar(6));

print 'Сообщение : '+ error\_message();

print 'Уровень : '+ cast(error\_severity() as varchar(8));

return @rc;

end catch;

declare @rc2 int;

exec @rc2 = SUBJECT\_REPORT @p = 'Луч';

print 'Количество предметов = ' + cast(@rc2 as varchar(3));

exec @rc2 = SUBJECT\_REPORT @p = 'ОХ';

print 'Количество предметов = ' + cast(@rc2 as varchar(3));

exec @rc2 = SUBJECT\_REPORT @p = 'ИСиТ';

print 'Количество предметов = ' + cast(@rc2 as varchar(3));

6 задание

GO

CREATE procedure PAUDITORIUM\_INSERTX @a char(20), @n varchar(50), @c int = 0, @t char(10), @tn varchar(50) AS

BEGIN

BEGIN try

set transaction isolation level SERIALIZABLE;

begin transaction

INSERT into AUDITORIUM\_TYPE (AUDITORIUM\_TYPE,AUDITORIUM\_TYPENAME)

values(@t,@tn);

EXEC PAUDITORIUM\_INSERT @a=@a, @n=@n, @c=@c, @t=@t;

commit transaction;

return 1;

END try

BEGIN catch

print 'Error number: ' + cast(ERROR\_NUMBER() as varchar(6));

print 'Error message: ' + ERROR\_MESSAGE();

print 'Error line: ' + cast(ERROR\_LINE()as varchar(8));

if ERROR\_PROCEDURE() is not null

print 'Error procedure: ' + ERROR\_PROCEDURE();

print 'Error secerity: ' + cast(ERROR\_SEVERITY()as varchar(6));

print 'Error state: ' + cast(ERROR\_STATE()as varchar(8));

RETURN -1;

END catch

END;

GO

DECLARE @paud int = 0;

EXEC @paud = PAUDITORIUM\_INSERTX @a='423-1',@n='423-1',@c='90',@t='ЛК-К', @tn ='test auditorium name';