AFTER — триггер вызывается после всех действий оператора, если оператор был  
выполнен успешно. Синонимом в синтаксисе оператора создания триггера является FOR;  
 INSTEAD OF — триггер вызывается вместо действий, заданных оператором.

use drova;

go

alter TRIGGER insert\_Table\_1 ON Table\_1

after insert

AS

BEGIN

declare @rez varchar(max);

select @rez=(select department from inserted)

insert into Log\_table values (@rez, GETDATE())

END

GO

**USE** SampleDb;

*/\* Таблица AuditBudget используется в качестве*

*журнала логов действий в таблице Project \*/*

GO

**CREATE** **TABLE** AuditBudget (

ProjectNumber CHAR(4) **NULL**,

UserName CHAR(16) **NULL**,

Date DATETIME **NULL**,

BudgetOld FLOAT **NULL**,

BudgetNew FLOAT **NULL**

);

GO

**CREATE** **TRIGGER** trigger\_ModifyBudget

**ON** Project **AFTER** **UPDATE**

**AS** **IF** **UPDATE**(budget)

**BEGIN**

**DECLARE** @budgetOld FLOAT

**DECLARE** @budgetNew FLOAT

**DECLARE** @projectNumber CHAR(4)

**SELECT** @budgetOld = (**SELECT** Budget **FROM** deleted)

**SELECT** @budgetNew = (**SELECT** Budget **FROM** inserted)

**SELECT** @projectNumber = (**SELECT** Number **FROM** deleted)

**INSERT** **INTO** AuditBudget **VALUES**

(@projectNumber, USER\_NAME(), **GETDATE**(), @budgetOld, @budgetNew)

**END**

CREATE TRIGGER trigger\_TotalBudget

ON Project AFTER UPDATE

AS IF UPDATE (Budget)

BEGIN

DECLARE @sum\_old1

FLOAT DECLARE @sum\_old2

FLOAT DECLARE @sum\_new FLOAT

SELECT @sum\_new = (SELECT SUM(Budget) FROM inserted)

SELECT @sum\_old1 = (SELECT SUM(p.Budget)

FROM project p WHERE p.Number

NOT IN (SELECT d.Number FROM deleted d))

SELECT @sum\_old2 = (SELECT SUM(Budget) FROM deleted)

IF @sum\_new > (@sum\_old1 + @sum\_old2) \* 1.5

BEGIN

PRINT 'Бюджет не изменился'

ROLLBACK TRANSACTION

END

ELSE

PRINT 'Изменение бюджета выполнено'

END

CREATE TRIGGER trigger\_orders

ON view\_AllOrders INSTEAD OF INSERT

AS BEGIN

INSERT INTO Orders

SELECT OrderId, Price, Quantity, OrderDate

FROM inserted

END