

USER DEFINE FUNCTION

توابع مانند پروسیجرها مجموعه ای از دستورات T-SQL می باشند و مانند آنها یک سری پارامتر ورودی و مقدار خروجی دارند.

توضیح: تابع معمولاً یک مقدار را برمی گرداند.

توابع به دو دسته تقسیم می شوند:

۱. **Scalar Functions** که مقدار بازگشتی آن ها یک مقدار واحد است.

این توابع در در قسمت **Programmability\Functions\Scalar-Valued Functions** ذخیره می شوند.

۲. **Table Functions** که مقدار بازگشتی آن ها یک جدول است.

این توابع در در قسمت **Programmability\Functions\Table-Valued Functions** ذخیره می شوند.

دستور کار: هر یک از توابع زیر را ایجاد و اجرا نمایید و خروجی آن را مشاهده و در گزارش کار خود بنویسید.

۱. Scalar Functions

مثال (۱) تابعی بنویسید که شماره پرسنلی را بگیرد و نام و نام خانوادگی او را نمایش می دهد.

```
CREATE FUNCTION fname_lname (@No int) RETURNS nvarchar(50)
AS
BEGIN
    declare @ret_Value nvarchar (50)
    select @ret_Value=fname + '!' +lname
    from Spec
    where emp_no =@No
    return (@ret_Value)
END
```

در این مثال **@No** یک پارامتر ورودی است.

حال Query را اجرا نمایید ، مشاهده می کنید که یک روال در قسمت

Programmability\Functions\Scalar-valued functions ایجاد شده است. (این تابع یک رشته از

نوع nvarchar که شامل نام و نام خانوادگی است برمی گرداند)

برای فراخوانی این تابع می توانید یکی از دو کوئری زیر را اجرا نمایید.

(فرض کنید شماره پرسنلی = ۱۰۰۳)

```
print dbo.fname_lname(1003)
```

یا

```
select dbo.fname_lname(1003)
```

نکته: ۱ - هنگام فراخوانی توابع پارامتر ورودی در () بعد از اسم تابع قرار میگیرد.

۲ - حتی اگر پارامتر ورودی نداشته باشیم () حتما لازم است.

توضیح: چنانچه تابع ایجاد شده نیاز به تغییر دارد از دستور `ALTER function function_name` و برای حذف تابع از `DROP function function_name` استفاده میکنیم.

مثال (۲) فرض کنید `Sum(price)` مجموع قیمت پروژه ها در سال است که می خواهیم $\frac{1}{12}$ آن که هزینه متوسط پروژه ها در ماه است را بدست آوریم.

```
CREATE FUNCTION pmon (@par int) RETURNS decimal(16,2)
BEGIN
declare @sum_m decimal (16,2)
select @sum_m =sum(price) / @par from project
return @sum_m
END
```

* این تابع را فراخوانی کنید.

۲. Table Functions

مثال (۳) تابع زیر شماره پرسنلی ، نام و نام خانوادگی و شماره تلفن کارکنان را برمی گرداند.

```
CREATE FUNCTION Fspec() RETURNS Table
AS
return (Select emp_no,fname,lname,phon from spec)
```

توضیح مهم: اگر بدنه تابع فقط یک جمله می باشد نیازی به `Begin` و `End` نیست.

* این تابع را فراخوانی کنید.

تمرین ۱: تابعی بنویسید که شماره پرسنلی ، نام و نام خانوادگی ، شغل ، تاریخ شروع کار همه کارکنان را برگرداند.

- تابع فوق را فراخوانی نمایید.

تمرین ۲: تابعی بنویسید که شماره پرسنلی ، نام و نام خانوادگی و گروه شغلی افرادی که گروه شغلی آنها کمتر از v1 است را برگرداند.

- تابع فوق را با مقدار $v1 = 13$ فراخوانی نمایید.

تمرین ۳: در سیستم کتابخانه

تابعی بنویسید که شماره مشتری را بگیرد و کتابهای امانت داده شده به مشتری شامل (شماره مشتری ، نام

مشتری ، کد کتاب ، نام کتاب ، تاریخ امانت، تاریخ برگشت) را برگرداند.

- تابع فوق را برای یک مشتری فراخوانی نمایید.

- تعداد کتاب های امانت داده شده به مشتری را فراخوانی کنید.

کار روی پروژه: روی پایگاه داده خود تابعی بنویسید که تاریخ سیستم را به شمسی به صورت رشته ۸

تابی برگرداند و روی صفحه اول پروژه نمایش دهد.