

به نام خدا

آموزش رجیستری و کار با رجیستری در دلفی **Registry In Delphi**

کاری از شرکت : اندیشه پرور پارس باستان



رجیستری چیست ؟

رجیستری يك پایگاه داده با ساختار درختواره ای است كه از آن برای ذخیره و بازیابی تنظیمات و پیکربندیهای موجود در ویندوز های ۳۲ بیتی مشتمل بر ویندوزهای 9x, XP, 2000, Me, NT استفاده می شود . رجیستری شامل اطلاعات و تنظیمات کلیه قطعات سخت افزاری ، نرم افزارها ، کاربران و اولویت های آنها می باشد . زمانی كه كاربر تغییراتی در تنظیمات كنترل پانل ، انتصاب فایل به برنامه كاربري ، تنظیم System Policy و نرم افزارهای نصب شده را اعمال می كند ، بازتاب این تغییرات بصورت ذخیره سازی اطلاعاتی در رجیستری انعكاس می یابد .

رجیستری از دو فایل با نامهای USER.DAT و SYSTEM.DAT تشکیل شده است. البته اگر شما عضو يك شبکه هستید ، فایل دیگری تحت عنوان CONFIG.POL نیز وجود دارد. همه این ۳ فایل ، از نوع Hidden , System , ReadOnly هستند و در شاخه اصلی ویندوز قرار دارند.

زمینه پیدایش رجیستری :

تا قبل از نسخه های ۳۲ بیتی ویندوز بسیاری از پیکربندیهای مختلف نرم افزاری و سخت افزاری در فایل های win.ini و System.ini ذخیره می شوند . البته تنها از این دو فایل جهت ذخیره سازی پیکربندیها استفاده نمی شد بلکه با نصب هر نرم افزار یا هر سخت افزار يك سري فایل های ini دیگر نیز به سیستم اضافه می شوند این ساختار دارای مشکلات اساسی بود .

اساسی ترین مشکل آن ، نبودن غالب استاندارد جهت ثبت این تنظیمات بود به اینصورت كه هربرنامه ای جهت ذخیره سازی تنظیمات مربوط به خود از يك ساختار استفاده می كرد كه باساختارهای مربوط به برنامه های دیگر لزوما هماهنگی نداشت . علاوه بر این مكان ذخیره سازی این اطلاعات نیز پراكنده بود و فایل های ini مذکور در كل درایوها می توانستند توزیع شوند . بدین صورت بود كه مایكروسافت به فكر افتاد كه يك قالب استاندارد جهت ذخیره سازی تنظیمات مربوط ارائه كند كه با ظهور ویندوز ۹۵ ، رجیستری به عنوان يك راه جهت حل مشکلات فوق ارائه شد .

محل ذخیره سازی در رجیستری :

فایلهایی كه رجیستری را تشکیل می دهند بسته به نسخه ویندوز به طور متفاوتی ذخیره می شوند. در ویندوز 9x ، كه همان ویندوز ۹۸ باشد، رجیستری در دو فایل مخفی در دایرکتوری ویندوز با نام های System.dat و User.dat ذخیره می شود، در حالیکه در ویندوز NT این فایلها به طور جداگانه در دایرکتوری زیر ذخیره میشود :

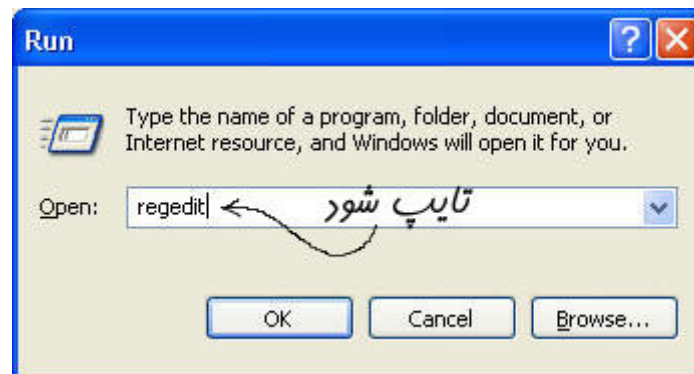
Windows/system32/config

Backup گیری از رجیستری :

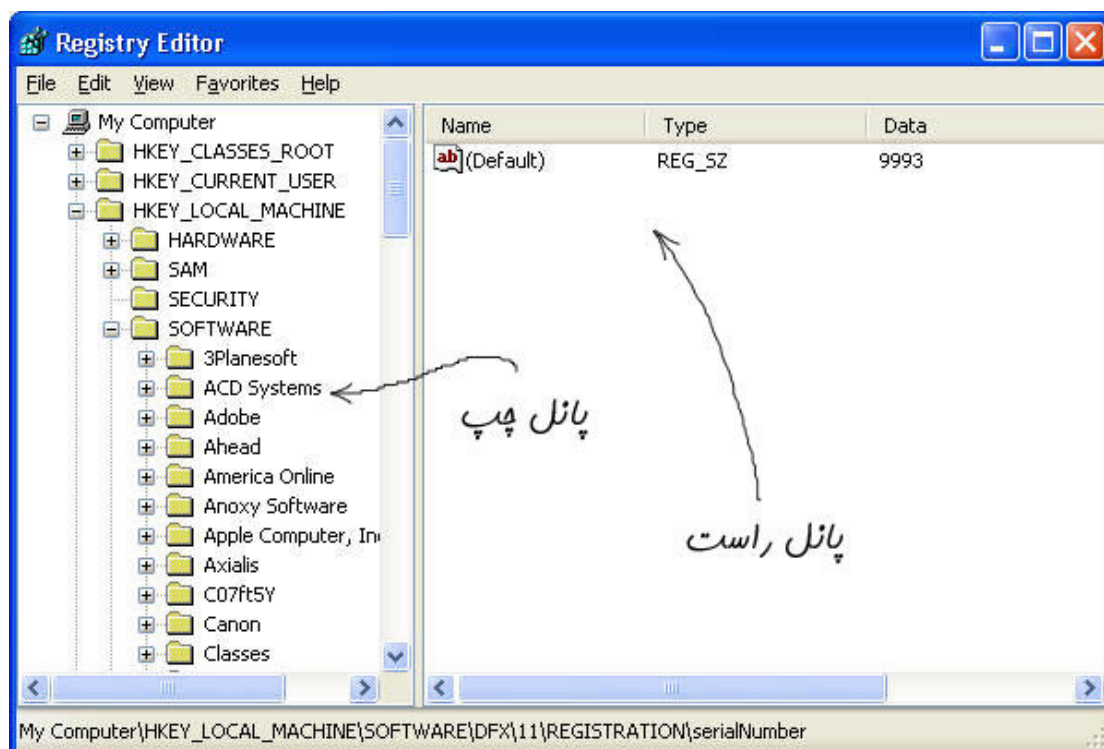
برای اینکه از رجیستری خود Backup بگیرید ، می بایست پس از اجرا نمودن نرم افزار Registry Editor، گزینه Export را از منوی File انتخاب کنید. اگر كه با رجیستری كار كردید و دیدید كه اشتباهی يك فایل را پاک کرده و یا نام آنرا تغییر دادید ، می توانید بر روی فایل ایجاد شده (همان فایلی را كه از طریق فرمان Export ذخیره كردید) ، دوبار کلیک کنید و به سوالی كه از شما می پرسد ، جواب Yes دهید. دقت داشته باشید كه پسوند فایل های رجیستری ، REG می باشد.

نحوه استفاده از Regedit :

برای اجرا كردن ویرایشگر رجیستری باید به پوشه ویندوز رفته و در آن پوشه به دنبال فایل regedit گشت. برای سهولت كار میتوانید با استفاده از Run در منوی Start این فایل را سریعتر پیدا کرده و آنرا اجرا کنید.



پس از فشردن OK با این منو مواجه خواهید شد.



محیط ویرایشگر رجیستری :

ویرایشگر رجیستری از دو پانل تشکیل شده است.

پانل سمت چپ : این پانل نمایشگر نحوه سازماندهی رجیستری می باشد . اولین و بالاترین مدخل My computer است که دارای چندین کلید است که در ویرایشگر رجیستری بصورت پوشه نمایش داده می شود . هر کلید می تواند دارای چند زیر کلید باشد و این امر تا هر سطح می تواند ادامه یابد . جهت باز نمودن هر کلید و مشاهده زیر کلیدهای آن باید بر روی علامت مثبت (+) آن کلید کلیک نماییم و جهت بستن کلید می توان بر روی علامت منفی (-) آن کلیک نمود.

پانل سمت راست : این پانل ، مداخل حاوی مقادیر یک کلید در رجیستری را نشان می دهد . هر کلید می تواند شامل چندین مقدار باشد که هر یک از این مقادیر دارای نام و محتوا (Data) هستند .

توجه داشته باشید که آیکن مربوط به مدخل هر مقدار در رجیستری قبل از آن مقدار قرار دارد . آیکن مربوط به مداخل مشخص می نمایند که چه نوع از مقدار داده ای در مدخل رجیستری استفاده شده است .

نحوه ایجاد يك كلید با مقدار :

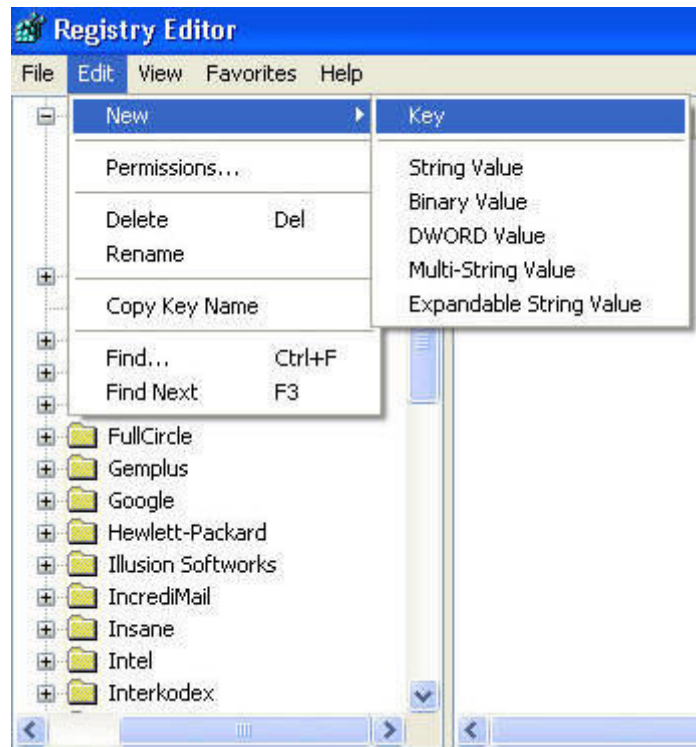
معمولا بوجود آوردن يك مقدار یا كلید جدید بی ضرر است . در صورتیکه ویندوز 9x یا یکی از برنامه های تحت آن از كلید جدید استفاده نکنید. در اینجا توضیح خواهیم داد که چگونه يك كلید یا مقدار جدید ایجاد کنید .

ایجاد كلید :

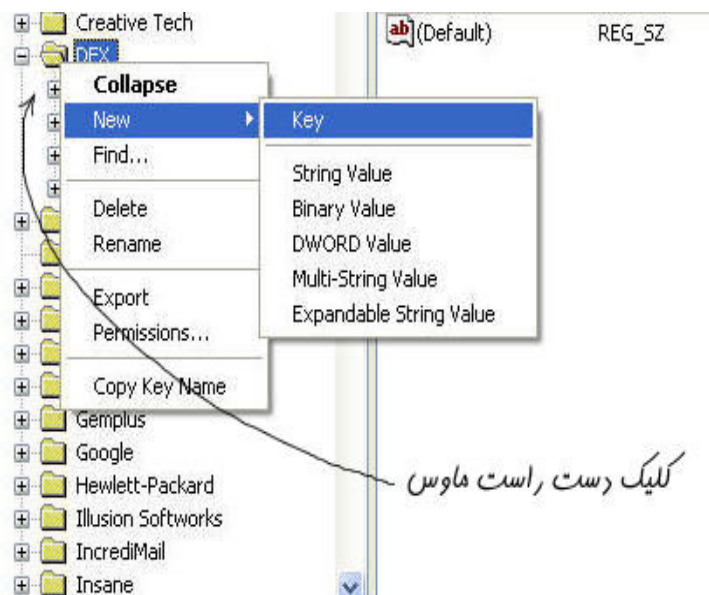
جهت ایجاد زیر كلید ، ابتدا باید كلید مورد نظر را انتخاب کنیم .



روش اول : از منوی Edit روی گزینه New قرار گرفته تا زیر منوی آن باز شود، سپس بر روی گزینه Key کلیک می کنیم. یک زیر كلید به نام New key#1 ایجاد می شود. که می توانیم نام آن را تغییر دهیم .



روش دوم : روی کلید مورد نظر جهت ایجاد زیر کلید ، کلیک راست می نماییم. روی گزینه New قرار گرفته تا زیر منوی آن باز شود ، سپس بروی گزینه Key کلیک می نماییم. یک زیر کلید به نام New key#1 ایجاد می شود. که می توانیم نام آن را تغییر دهیم.



ایجاد مقدار :

جهت ایجاد مقدار ابتدا باید کلید مورد نظر را که می خواهیم مقدار جدیدی را در آن ایجاد کنیم انتخاب نماییم .

روش اول : بروی گزینه New از منوی Edit کلیک می کنیم تا زیر منوی آن باز شود. سپس گزینه مربوط به هر نوع از مقادیر مانند String , Binary , DWORD که می خواهیم ایجاد کنیم کلیک می نماییم . سپس مقدار مورد نظر ما ایجاد می شود .

روش دوم : روی کلید مورد نظر جهت ایجاد مقدار، کلیک راست می نماییم. روی گزینه New قرار گرفته تا زیر منوی آن باز شود ، سپس گزینه مربوط به هر نوع از مقادیر که می خواهیم ایجاد کنیم کلیک می نماییم . سپس مقدار مورد نظر ما ایجاد می شود .

رجیستری در دلفی :

رجیستری رو در زبان عامیانه قلب ویندوز NT می دانند . رجیستری یک مرکز بسیار مهم در ویندوز می باشد که در آنجا تمامی برنامه ها و مقدار ها و فرمت ها و تنظیمات ثبت میشوند ، در رجیستری می توان به تنظیمات ویندوز دست پیدا کرد و با استفاده از آن بدون استفاده از کنترل پنل تنظیمات ویندوز رو عوض کرد .

برای کار با رجیستر به یونیت با همین نام Registry ارائه داده که تمام نیاز ها رو برای کار با رجیستری تامین می کنه که شامل کلاس TRegistry می باشد که دارای روال ها و تابع های خاصی مثل : CreateKey , WriteString , DeleteKey , MoveKey , ReadFloat و می باشد .

نحوه اصلی کار با رجیستری و ثبت کردن مقدار ، ایجاد یک کلید و.... به این صورت می باشد که ، ابتدا یونیت رجیستری رو به لیست Uses یونیتمون اضافه می کنیم به شکل زیر :

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, Registry ;

خب حالا به متغیر باید از نوع کلاس TRegistry تعریف کنیم تا بتوانیم از توابع آن ها استفاده کنیم . به شکل زیر :

Var
B : TRegistry;

سپس باید یک نوع از کلاس را بوجود بیاوریم که طبق معمول باید از constructor کریت (Create) مر بوط به اکثر کلاس ها استفاده کنیم ، به شکل زیر :

B := TRegistry.Create ;

برای کار با رجیستری باید به یک لانه مربوط در آن اشاره کنیم که عبارتند از :

HKEY_CLASSES_ROOT
HKEY_CURRENT_USER
HKEY_LOCAL_MACHINE
HKEY_USERS
HKEY_PERFORMANCE_DATA
HKEY_CURRENT_CONFIG
HKEY_DYN_DATA

برای این که یکی از این مقدار ها را انتخاب کنیم ، باید از پراپرتی RootKey به شکل زیر استفاده کنید:

B . RootKey := HKEY_LOCAL_MACHINE;

سپس باید آدرس مربوط به کلیدی که می خواهیم در آن اعمال انجام دهیم رو با استفاده از تابع OpenKey مشخص کنیم به شکل زیر :

B . OpenKey(اجازه بوجود آوردن کلید ، آدرس کلید مربوطه)

در مورد آدرس ، شما باید آدرس کلید مربوطه را بدون KeyRoot آن تایپ کنید (به صورت یک String) و در مورد اجازه بوجود آوردن که با یک مقدار False یا True ، Set می شود ، به این شکل است که شما می توانید با Set کردن آن به صورت True به آن اجازه بدهید تا یک کلید با آدرس که شما تایپ کردید بوجود بیاورد البته ما به شما پیشنهاد می کنیم که مقدار مربوطه را همیشه False کنید چون شما می توانید با تابع CreateKey یک Key بسازید ، پس بهتره که شما از تابع OpenKey آدرس کلید مربوطه تان را لود کنید .

سپس شما در این مرحله قادر هستید تا هر کاری انجام دهید مثلا می توانید :

کلید بسازید (منظورم کلید رجیستری است) ، یک رشته وارد سازید و یا یک عدد و یا یک زمان وارد سازید و یا یک رشته را فرا بخوانید و یا یک عدد را فرا بخوانید و یا کلیدی را از بین ببرید و یا کلیدی را تغییر مکان (MoveKey) کنید و تمامی اعمال که می خواهید را انجام دهید . برای مثال به کد زیر یک نگاه بیندازید ، این کد TaskManager را غیر فعال می کند :

```
var  
dbm : Tregistry ;  
Begin  
  DbM := tregistry.Create ;  
  DbM . RootKey =: HKEY_CURRENT_USER ;  
  DbM . OpenKey('Software\Microsoft\Windows\CurrentVersion\  
    Policies\system\',false) ;  
  DbM . WriteInteger('DisableTaskMgr',0) ;  
  DbM . Free ;  
End ;
```

خب فکر کنم که کار با توابع رجیستری رو آموخته باشید همین طور که می بینید تمام کار ها گفته شده انجام شده و تمام تابع ها آشنا می باشد در ضمن تابع WriteInteger توابعی از این دست برای ثبت یک عدد و یا یک مقدار در رجیستری می باشد این تابع ها خیلی زیادند و همه دارای ساختمانی به شکل زیر می باشند :

;(مقدار , نام مقداری که باید از نو ثبت شود و یا با نویسی شود) نام تابع

همچنین توابعی نیز برای خواندن یک مقدار و باز گرداندن یک مقدار وجود دارد مثل ReadString که اکثرا دارای یک نوع ساختمان می باشند برای مثال با استفاده از کد زیر می توانید مسیر کوکی ها را از رجیستری بخوانید :

```
var
reg : tregistry ;
Cooki : String ;
Begin
    reg := tregistry . Create ;
    reg . RootKey := hkey_current_user ;
    reg . OpenKey('Software\Microsoft\Windows\CurrentVersion\
        Explorer\Shell Folders',true) ;
    Cooki := reg . ReadString('Cookies') ;
End;
```

ساختمان تابع های خواندنی به شکل زیر می باشد :

;(نام عبارتی که باید خوانده شود) نام تابع

قابل توجه دوستان عزیز :

من تمام دستورات زیر را در دلفی اجرا کرده ام و جواب هم گرفته ام و بعد از گرفتن جواب نیز دستورات را در این مقاله قرار داده ام ... پس شما مطمئن باشید که این دستورات حتما جواب می دهند .

چک کردن برای وجود داشتن یک کلید :

Read String , Return Boolean

```
procedure TForm1.Button1Click(Sender: TObject);
var
    b:TRegistry;
begin
    b := TRegistry.Create;
    b.RootKey := HKEY_LOCAL_MACHINE;
    if b.KeyExists('Software\Microsoft\Windows\Currentversion\Policies\Explorer')
    Then
        ShowMessage('Key Exicts')
    else
        ShowMessage('Key Not Exicts');
end;
```

ساختن یک کلید جدید :

Read String , Return Boolean

راه اول :

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var
  b:TRegistry;
begin
  b := TRegistry.Create;
  b.RootKey := HKEY_LOCAL_MACHINE;
  b.OpenKey('Software\Microsoft\Windows\Currentversion\Policies',false);
  if b.CreateKey('Andisheh Parvar PARS Bastan') Then
    ShowMessage('Key Create Complete');
end;
```

راه دوم :

```
procedure TForm1.Button1Click(Sender: TObject);
var
  b:TRegistry;
begin
  b := TRegistry.Create;
  b.RootKey := HKEY_LOCAL_MACHINE;
  if b.CreateKey('Software\Microsoft\Windows\Currentversion\Policies\Andisheh
  Parvar PARS Bastan') Then
    ShowMessage('Create Succesfull')
  else
    ShowMessage('No Create');
end;
```

بستن کلید :

b.Closekey;

حذف کردن کلید :

Read String , Return Boolean

راه اول :

```
procedure TForm1.Button1Click(Sender: TObject);
var
  b:TRegistry;
begin
  b := TRegistry.Create;
  b.RootKey := HKEY_LOCAL_MACHINE;
  if b.DeleteKey('Software\Microsoft\Windows\Currentversion\Policies\Andisheh
  Parvar PARS Bastan') Then
    ShowMessage('Delete SuccessFull')
  else
    ShowMessage('No Delete');
end;
```

راه دوم :

```
procedure TForm1.Button1Click(Sender: TObject);
var
  b:TRegistry;
begin
  b := TRegistry.Create;
  b.RootKey := HKEY_LOCAL_MACHINE;
```

```

if b.KeyExists('Software\Microsoft\Windows\Currentversion\Policies\Andisheh
Parvar PARS Bastan') then
  b.DeleteKey('Software\Microsoft\Windows\Currentversion\Policies\Andisheh
Parvar PARS Bastan')
else
  ShowMessage('No Exicts To Delete');
end;

```

ایجاد مقادارهای مختلف در یک کلید :

```

procedure TForm1.Button1Click(Sender: TObject);
var
  b:TRegistry;
begin
  b := TRegistry.Create;
  b.RootKey := HKEY_LOCAL_MACHINE;
  b.OpenKey('Software\Microsoft\Windows\Currentversion\Policies\Andisheh
Parvar PARS Bastan',false);
  b.WriteString('Persian','Andisheh Parvar PARS Bastan'); //Write String Value
  b.WriteTime('Today Date',Date); // Write Time Value
  b.WriteBool('Question',True); // Write Boolean Value
  b.WriteInteger('Value',100); // Write Integer Value
  b.WriteCurrency('Money',300000); // Write Currency Value
  b.WriteFloat('Float',1.99); // Write Float Value
end;

```

جمع آوری و تکمیل کننده اطلاعات : قاسم آقابابایی نبی

ایمیل : Ghasem_Babaie645@Yahoo.CoM

شرکت : اندیشه پرور پارس باستان

مشاهده پروژه های انجام شده توسط شرکت ما : <http://www.projects.rightclick.ir>

شهریور سال ۱۳۸۵

پایان