

عنوان: آموزش ++VC از مقدماتی تا پیشرفته

نویسنده: حمیدرضا

تاریخ: ۱۳۹۱/۱۲/۲۶ ۱۲:۳۰

آدرس: [www.dotnettips.info](http://www.dotnettips.info)

برچسب‌ها: Win32 Project, Win32 Console Application, C

بحثی که بنده قصد آموزش آن را دارم آموزش ++C در IDE مایکروسافت visual studio می‌باشد. آموزش از پروژه‌های Win32 Console Application شروع شده و قسمت پیشرفته آموزش در پروژه‌های Win32 Project ادامه می‌یابد.

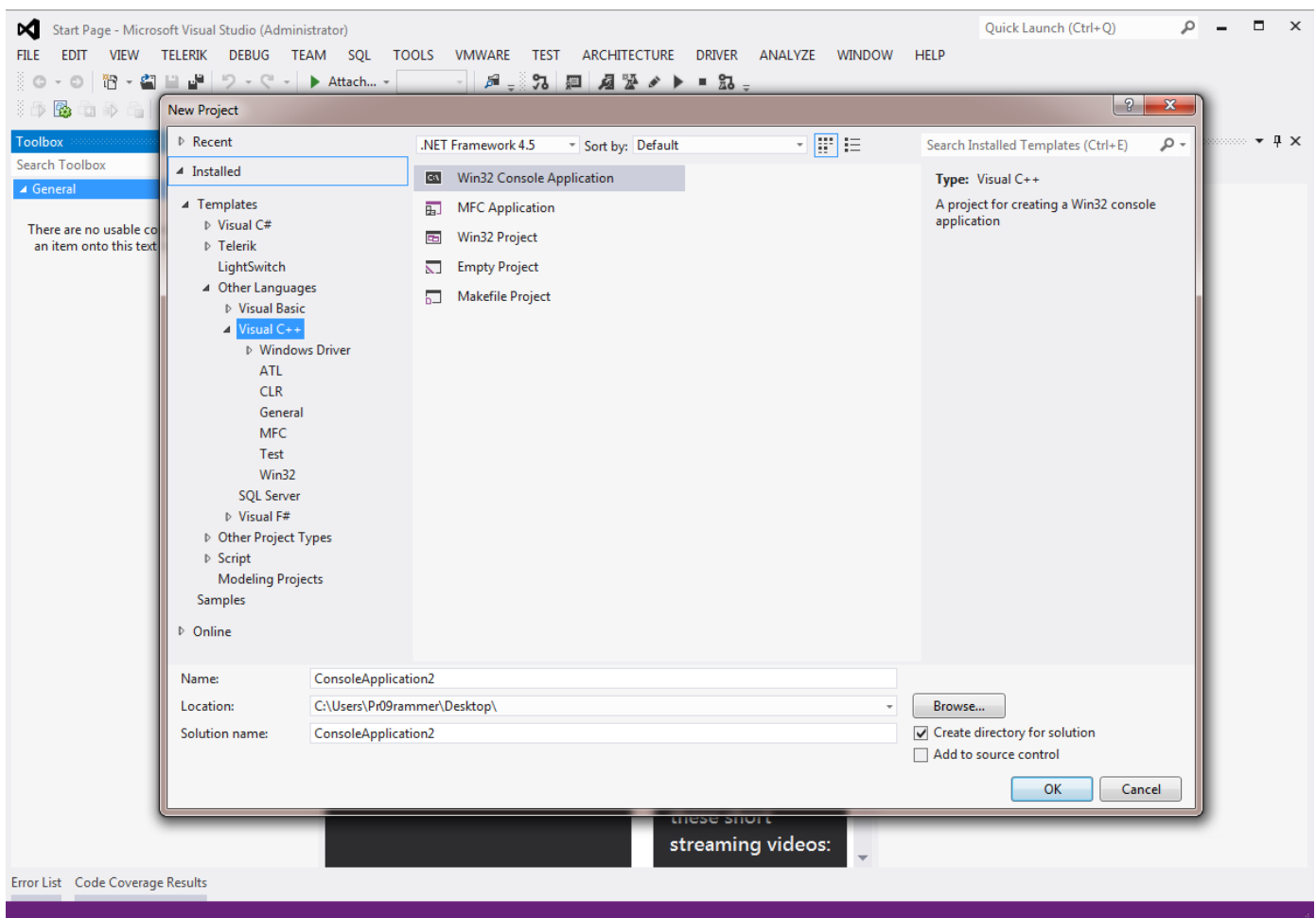
...

## اولین پروژه

معمولا برای شروع از تاریخچه و توضیحات دیگر استفاده میکنند اما روش آموزشی که در پیش خواهیم گرفت با انجام پروژه‌های عملی بوده و هر جا که نیاز به توضیح باشد، بیان میکنیم ...

ایجاد اولین پروژه Win32 Console Application

ویژوال استادیو را اجرا نمایید و از گزینه File -> New -> Project و سپس طبق عکس زیر پروژه Win32 Console Application را انتخاب نمایید، دقت کنید که زبان انتخاب شده Visual C++ باشد.



در این مرحله میتوانید محل ذخیره شده پروژه را در قسمت Location تنظیم نمایید و از قسمت Name میتوانید نام دلخواه را وارد کنید در حالت پیش فرض اگر اولین پروژه Win32 Console در مسیر تعیین شدهی قسمت Location باشد ، نام ConsoleApplication1 قرار گرفته است . پس از تنظیمات Ok کنید .

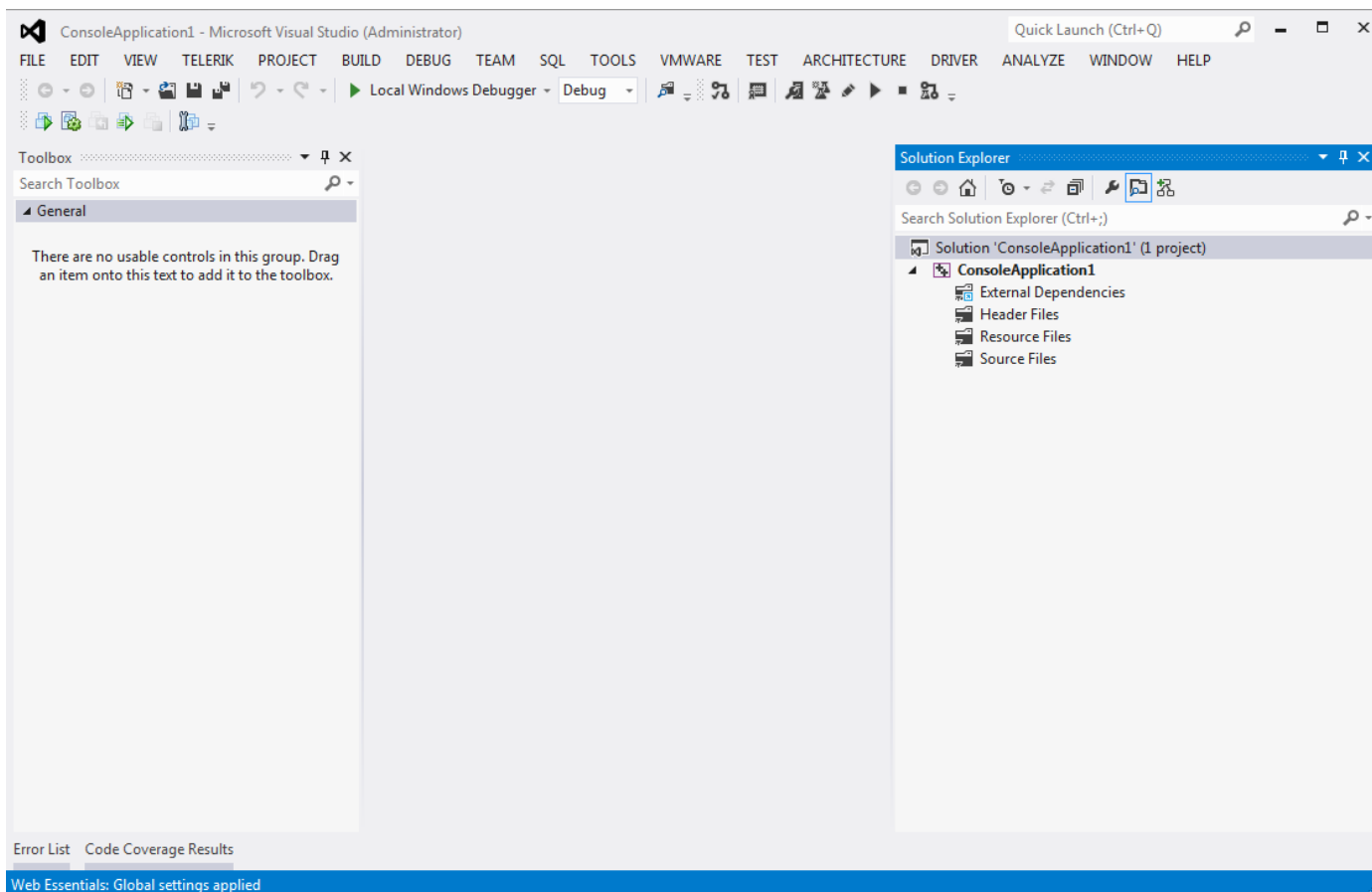


در این مرحله Next را بزنید .

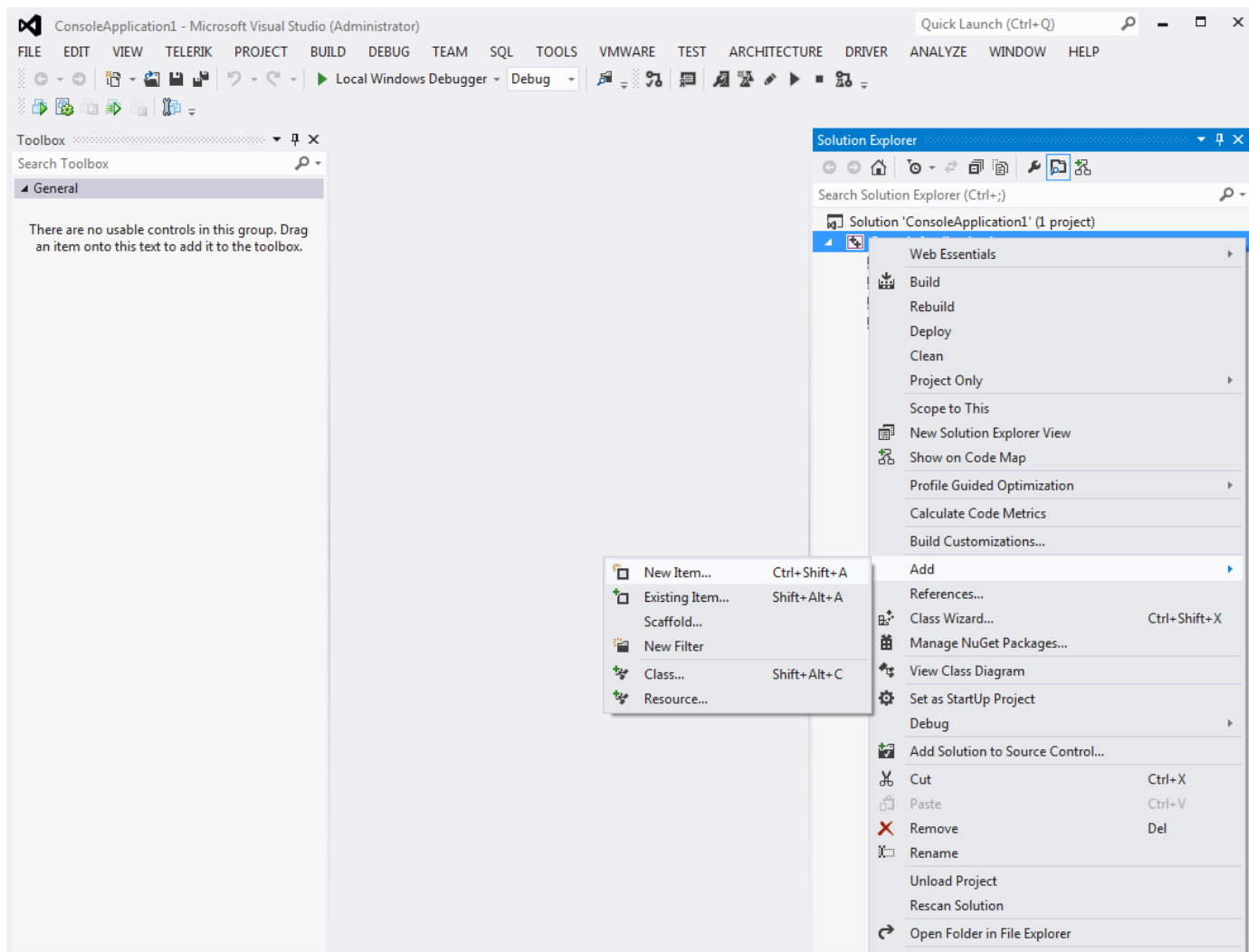


در این مرحله در قسمت Additional options تیک **Empty project** را بزنید ، همانند عکس فوق تنظیمات را انجام دهید .

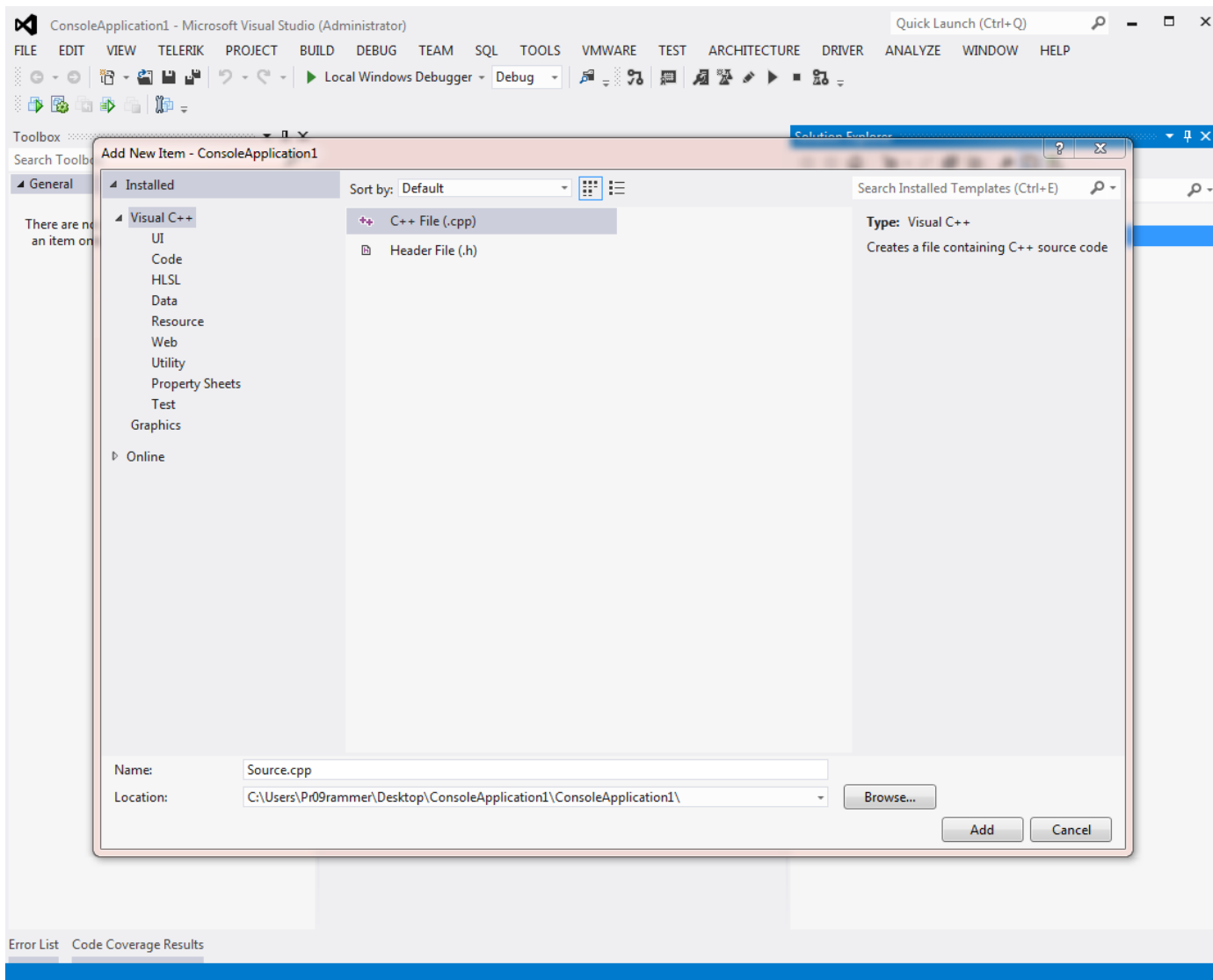
پس از انجام مراحل فوق پروژه بصورت شکل زیر ظاهر میشود .



برای کد نویسی روی نام پروژه که در اینجا ConsoleApplication1 می باشد ، راست کلیک میکنیم و گزینه Add و سپس New Item را انتخاب میکنیم .



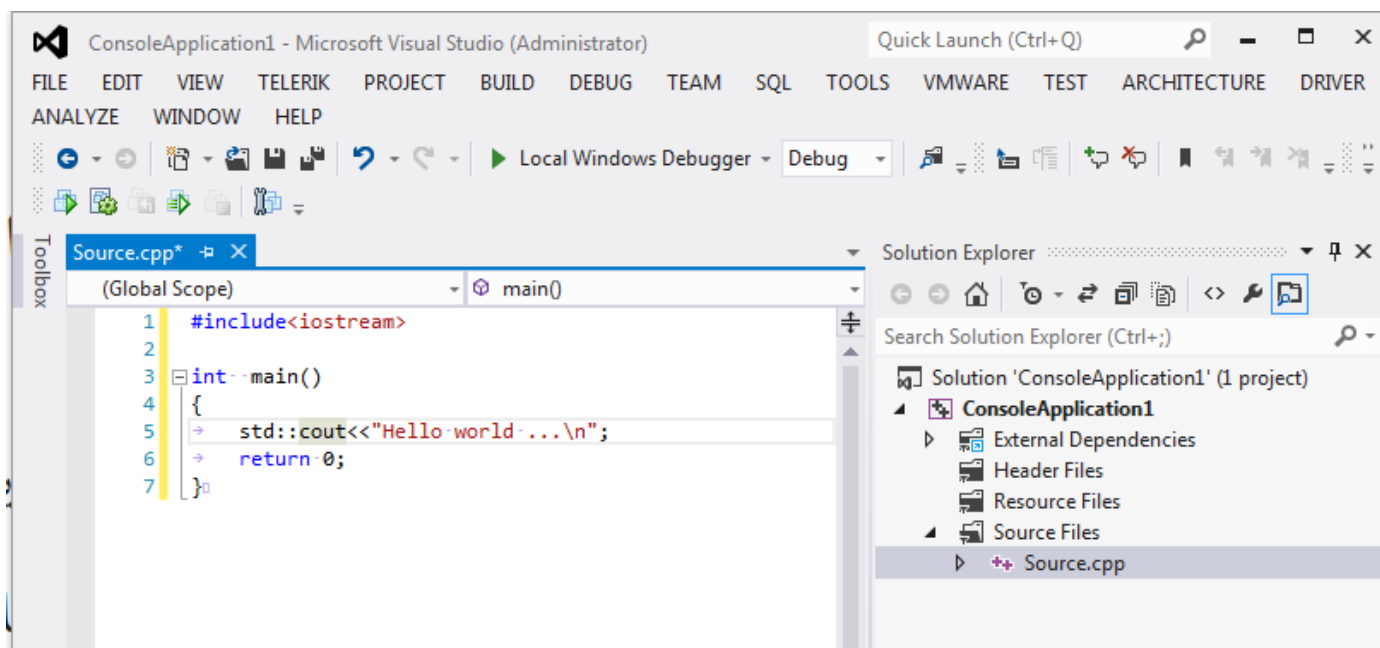
طبق عکس زیر فایل با پسوند cpp را انتخاب و Add میکنیم .



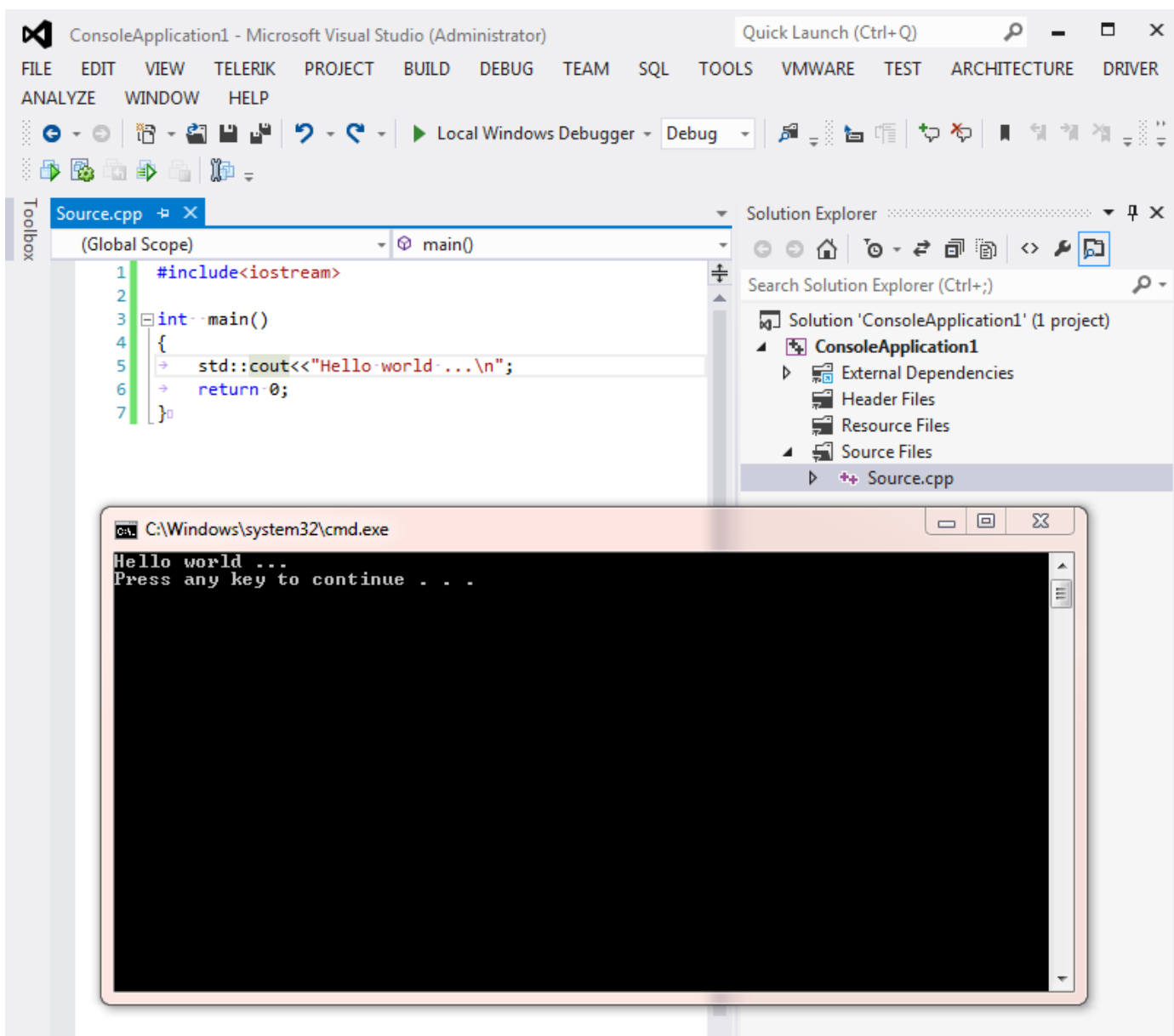
فایلی که اکنون به پروژه اضافه کردیم خالی و با نام پیش فرض Source.cpp می‌باشد ، دستورات زیر را در آن تایپ کنید . حال پروژه به شکل زیر خواهد بود .

```
#include<iostream>

int main()
{
    std::cout<<"Hello world ...\n";
    return 0;
}
```



برای اجرای پروژه کلید F5 را فشار دهید و اگر میخواهید نتیجه کار را مشاهده کنید کلید Ctrl + F5 را امتحان کنید .



شما اولین پروژه ++VC را اجرا نمودید ( آفرین ) .

### اما توضیحات :

خط اول برنامه یک راهنمای پیش پردازنده است ، کاراکتر # که نشان می‌دهد این خط یک راهنمای پیش پردازنده است و بعد عبارت `include` و نام یک فایل کتابخانه ای که بین علامت <> قرار داده شده ، فایل سرآیند استفاده شده در اینجا `iostream` میباشد . ( به فایل‌های کتابخانه ای ، فایل‌های سرآیند (Header Files) نیز گفته میشود . ) راهنمای پیش پردازنده خطی است که به کامپایلر اطلاع میدهد در برنامه موجودیتی است که تعریف آن را در فایل سرآیند مذکور جستجو کند . در این برنامه از `std::cout` استفاده شده ، که کامپایلر در مورد آن چیزی نمیداند لذا به فایل `iostream` مراجعه نموده ، تعریف آن را می‌یابد و آن را اجرا میکند .

خط 3 :

بخشی از هر برنامه تابع می‌باشد . پراتزهای واقع پس از آن `main` نشان می‌دهند که `main` یک بلوک برنامه بنام تابع است . برنامه‌ها می‌توانند حاوی یک یا چندین تابع باشند ، اما `main` تابع اصلی برنامه است که وجود آن الزامی میباشد . کلمه کلیدی `int` که



در سمت چپ **main** قرار گرفته، بر این نکته دلالت دارد که **main** یک مقدار صحیح برمی‌گرداند.

خط 5 :

با استفاده از این دستور رشته ای را به خروجی استاندارد که معمولاً صفحه نمایش باشد ارسال میکنیم .

خط 6 :

که `return 0` میباشد مقدار برگشتی تابع را مشخص میکند در حقیقت این خط که مقدار 0 را برمیگرداند نشان دهنده اتمام موفقیت آمیز برنامه می‌باشد .

به مرور زمان نسبت به موارد بالا بیشتر و مفصل صحبت خواهیم نمود .

## نظرات خوانندگان

نویسنده: علي

تاریخ: ۱۶:۱۸ ۱۳۹۱/۱۲/۲۶

دوست عزیز، بهتر نیست مباحث جدید C++0x رو مطرح کنید؟

نویسنده: حمیدرضا

تاریخ: ۱۷:۳۹ ۱۳۹۱/۱۲/۲۶

انشا... قسمت باشه حتما صحبت میکنیم ، ولی اگر به تیر موضوع دقت کنید هدف آموزش از مقدماتی تا پیشرفته است که اگر خدا یاری کند ادامه میدم و در خدمت اساتید و دوستان خواهم بود .  
در مورد مبحثی که شما فرمودین [در اینجا](#) کامل توضیح داده شده است .  
مباحث جدید حتما جزئی از آموزش خواهند بود .

نویسنده: CSHARPDOOST

تاریخ: ۱۵:۱۴ ۱۳۹۲/۰۱/۱۷

ممنون . خوبه بی زحمت ادامه بدید.