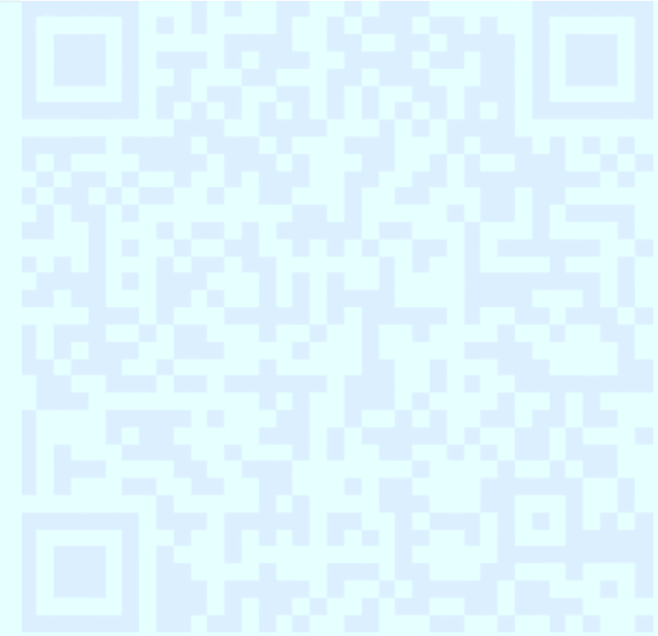


CUDA 프로그래밍

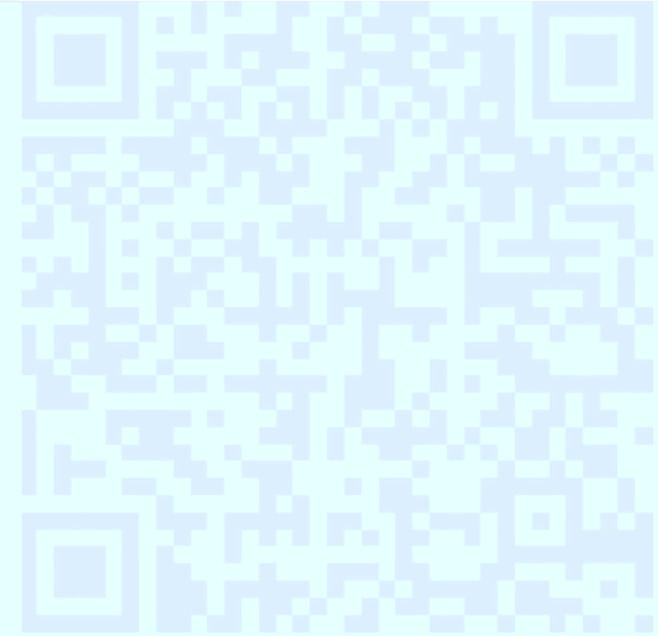
CUDA Programming
An Massively Parallel Computing Approach

biztripcru@gmail.com

© 2021-2022. biztripcru@gmail.com. All rights reserved.
모든 저작권은 biztripcru@gmail.com 에게 있습니다.



Hello, CUDA!



본 동영상과, 본 동영상 촬영에 사용된 발표 자료는 저작권법의 보호를 받습니다.

본 동영상과 발표 자료는 공개/공유/복제/상업적 이용 등, **개인 수강 이외의 다른 목적으로 사용하지 못합니다.**

© 2021-2022. biztripcru@gmail.com. All rights reserved.
모든 저작권은 biztripcru@gmail.com 에게 있습니다.

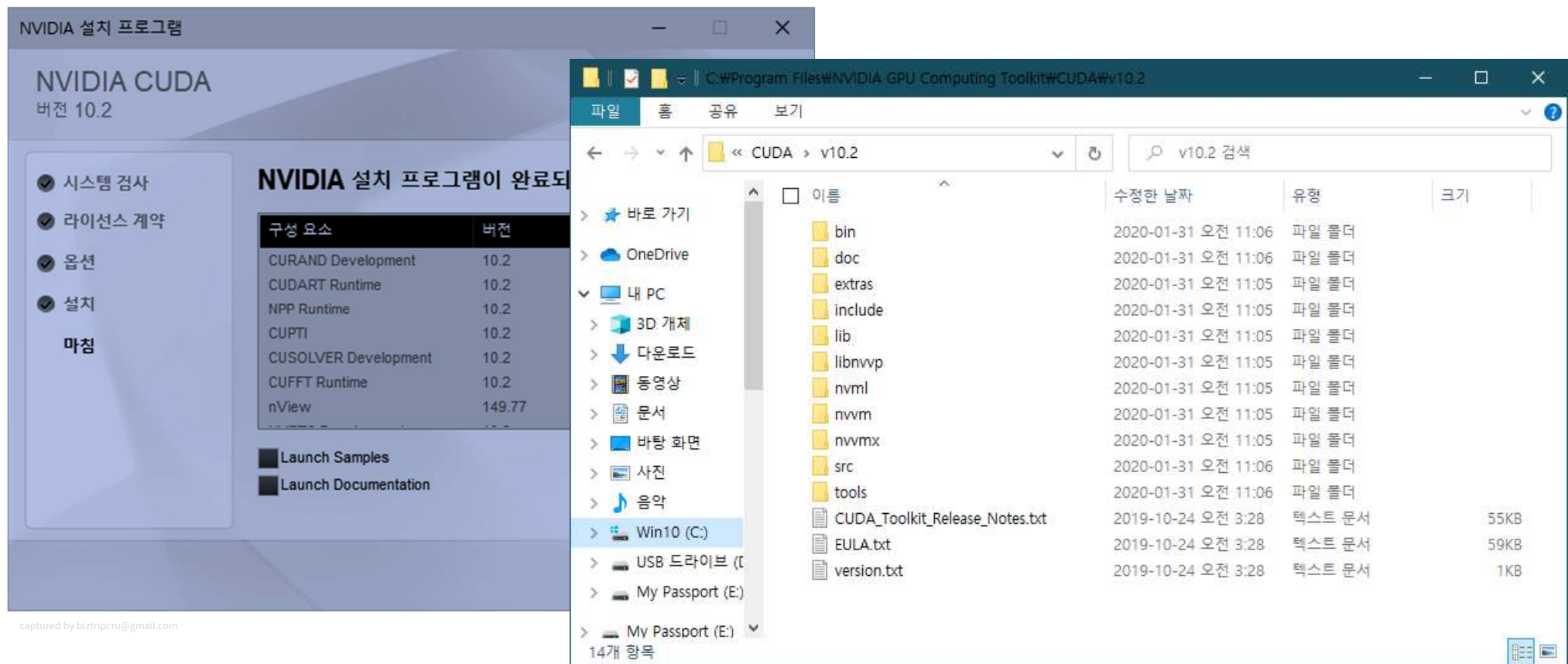
내용 contents

- 첫번째 CUDA 프로그램을 작성한다
 - C++: hello world
 - CUDA: hello CUDA !
 - 병렬 컴퓨팅 - 동시 수행
- 실습:
 - hello-cuda 프로젝트 만들기
 - hello-cuda.cu 작성/컴파일/실행

CUDA 툴킷^{toolkit} 설치^{install}

- **CUDA Toolkit version X.Y**

- C:/Program Files/NVIDIA GPU Computing Toolkit/CUDA/vX.Y/



CUDA 데모demo_suite

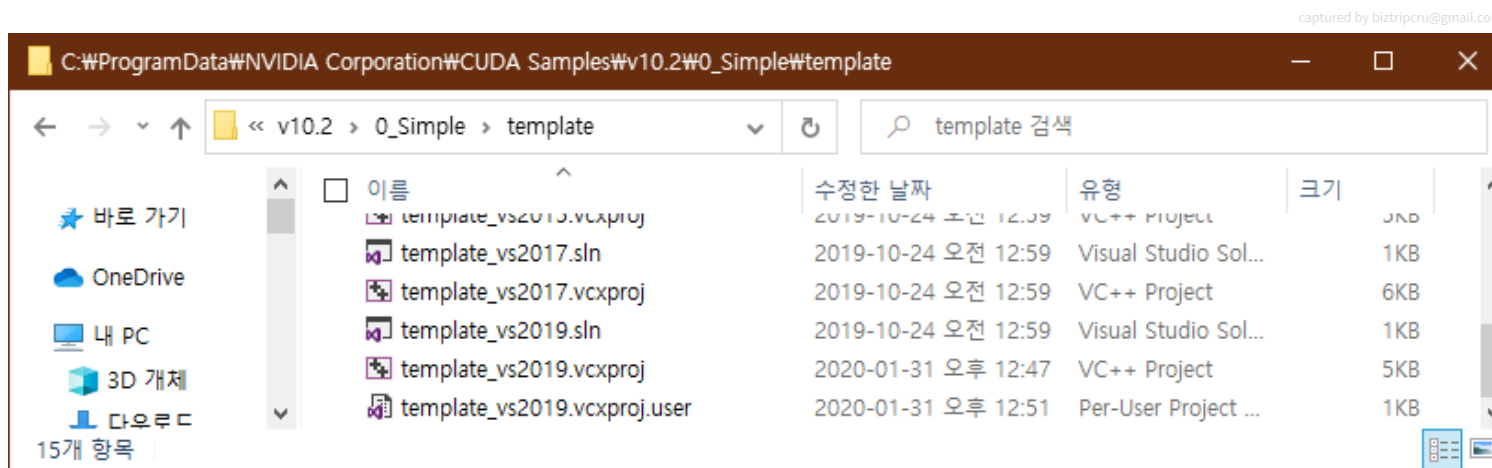
- CUDA "n-body" demonstrations, from “samples” folder:
 - .../CUDA/vX.Y/5_Simulations/nbody → nbody_vs2019.sln 선택, ...
 - 주의: “Release” mode로 compile !

captured by biztripcru@gmail.com



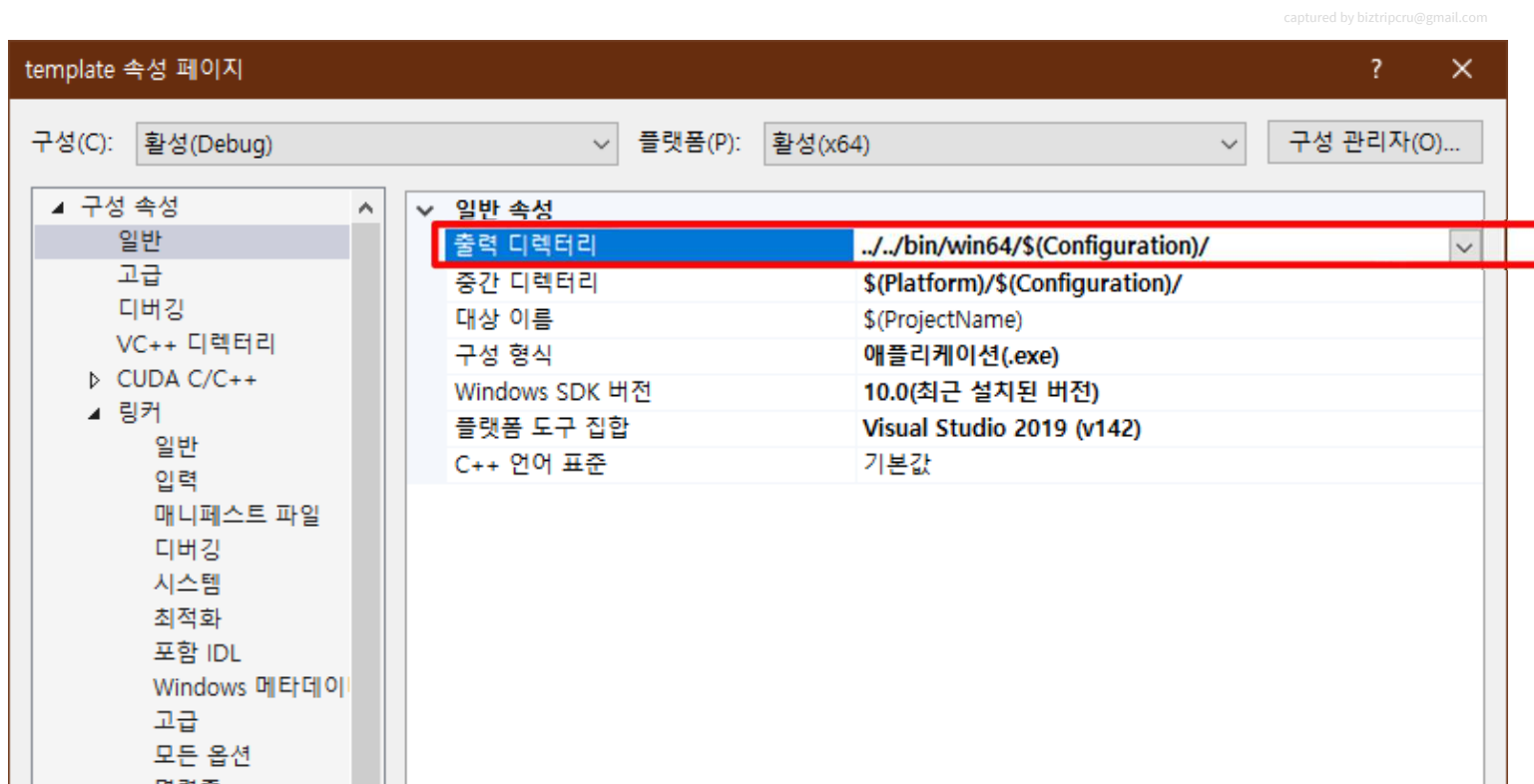
CUDA 샘플 프로젝트 재사용

- CUDA 프로그램 작성 → VS 프로젝트 파일이 필요
- 방법 1: CUDA 샘플의 템플릿^{template} 프로젝트를 재사용
 - C:/ProgramData/NVIDIA Corporation/CUDA Samples/vX.Y/0_Simple/template/
 - 별도 폴더로 복사, VS 에서 편집해서 사용



CUDA 샘플 프로젝트 재사용 계속

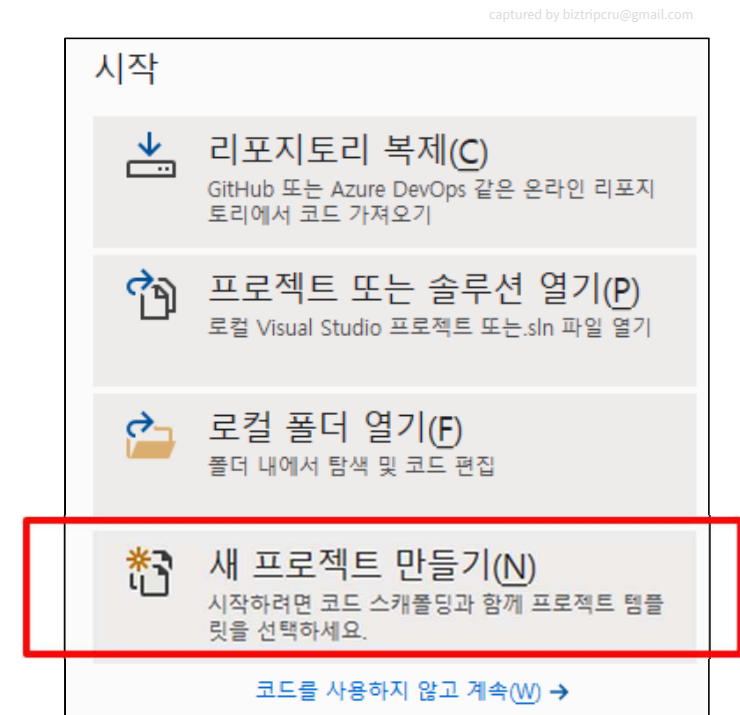
- 방법 1: CUDA 샘플의 템플릿^{template} 프로젝트를 재사용
 - 별도 폴더로 복사, VS 에서 편집해서 사용
 - 생각보다 바뀌어야 할 곳이 많음



CUDA 프로젝트 새로 만들기

- 방법 2: 처음부터 시작

- 비주얼 스튜디오 Visual Studio 시작
- 새 프로젝트 만들기 → CUDA X.Y Runtime 선택
- 파일 File → 새로만들기 New
 - 프로젝트 Project...
 - CUDA X.Y Runtime 선택



CUDA 프로젝트 새로 만들기 계속

captured by biztripcru@gmail.com

새 프로젝트 만들기

최근 프로젝트 템플릿(R)

 콘솔 앱 C++

템플릿 검색(Alt+S)(S)

모든 언어(L) 모든 플랫폼(P) 모든 프로젝트 형식(T)

 기존 Microsoft Cppunittest 스레드뷰를 사용하여 C++ 단위 테스트를 작성합니다.

C++ Windows 테스트

 Google Test

Google Test를 사용하여 C++ 단위 테스트를 작성하세요. 사용할 Google Test 라이브러리의 복사본을 포함합니다.

C++ Windows 테스트

 **CUDA 11.3 Runtime**

A project that uses the CUDA 11.3 runtime

C++ CUDA Windows Linux 클라우드 콘솔 DataScience

데스크톱 Machine Learning

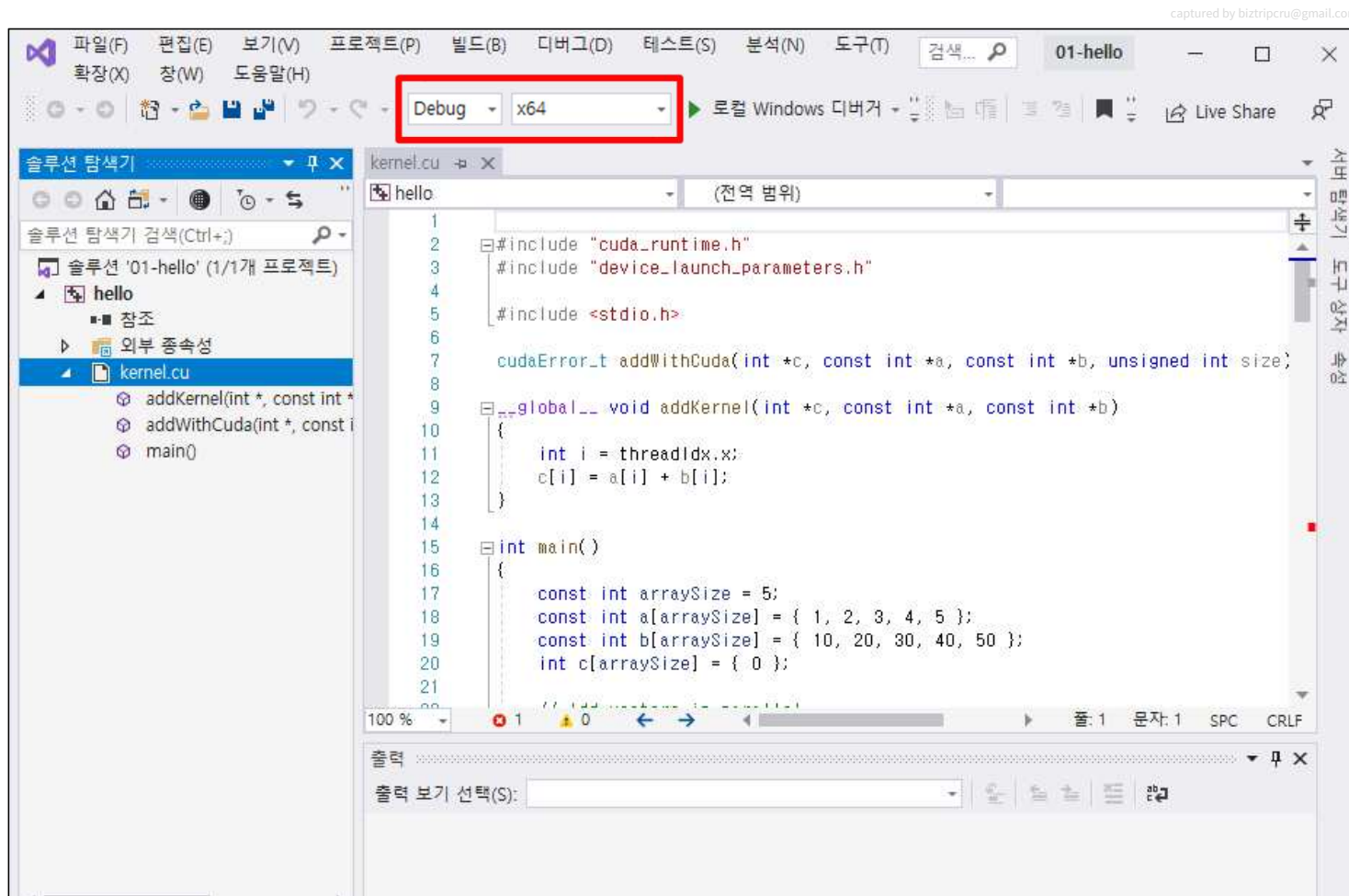
 내보내기가 있는 동적 연결 라이브러리(DLL)

실행 중인 여러 Windows 앱 간에 공유할 수 있는 .dll을 빌드하세요.

원하는 항목을 찾지 못하셨나요?
[추가 도구 및 기능 설치](#)

CUDA 프로그램 작성

- 예제 코드가 들어 있음



CUDA 프로그램 작성 계속

• 코드 지우고, 새로 작성

captured by biztripcru@gmail.com

The screenshot shows the Microsoft Visual Studio IDE with a CUDA program being debugged. The code in `hello.cu` is as follows:

```
1 #include <stdio>
2
3 global void hello(void) {
4     printf("hello, CUDA!\n");
5 }
6
7 int main(void) {
8     hello<<<1,1>>>();
9     return 0;
10 }
```

The debug console shows the output of the program:

```
hello, CUDA!
C:\Work\200-CUDA\Programs\01-hello\x64\Debug\hello.exe(프로세스 2421
2개)이(가) 종료되었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디
버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용하도록 설정합니
다.
이 창을 닫으려면 아무 키나 누르세요...
```

The output window shows the following messages:

```
출력 보기 선택(S): 디버그
0x640c 스레드가 종료되었습니다(코드: 0 (0x0)).
0x233c 스레드가 종료되었습니다(코드: 0 (0x0)).
0x63fc 스레드가 종료되었습니다(코드: 0 (0x0)).
0x60e4 스레드가 종료되었습니다(코드: 0 (0x0)).
0x63b4 스레드가 종료되었습니다(코드: 0 (0x0)).
'[24212] hello.exe' 프로그램이 종료되었습니다(코드: 0 (0x0)).
```

cu 확장자 extension

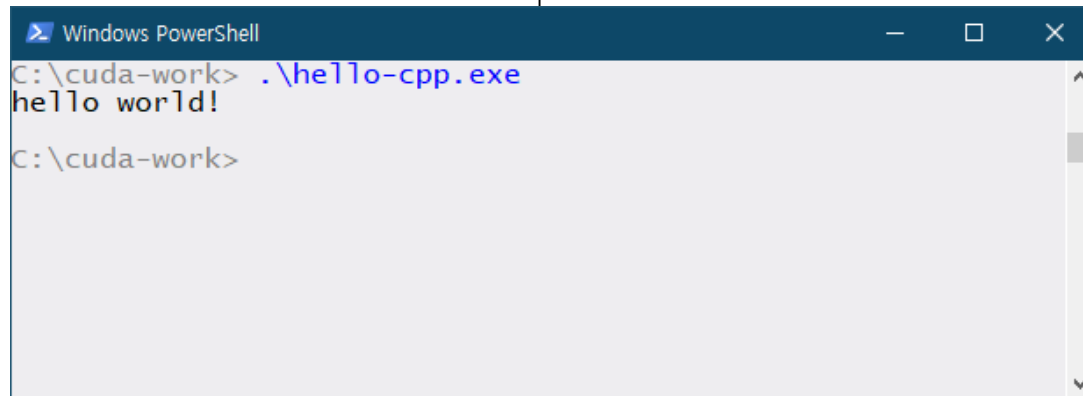
- **.cu** : CUDA 프로그램의 확장자 extension
- **.cu file = .cpp file (C/C++) + CUDA 커널 소스 코드 kernel source code**
 - CUDA kernel : **__global__** 설정자 specifier 사용
- **hello-cpp.cu** : C/C++ source code만 있어도 되는가?
 - 가능 !
 - **CUDA kernel** 도 포함시킬 수 있다는 의미이지, 꼭 필요한 것은 아님

hello-cpp.cu

```
#include <stdio.h>

void hello( void ) {
    printf("hello world!\n");
}

int main( void ) {
    hello();
    return 0;
}
```

A screenshot of a Windows PowerShell window. The title bar is dark blue with the text 'Windows PowerShell' and standard window controls. The command prompt shows the directory 'C:\cuda-work' and the command '.\hello-cpp.exe' being executed. The output of the program is 'hello world!'.

```
Windows PowerShell
C:\cuda-work> .\hello-cpp.exe
hello world!
C:\cuda-work>
```

프로그래밍 팁 programming tip

- Windows 에서, printf() 출력이 없는 경우가 있음
- fflush() 로 해결 가능
 - printf() → stdio buffers → output
 - fflush : flush stdio buffers
 - ▶ stdio : standard input/output, for C/C++

```
#include <stdio.h>
int main(void) {
    printf( "Hello, world!\n" );
    fflush( stdout );
}
```

hello-cuda.cu

- hello-cpp.cu

```
#include <stdio.h>
```

```
void hello( void ) {  
    printf("hello world!\n");  
}
```

```
int main( void ) {  
    hello();  
    return 0;  
}
```

- hello-cuda.cu

```
#include <stdio.h>
```

```
__global__ void hello( void ) {  
    printf( "hello CUDA!\n" );  
}
```

```
int main( void ) {  
    hello<<<1,1>>>();  
    fflush( stdout );  
    return 0;  
}
```

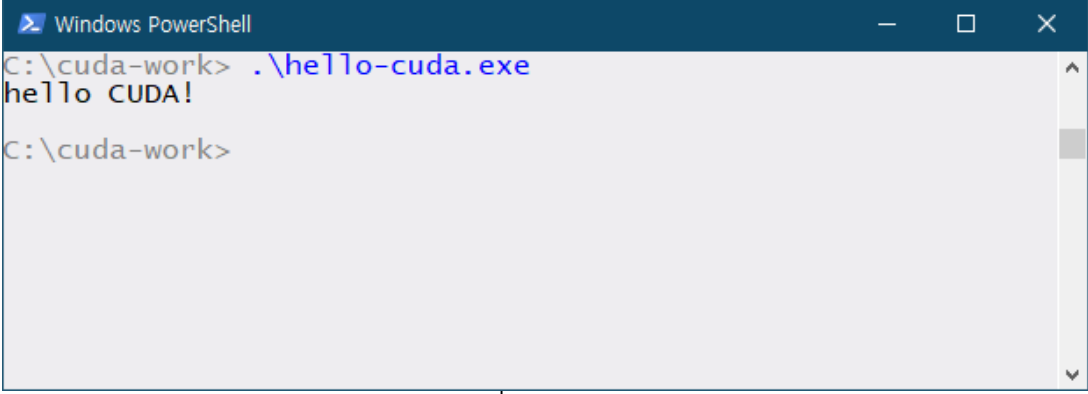
hello-cuda.cu

```
#include <stdio.h>

__global__ void hello( void ) {
    printf( "hello CUDA!\n" );
}

int main( void ) {
    hello<<<1,1>>>>();
    fflush( stdout );
    return 0;
}
```

- `__global__`
 - **CUDA 함수** 임을 표시
- `<<<1,1>>>`
 - $1 \times 1 = 1$ 개 core만 사용
- 주의: **Linux**에서는 실패할 수 있음



```
Windows PowerShell
C:\cuda-work> .\hello-cuda.exe
hello CUDA!
C:\cuda-work>
```


hello-parallel.cu

```
#include <stdio.h>

__global__ void hello( void ) {
    printf( "hello CUDA %d !\n", threadIdx.x );
}

int main( void ) {
    hello<<<1,8>>>>();
    fflush( stdout );
    return 0;
}
```

- <<<1,8>>>
 - 1 x 8 = 8개 core
- 함수를 8번 동시 실행
 - 병렬 컴퓨팅 parallel computing
- 주의: Linux에서는 실패할 수 있음



```
Windows PowerShell
C:\cuda-work> .\04c-hello-parallel.exe
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 2 !
hello CUDA 3 !
hello CUDA 4 !
hello CUDA 5 !
hello CUDA 6 !
hello CUDA 7 !
C:\cuda-work>
```

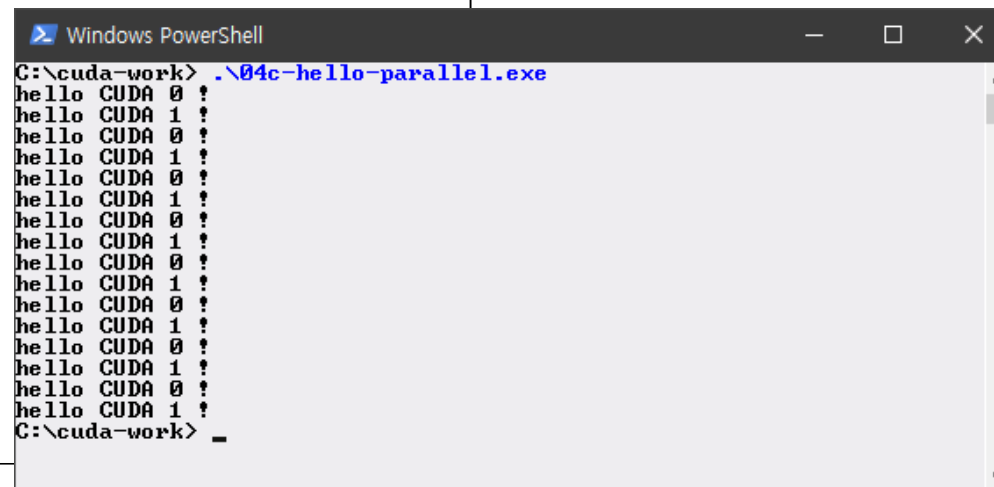
hello-parallel.cu 계속

```
#include <stdio.h>

__global__ void hello( void ) {
    printf( "hello CUDA %d !\n", threadIdx.x );
}

int main( void ) {
    hello<<<8,2>>>>();
    fflush( stdout );
    return 0;
}
```

- <<<8,2>>>
 - $8 \times 2 = 16$ 번 (16개의 core 사용)
- 함수를 8세트, 2번씩 동시 실행
 - 병렬 컴퓨팅 parallel computing
- 주의: 숫자가 다르게 나옴...



```
Windows PowerShell
C:\cuda-work> .\04c-hello-parallel.exe
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 0 !
hello CUDA 1 !
C:\cuda-work> _
```

hello-linux.cu

```
#include <stdio.h>

__global__ void hello( void ) {
    printf( "hello CUDA %d !\n", threadIdx.x );
}

int main( void ) {
    hello<<<1,8>>>>();
    #if defined(__linux__)
        cudaDeviceSynchronize();
    #endif
    fflush( stdout );
    return 0;
}
```

- **Linux 용으로 수정한 version**
 - 추가 설명은 다른 강의 시간에 ...

```
linux/cuda-work > make 04d-hello-linux.exe
nvcc -gencode=arch=compute_75,code=\"sm_75,compute_75\" -arch=sm_75 -O2 -o 04d-hello-linux.exe
linux/cuda-work > ./04d-hello-linux.exe
hello CUDA 0 !
hello CUDA 1 !
hello CUDA 2 !
hello CUDA 3 !
hello CUDA 4 !
hello CUDA 5 !
hello CUDA 6 !
hello CUDA 7 !
linux/cuda-work > █
```

C++ stream은 실패

```
#include <iostream>
using namespace std;

__global__ void hello( void ) {
    std::cout << "hello world!" << std::endl;
}

int main( void ) {
    hello<<<1,1>>>();
    return 0;
}
```

● 컴파일 결과

...

error: identifier "std::cout" is undefined in device code

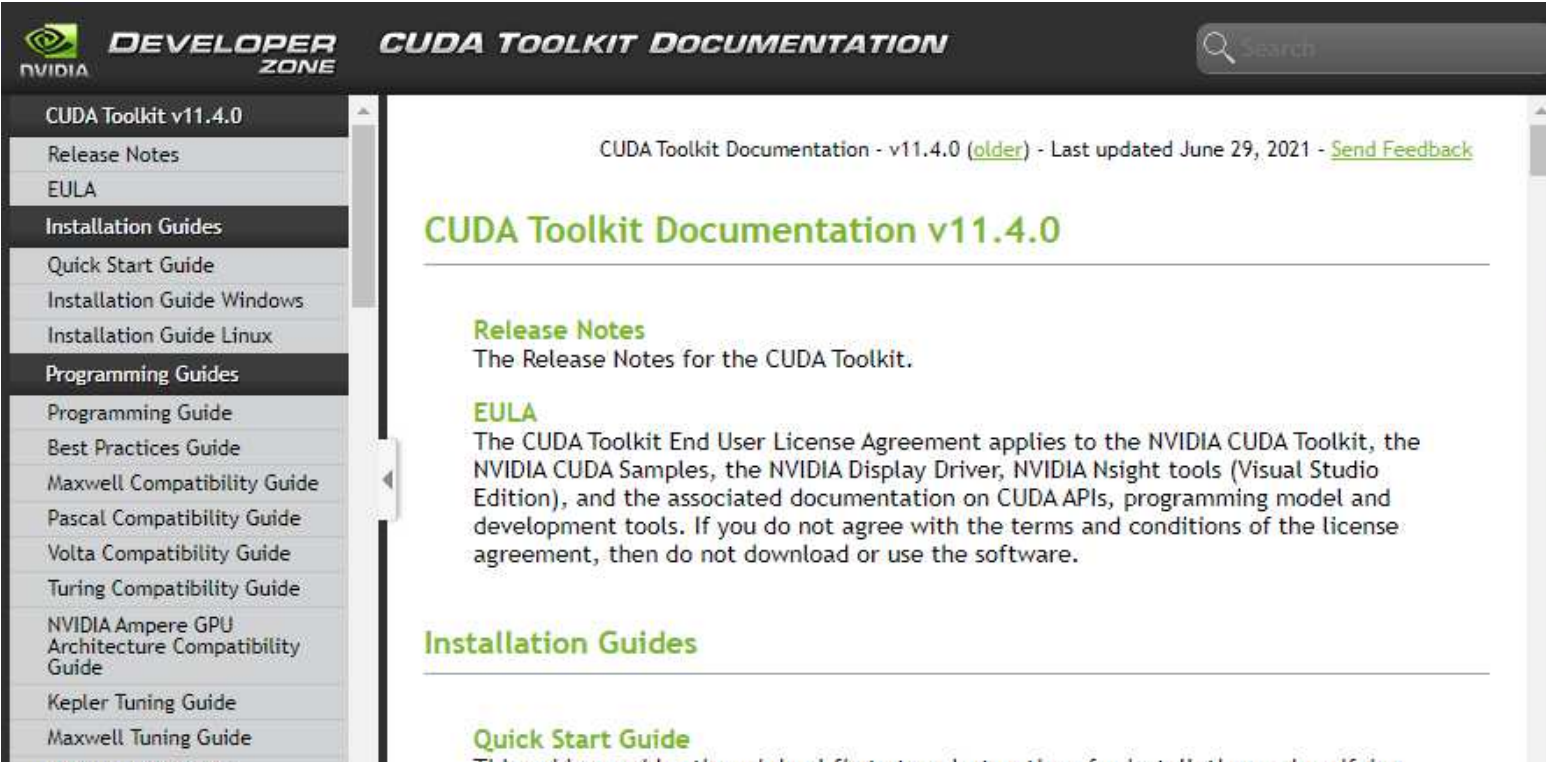
...

CUDA 관련 자료

- **CUDA API reference:**

- <http://docs.nvidia.com/cuda/index.html>
- <http://docs.nvidia.com/cuda/cuda-runtime-api/index.html>

captured by biztripcu@gmail.com



The screenshot shows the NVIDIA Developer Zone website for the CUDA Toolkit Documentation, version v11.4.0. The page has a dark header with the NVIDIA logo, 'DEVELOPER ZONE', and 'CUDA TOOLKIT DOCUMENTATION'. A search bar is located in the top right. A left sidebar contains a navigation menu with categories like 'CUDA Toolkit v11.4.0', 'Release Notes', 'EULA', 'Installation Guides', 'Programming Guides', and various compatibility guides. The main content area displays the title 'CUDA Toolkit Documentation v11.4.0' and includes sections for 'Release Notes', 'EULA', 'Installation Guides', and 'Quick Start Guide'. The 'EULA' section is currently expanded, showing the text of the End User License Agreement.

CUDA Toolkit v11.4.0

- Release Notes
- EULA
- Installation Guides
 - Quick Start Guide
 - Installation Guide Windows
 - Installation Guide Linux
- Programming Guides
 - Programming Guide
 - Best Practices Guide
 - Maxwell Compatibility Guide
 - Pascal Compatibility Guide
 - Volta Compatibility Guide
 - Turing Compatibility Guide
 - NVIDIA Ampere GPU Architecture Compatibility Guide
 - Kepler Tuning Guide
 - Maxwell Tuning Guide

CUDA Toolkit Documentation - v11.4.0 (older) - Last updated June 29, 2021 - [Send Feedback](#)

CUDA Toolkit Documentation v11.4.0

Release Notes

The Release Notes for the CUDA Toolkit.

EULA

The CUDA Toolkit End User License Agreement applies to the NVIDIA CUDA Toolkit, the NVIDIA CUDA Samples, the NVIDIA Display Driver, NVIDIA Nsight tools (Visual Studio Edition), and the associated documentation on CUDA APIs, programming model and development tools. If you do not agree with the terms and conditions of the license agreement, then do not download or use the software.

Installation Guides

Quick Start Guide

This guide provides the minimal first steps instructions for installation and verification.

CUDA 관련 자료 계속

- **CUDA course:**

- <https://developer.nvidia.com/educators/existing-courses>

captured by biztripcru@gmail.com

[Home](#) > [Higher Education & Research](#) > [Education Resources & Teaching Programs](#) > [Educators Resources](#) > [Existing University Courses](#)

Existing University Courses

This page has online courses to help you get started programming or teaching CUDA as well as links to Universities teaching CUDA.

This page organized into three sections to get you started

- [Introductory CUDA Technical Training Courses](#)
- [CUDA University Courses](#)
- [CUDA Seminars and Tutorials](#)

Introductory CUDA Technical Training Courses

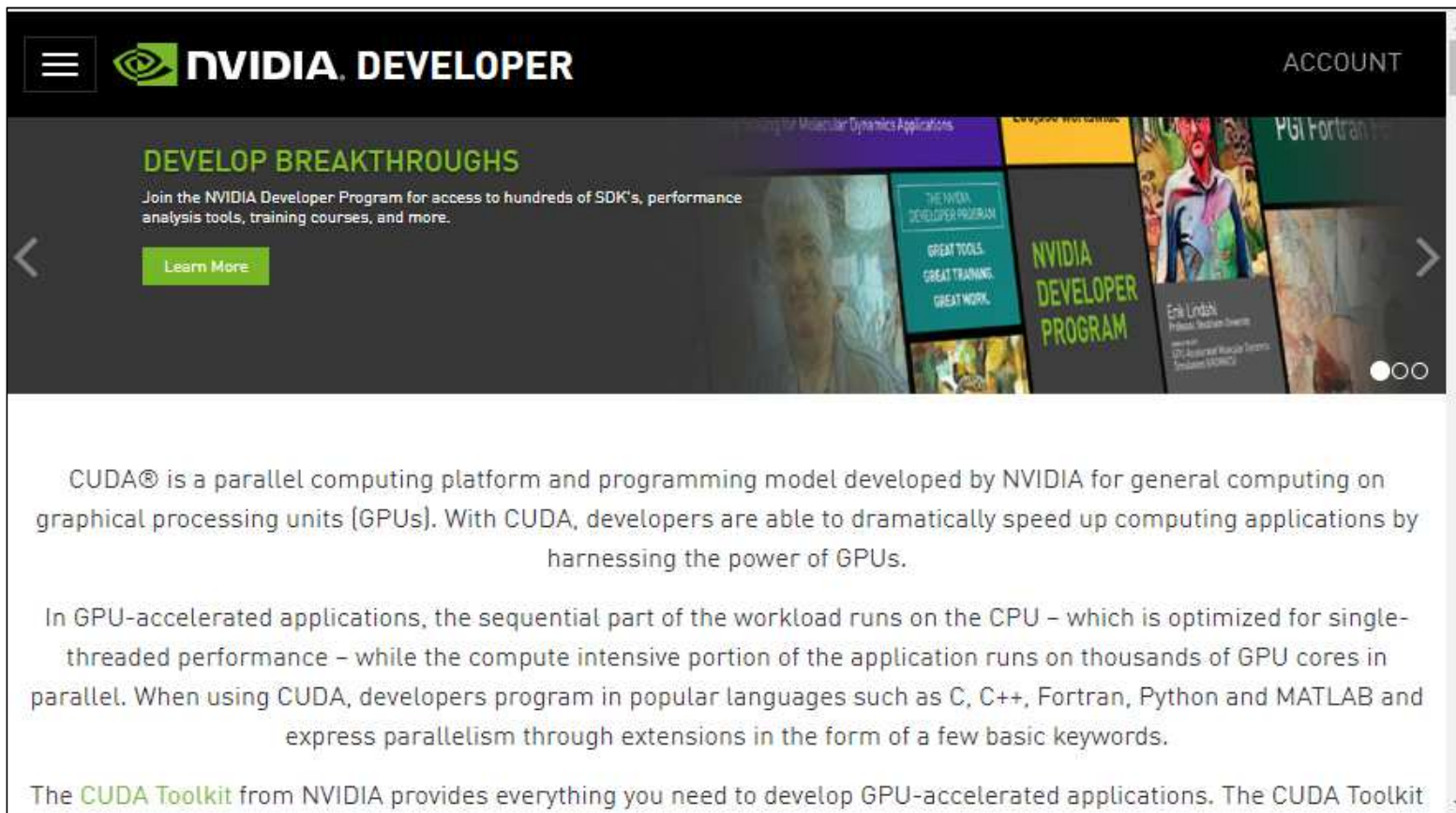
- [Udacity: CS344 Intro To Parallel Programming](#)
- [Volume I: Introduction to CUDA Programming](#)
 - [Exercises](#) (for Linux and Mac)
 - [Visual Studio Exercises](#) (for Windows)
 - [Instructions for Exercises](#)
- [Volume II: CUDA Case Studies](#)

CUDA 관련 자료 계속

- **CUDA Zone**

- <https://developer.nvidia.com/cuda-zone>

captured by biztripcru@gmail.com



The screenshot shows the NVIDIA Developer website. At the top, there is a navigation bar with the NVIDIA logo, the text "NVIDIA DEVELOPER", and an "ACCOUNT" link. Below the navigation bar, there is a large banner with the text "DEVELOP BREAKTHROUGHS" and a sub-header "Join the NVIDIA Developer Program for access to hundreds of SDK's, performance analysis tools, training courses, and more." A "Learn More" button is visible. The banner also features a collage of images, including a portrait of a woman and text about the "NVIDIA DEVELOPER PROGRAM". Below the banner, there is a paragraph of text about CUDA, followed by a paragraph about GPU-accelerated applications, and a final paragraph about the CUDA Toolkit.

DEVELOP BREAKTHROUGHS
Join the NVIDIA Developer Program for access to hundreds of SDK's, performance analysis tools, training courses, and more.
[Learn More](#)

CUDA® is a parallel computing platform and programming model developed by NVIDIA for general computing on graphical processing units (GPUs). With CUDA, developers are able to dramatically speed up computing applications by harnessing the power of GPUs.

In GPU-accelerated applications, the sequential part of the workload runs on the CPU – which is optimized for single-threaded performance – while the compute intensive portion of the application runs on thousands of GPU cores in parallel. When using CUDA, developers program in popular languages such as C, C++, Fortran, Python and MATLAB and express parallelism through extensions in the form of a few basic keywords.

The **CUDA Toolkit** from NVIDIA provides everything you need to develop GPU-accelerated applications. The CUDA Toolkit

내용 contents

- 첫번째 CUDA 프로그램을 작성한다
 - C++: hello world
 - CUDA: hello CUDA !
 - 병렬 컴퓨팅 - 동시 수행
- 실습:
 - hello-cuda 프로젝트 만들기
 - hello-cuda.cu 작성/컴파일/실행

Hello, CUDA!



폰트 끝단 일치 → 큰 교자 타고 혼례 치른 날
정참판 양반댁 규수 큰 교자 타고 혼례 치른 날
정참판 양반댁 규수 큰 교자 타고 혼례 치른 날
본고딕 Noto Sans KR

© 2021-2022. biztripcru@gmail.com. All rights reserved.
모든 저작권은 biztripcru@gmail.com 에게 있습니다.

The quick brown fox jumps over the lazy dog
The quick brown fox jumps over the lazy dog
The quick brown fox jumps over the lazy dog
Source Sans Pro

Mathematical Notations $O(n \log n)$
Source Serif Pro