oneAPI DevSummit hands-on

contents

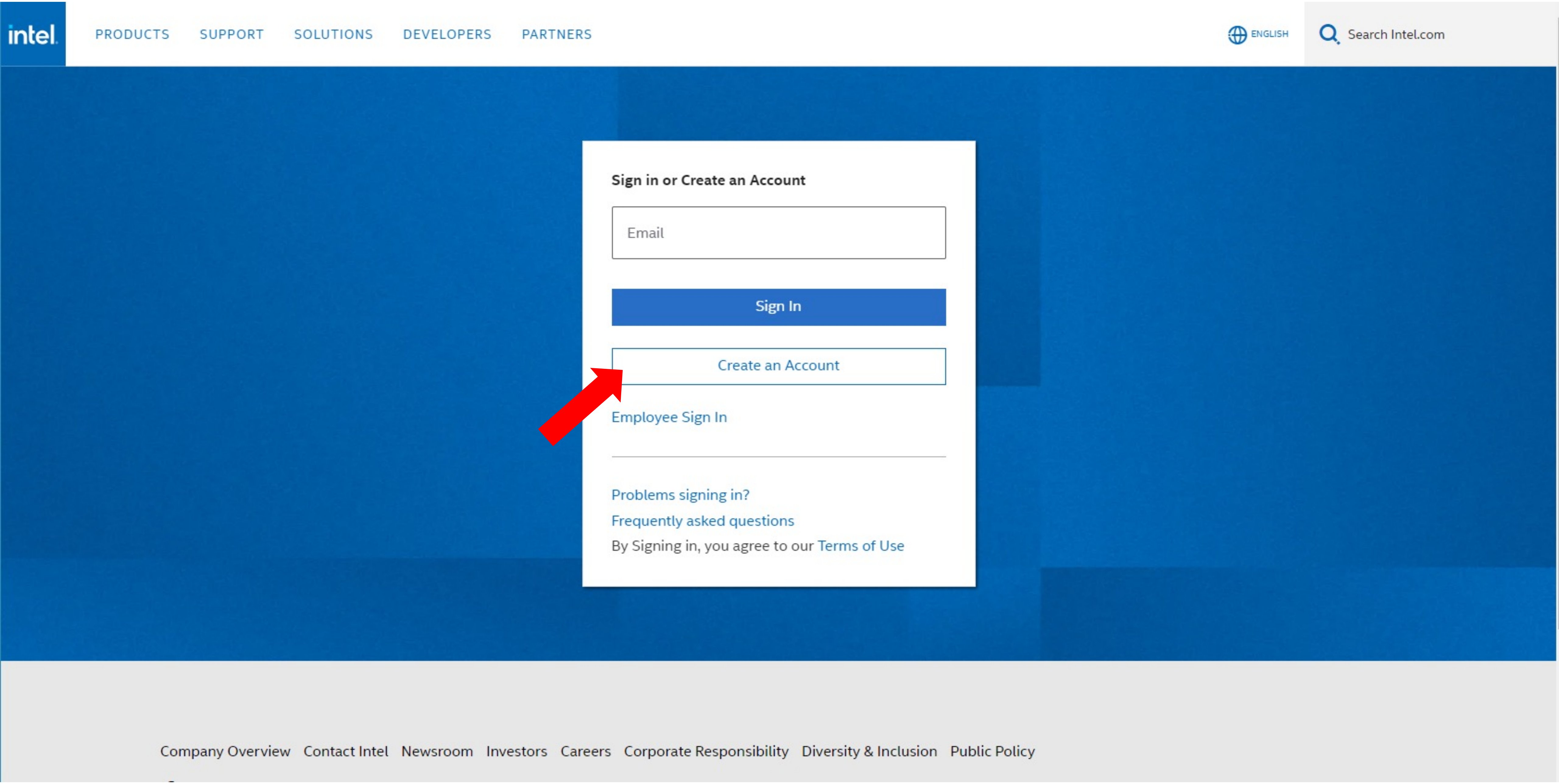
- Intel Devcloud 접속
- CUDA 코드를 SYCL로 변환하기
- Vtune 사용하여 profiling 및 oneMKL를 활용한 최적화

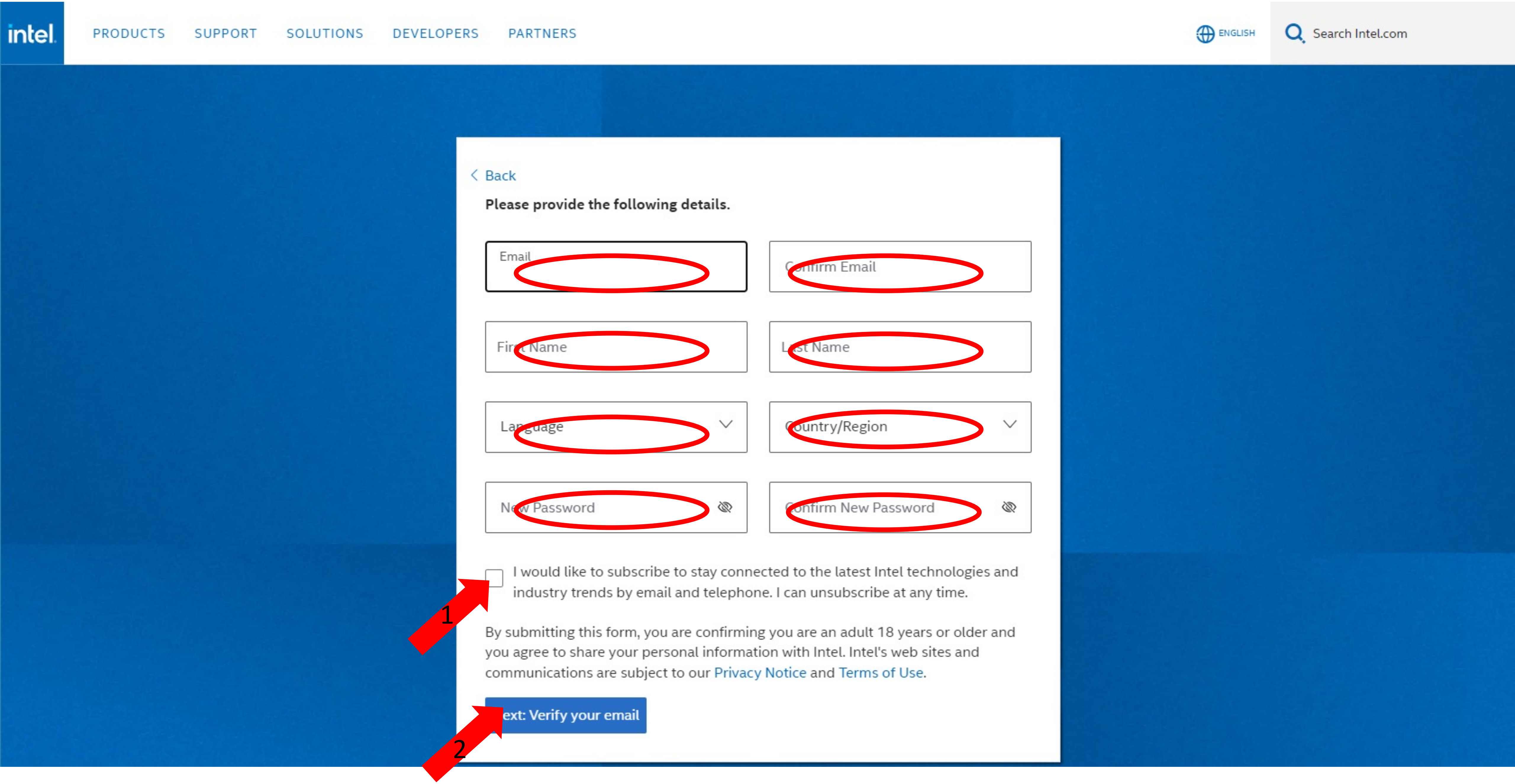
- WSL 설치 (참고)
  https://learn.microsoft.com/en-us/windows/wsl/install

- WSL의 terminal 뿐만 아니라
  windows command prompt나 powershell 에서도 접속이 가능합니다.

Intel® Developer Cloud Guide

- Intel® Developer Cloud site
  https://console.cloud.intel.com/GetStartedDevCloud


intel
PRODUCTS
SUPPORT
SOLUTIONS
DEVELOPERS
PARTNERS
ENGLISH
Search Intel.com
Sign in or Create an Account
Email
Sign In
Create an Account
Employee Sign In
Problems signing in?
Frequently asked questions
By Signing in, you agree to our Terms of Use
Company Overview
Contact Intel
Newsroom
Investors
Careers
Corporate Responsibility
Diversity & Inclusion
Public Policy


intel.
PRODUCTS
SUPPORT
SOLUTIONS
DEVELOPERS
PARTNERS
ENGLISH
Search Intel.com
Back
Please provide the following details.
Email
Confirm Email
First Name
Last Name
Language
Country/Region
New Password
Confirm New Password
I would like to subscribe to stay connected to the latest Intel technologies and industry trends by email and telephone. I can unsubscribe at any time.
By submitting this form, you are confirming you are an adult 18 years or older and you agree to share your personal information with Intel. Intel's web sites and communications are subject to our Privacy Notice and Terms of Use.
ext: Verify your email
1
2

인텔 계정 이메일 확인 코드 받은편지함 ×

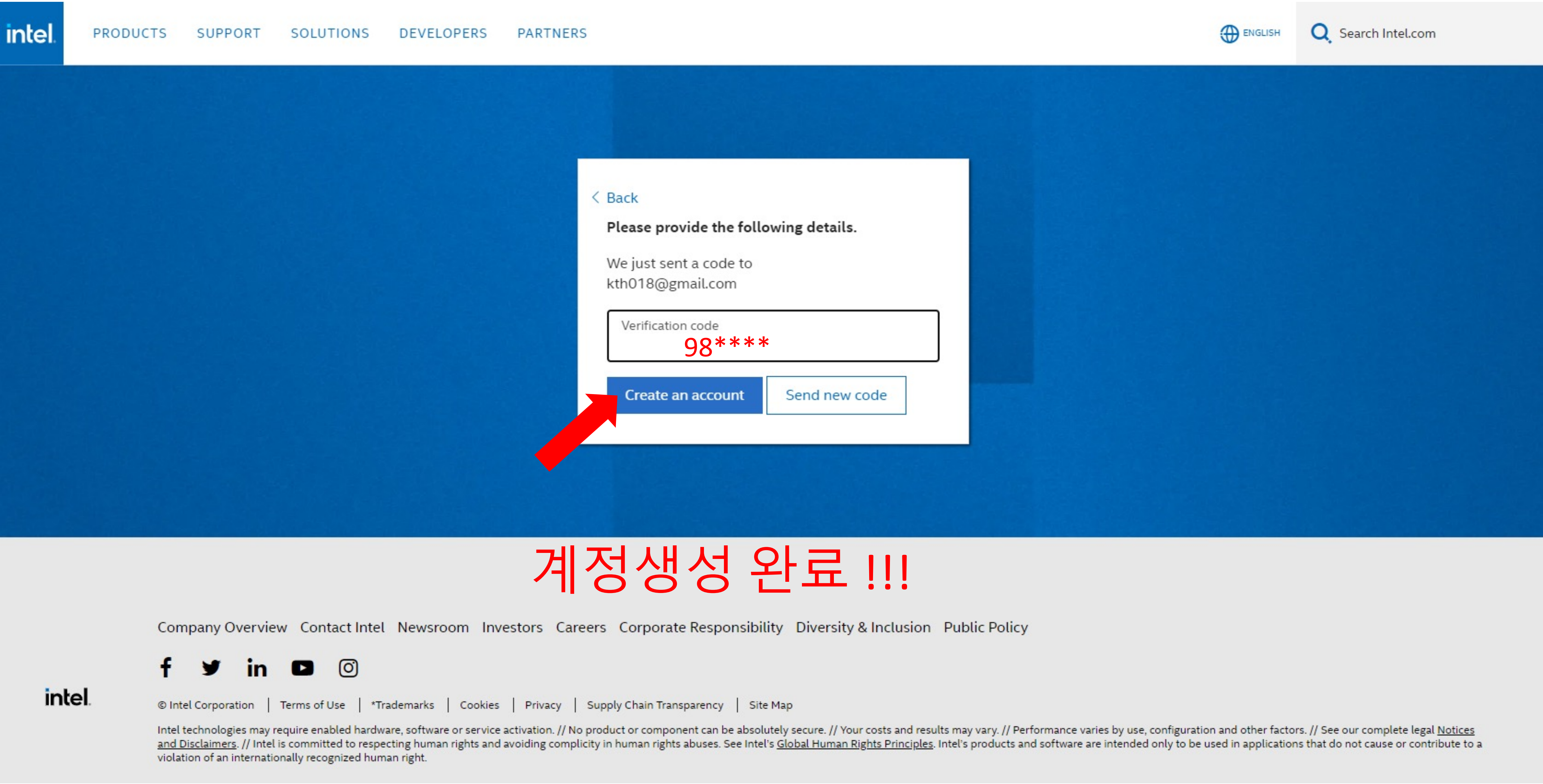
noreply@consumer.intel.com
나에게

이메일 주소 확인

인텔 일회용 암호 986551

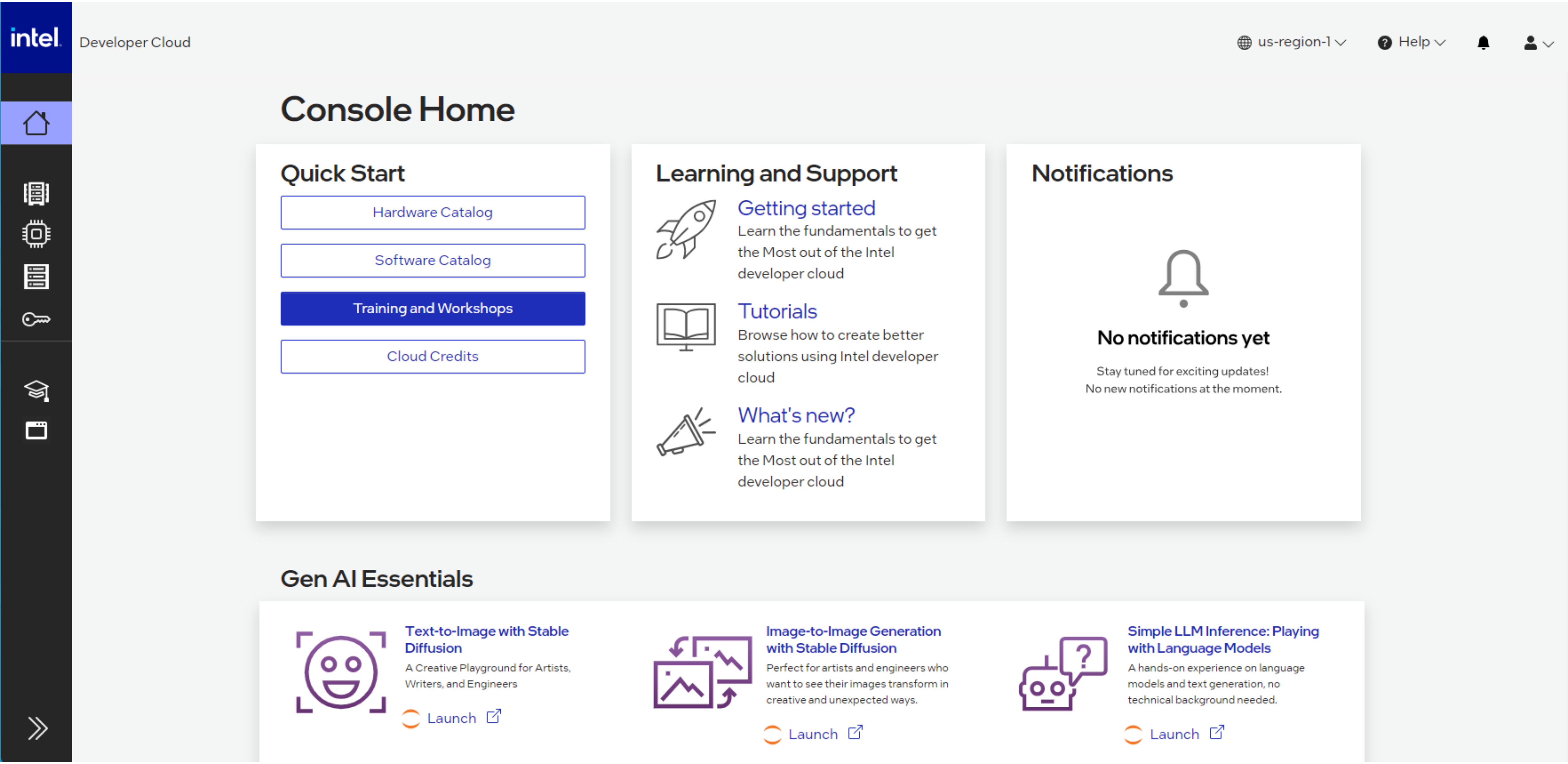
이 코드는 20분 후에 만료됩니다. 요청하지 않은 경우, 이 이메일을 무시할 수 있습니다.

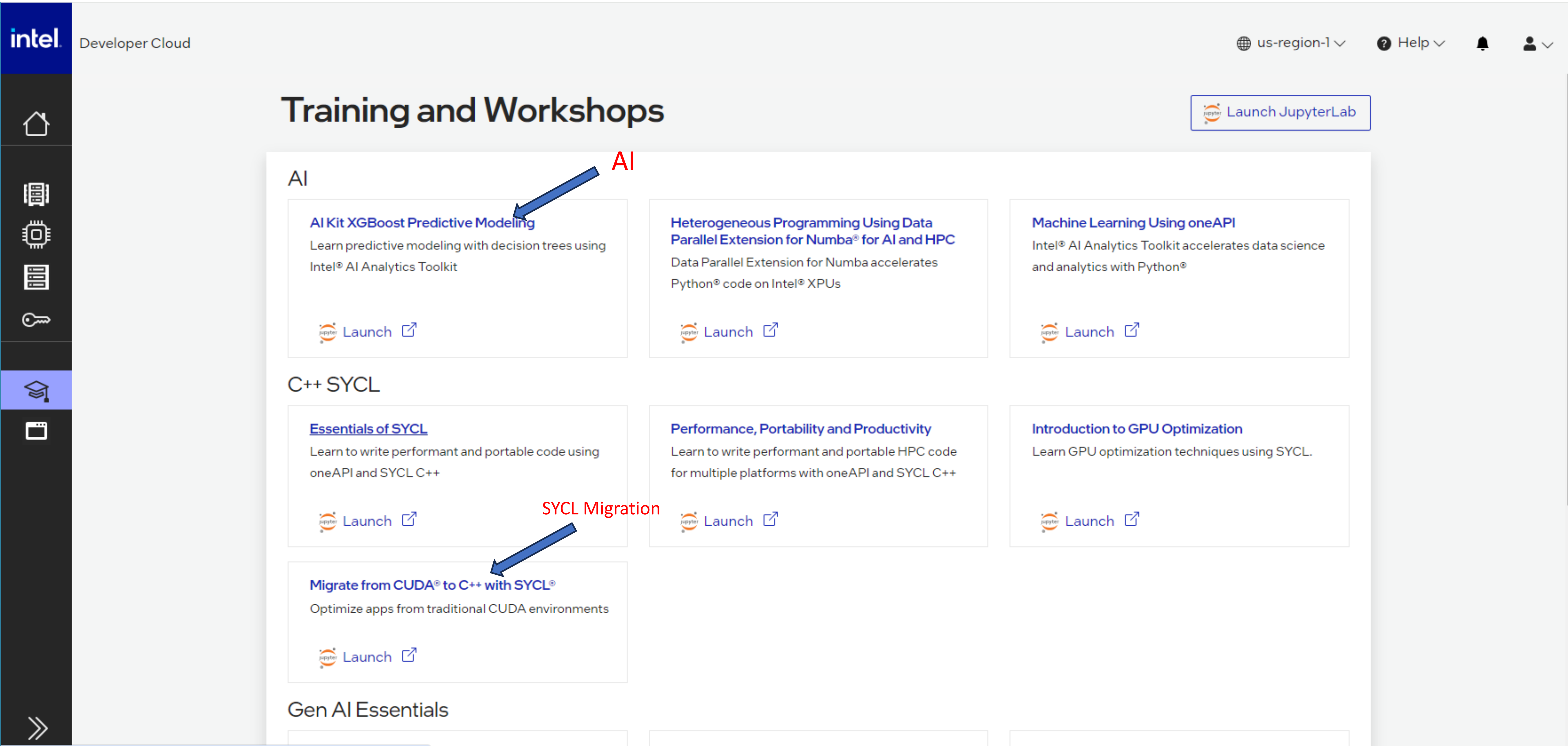
감사합니다.
인텔


intel
PRODUCTS
SUPPORT
SOLUTIONS
DEVELOPERS
PARTNERS
ENGLISH
Search Intel.com
Back
Please provide the following details.
We just sent a code to kth018@gmail.com
Verification code
98****
Create an account
Send new code
계정생성 완료 !!!
Company Overview
Contact Intel
Newsroom
Investors
Careers
Corporate Responsibility
Diversity & Inclusion
Public Policy
intel
© Intel Corporation | Terms of Use | *Trademarks | Cookies | Privacy | Supply Chain Transparency | Site Map
Intel technologies may require enabled hardware, software or service activation. // No product or component can be absolutely secure. // Your costs and results may vary. // Performance varies by use, configuration and other factors. // See our complete legal Notices and Disclaimers. // Intel is committed to respecting human rights and avoiding complicity in human rights abuses. See Intel's Global Human Rights Principles. Intel's products and software are intended only to be used in applications that do not cause or contribute to a violation of an internationally recognized human right.

- Intel® Developer Cloud site 에 접속

https://console.cloud.intel.com/GetStartedDevCloud


intel
Developer Cloud
us-region-1
Help
Console Home
Quick Start
Hardware Catalog
Software Catalog
Training and Workshops
Cloud Credits
Learning and Support
Getting started
Learn the fundamentals to get the Most out of the Intel developer cloud
Tutorials
Browse how to create better solutions using Intel developer cloud
What's new?
Learn the fundamentals to get the Most out of the Intel developer cloud
Notifications
No notifications yet
Stay tuned for exciting updates!
No new notifications at the moment.
Gen AI Essentials
Text-to-Image with Stable Diffusion
A Creative Playground for Artists, Writers, and Engineers
Launch
Image-to-Image Generation with Stable Diffusion
Perfect for artists and engineers who want to see their images transform in creative and unexpected ways.
Launch
Simple LLM Inference: Playing with Language Models
A hands-on experience on language models and text generation, no technical background needed.
Launch


intel.
Developer Cloud
us-region-1
Help
Training and Workshops
Launch JupyterLab
AI
AI
AI Kit XGBoost Predictive Modeling
Learn predictive modeling with decision trees using Intel® AI Analytics Toolkit
Launch
Heterogeneous Programming Using Data Parallel Extension for Numba® for AI and HPC
Data Parallel Extension for Numba accelerates Python® code on Intel® XPUs
Launch
Machine Learning Using oneAPI
Intel® AI Analytics Toolkit accelerates data science and analytics with Python®
Launch
C++ SYCL
Essentials of SYCL
Learn to write performant and portable code using oneAPI and SYCL C++
Launch
SYCL Migration
Performance, Portability and Productivity
Learn to write performant and portable HPC code for multiple platforms with oneAPI and SYCL C++
Launch
Introduction to GPU Optimization
Learn GPU optimization techniques using SYCL.
Launch
Migrate from CUDA® to C++ with SYCL®
Optimize apps from traditional CUDA environments
Launch
Gen AI Essentials

← Back to training

Migrate from CUDA® to C++ with SYCL®

Overview

The SYCLomatic Tool assists in migrating your existing CUDA code to SYCL® code.

Learning objectives

- Explain the advantages of using SYCL C++ language to program for accelerators
- Explain the program structure and execution model differences with CUDA and SYCL
- Setup and explain migration flow with SYCLomatic Tool

Getting started

1. Click on the Launch JupyterLab button.
2. Follow the instructions in the Jupyter notebook.

Launch Jupyter notebook

Options

Considerations

Your Jupyter notebook access will be live until 11/15/2023.

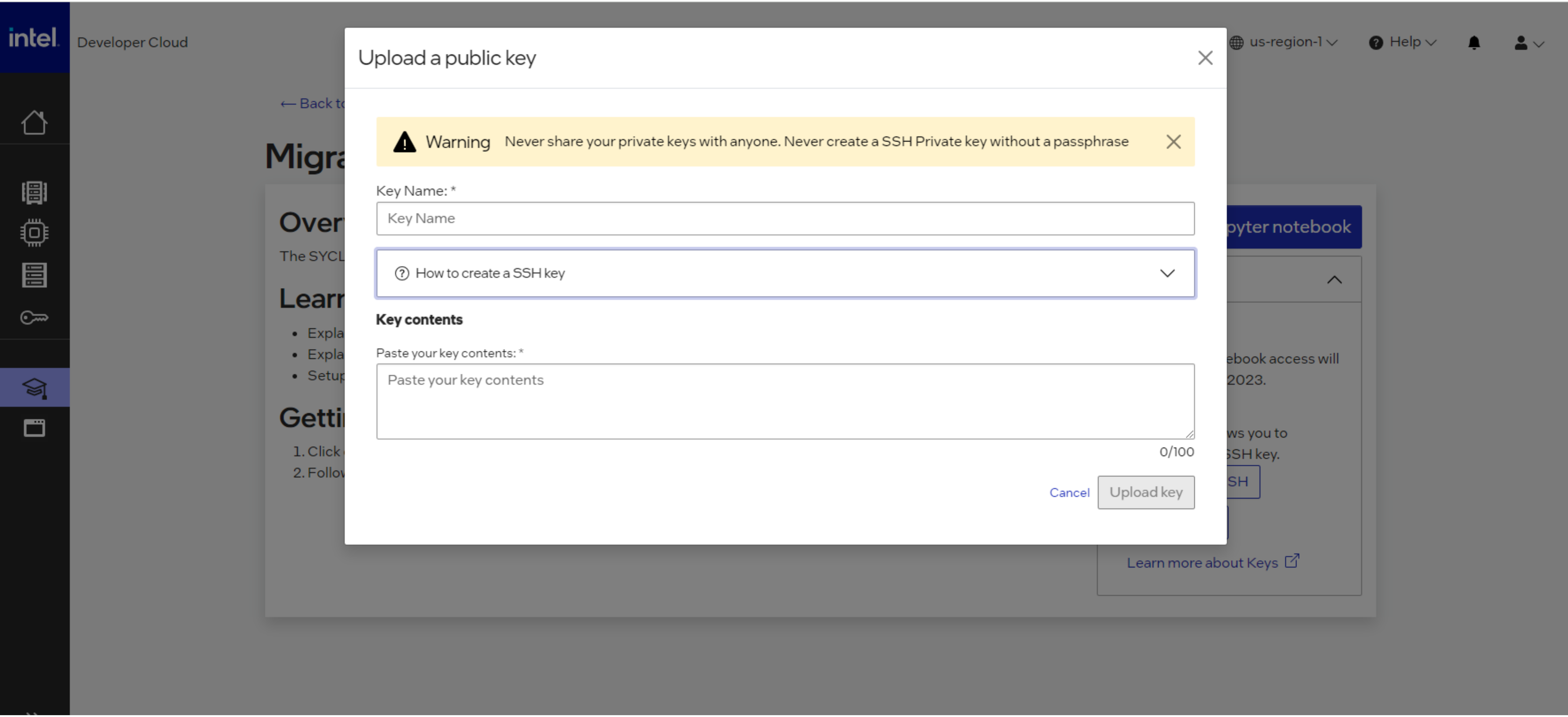
Expert mode

Expert mode allows you to connect using a SSH key.

Launch using SSH

Upload key

Learn more about Keys


intel.
Developer Cloud
us-region-1
Help
Upload a public key
Warning
Never share your private keys with anyone. Never create a SSH Private key without a passphrase
Key Name: *
Key Name
How to create a SSH key
Key contents
Paste your key contents: *
Paste your key contents
0/100
Cancel
Upload key
Learn more about Keys

Key Name: *

Key Name

How to create a SSH key

Select your OS:

- (•) Windows
- ( ) Linux/MacOS

1. Launch a new PowerShell window on your local system.
2. Optional: if you haven't generated a key before, create an .ssh directory.

```
mkdir $env:UserProfile\.ssh
```

3. Copy & paste the following to your terminal to generate SSH Keys.

```
ssh-keygen -t rsa -b 4096 -f $env:UserProfile\.ssh\id_rsa
```

4. If you are prompted to overwrite, select **No**.
5. Copy & paste the following to your Powershell window to view your SSH key:

```
cat $env:UserProfile\.ssh\id_rsa.pub
```

6. Paste your key's contents in the field below.

For more information go to SSH key documentation.

Key contents

Paste your key contents: *

intel. Developer Cloud

Key Name: *

Key Name

How to create a SSH key

Select your OS:

- ( ) Windows
- (•) Linux/MacOS

1. Launch a Terminal on your local system.
2. Copy & paste the following to your terminal to generate SSH Keys.

```
ssh-keygen -t rsa -b 4096 -f ~/.ssh/id_rsa
```

Copy

3. If you are prompted to overwrite, select **No**.
4. Copy & paste the following to your terminal to open your SSH key:

```
vi ~/.ssh/id_rsa.pub
```

Copy

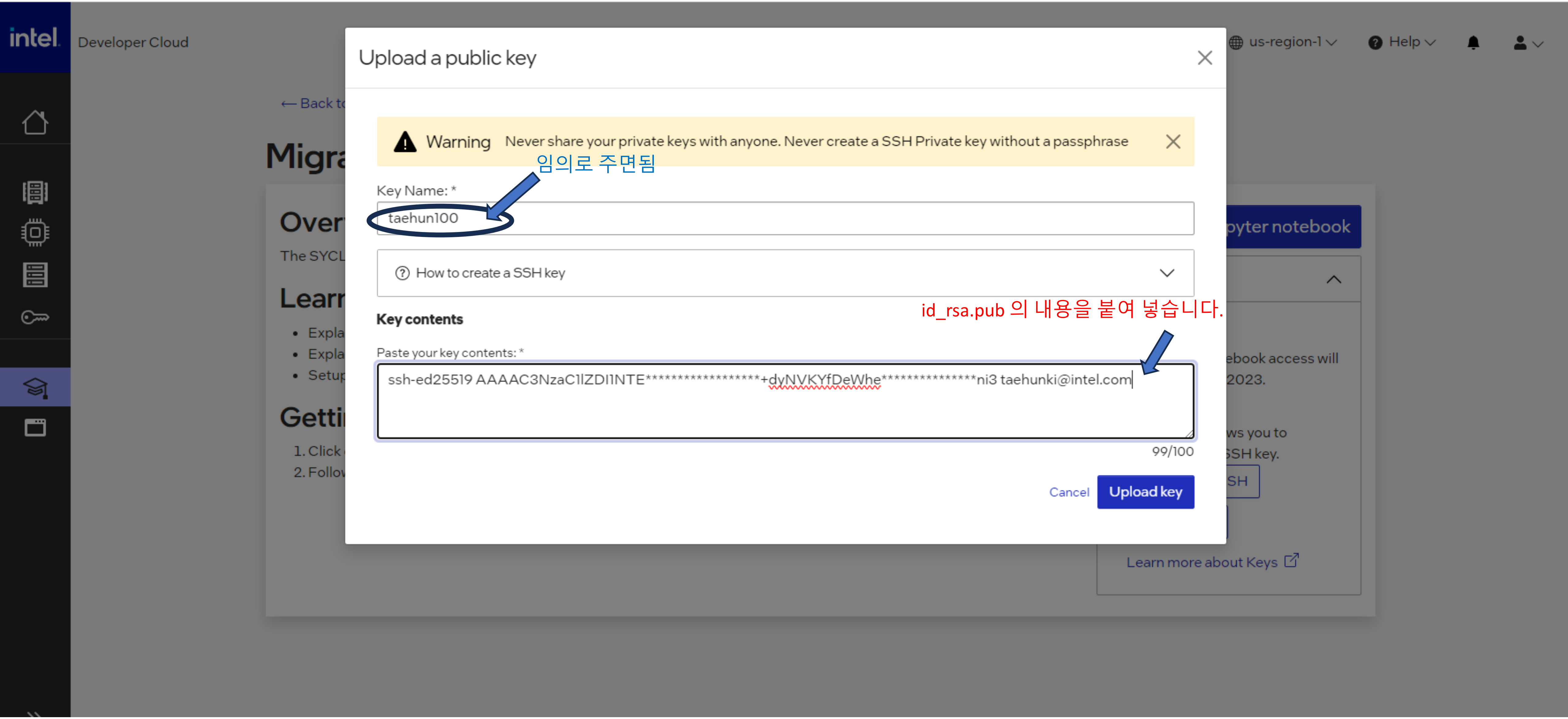
5. Upload the generated file

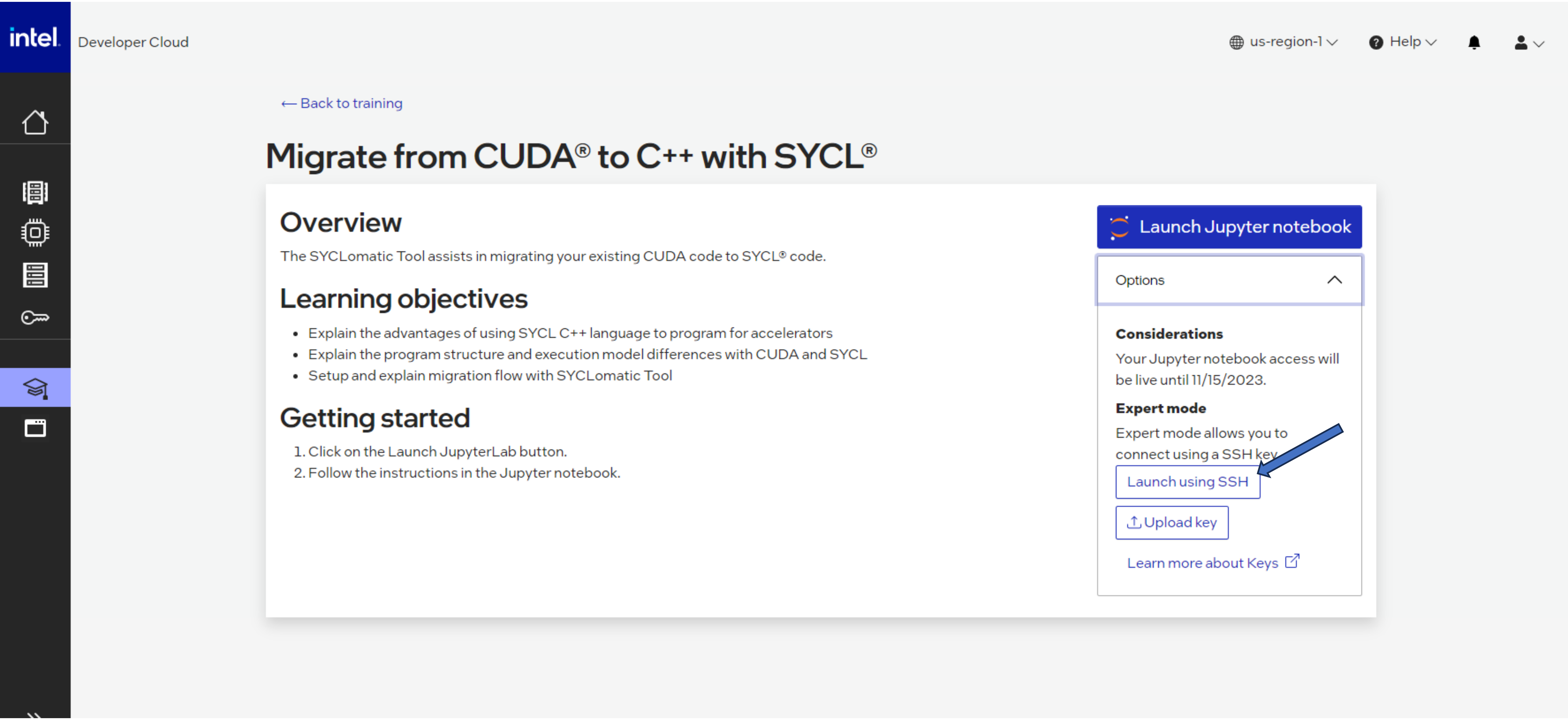
For more information go to SSH key documentation.

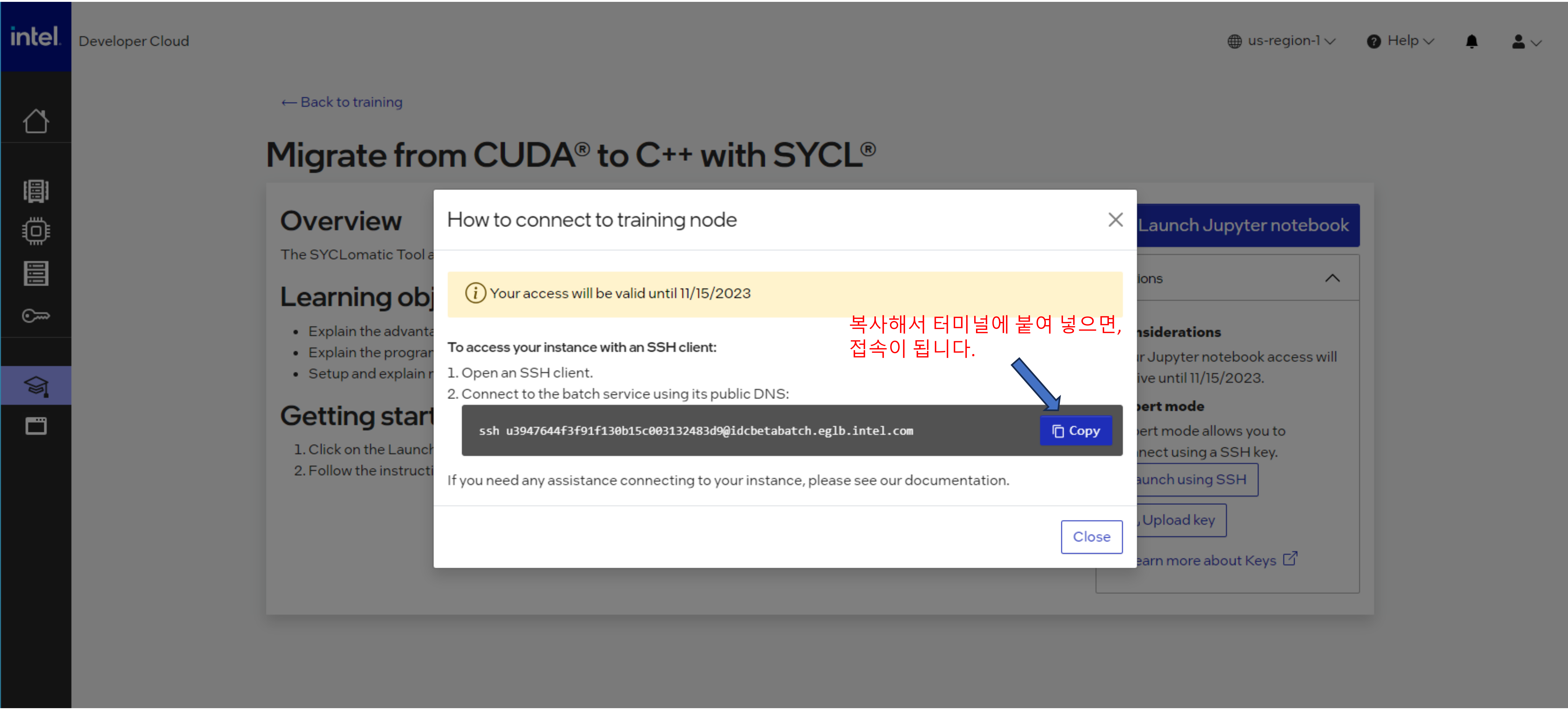
Key contents

Paste your key contents: *

Paste your key contents


intel.
Developer Cloud
us-region-1
Help
Upload a public key
Warning Never share your private keys with anyone. Never create a SSH Private key without a passphrase
임의로 주면됨
Key Name: *
taehun100
How to create a SSH key
id_rsa.pub 의 내용을 붙여 넣습니다.
Key contents
Paste your key contents: *
ssh-ed25519 AAAAC3NzaC1lZDI1NTE******************+dyNVKYfDeWhe***************ni3 taehunki@intel.com
99/100
Cancel
Upload key
Learn more about Keys


intel.
Developer Cloud
us-region-1
Help
← Back to training
Migrate from CUDA® to C++ with SYCL®
Overview
The SYCLomatic Tool assists in migrating your existing CUDA code to SYCL® code.
Learning objectives
Explain the advantages of using SYCL C++ language to program for accelerators
Explain the program structure and execution model differences with CUDA and SYCL
Setup and explain migration flow with SYCLomatic Tool
Getting started
1. Click on the Launch JupyterLab button.
2. Follow the instructions in the Jupyter notebook.
Launch Jupyter notebook
Options
Considerations
Your Jupyter notebook access will be live until 11/15/2023.
Expert mode
Expert mode allows you to connect using a SSH key
Launch using SSH
Upload key
Learn more about Keys


intel.
Developer Cloud
us-region-1
Help
← Back to training
Migrate from CUDA® to C++ with SYCL®
Overview
The SYCLomatic Tool a
Learning obj
Explain the advanta
Explain the prograr
Setup and explain r
Getting start
1. Click on the Launch
2. Follow the instructi
How to connect to training node
Your access will be valid until 11/15/2023
To access your instance with an SSH client:
1. Open an SSH client.
2. Connect to the batch service using its public DNS:
ssh u3947644f3f91f130b15c003132483d9@idcbetabatch.eglb.intel.com
Copy
If you need any assistance connecting to your instance, please see our documentation.
Close
복사해서 터미널에 붙여 넣으면,
접속이 됩니다.
Launch Jupyter notebook
ions
nsiderations
r Jupyter notebook access will
ive until 11/15/2023.
pert mode
pert mode allows you to
nnect using a SSH key.
aunch using SSH
Upload key
earn more about Keys

- IDC 서버 접속

터미널에서

ssh u******@idcbetabatch.eglb.intel.com

(WSL terminal, windows Command Prompt & Powershell 에서 ssh 가능)

* Summary slurm command

- slurm is scheduler on IDC
- sinfo : information about system status

  ex) sinfo –al
- srun : submit job

  ex) srun –p *<partition name>* -w *<node name>* --pty bash

* Software Prerequisites

Certain CUDA header files may need to be accessible to SYCLomatic [Supported CUDA version 11.8.]

A. Install on WSM and 'CUDA SDK's include directory' to 'your home directory of IDC'.

$ wget https://developer.download.nvidia.com/compute/cuda/11.8.0/local_installers/cuda_11.8.0_520.61.05_linux.run

$ sh cuda_11.8.0_520.61.05_linux.run

$ tar cvf cuda_inc.tar *<cuda-include-dir>*

$ sftp idcbeta

> put cuda_inc.tar

$ ssh idcbeta

$ tar xvf cuda_inc.tar

B. install 'CUDA SDK's include directory'

$ wget https://developer.download.nvidia.com/compute/cuda/11.8.0/local_installers/cuda_11.8.0_520.61.05_linux.run

$ sh cuda_11.8.0_520.61.05_linux.run

1. SYCLomatic Practice

- Download mini-nbody

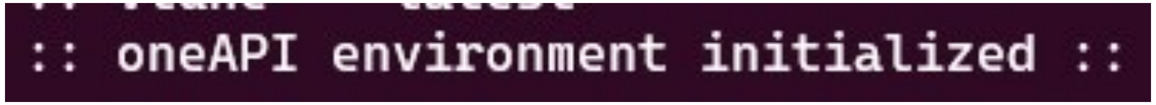
u*****@ idc-beta-batch-head-node:~$ git clone https://github.com/harrism/mini-nbody

- Mode to pvc-node (example)

u*****@ idc-beta-batch-**head-node**:~$ srun -p pvc-shared (-w idc-beta-batch-pvc-node-01) --pty bash

- setup oneAPI

u*****@ idc-beta-batch-**pvc-node-01**:~$ source /opt/intel/oneapi/setvars.sh



- CPU 에서 컴파일 및 수행

u*****@idc-beta-batch-pvc-node-01:~$ icx -O3 -fopenmp -DSHMOO –lm -o nbody nbody.c

u*****@ idc-beta-batch-pvc-node-01:~$ ./nbody 65536

65536, 46.703

(숫자가 클수록 성능이 좋음)

SYCLomatic mini-nbody on PVC

~$ cd mini-nbody/cuda/

~$ dpct --cuda-include-path=*<CUDA INCLUDE DIR>*
--extra-arg="-I/usr/include/c++/11"
--extra-arg="-I/usr/include/x86_64-linux-gnu/c++/11/"
--extra-arg="-I../" --extra-arg="-DSHMOO" nbody-orig.cu

Compile and Run mini-nbody on PVC

~$ cd dpct_output

~$ icpx -fsycl --verbose nbody-orig.dp.cpp -o nbody-orig-sycl-pvc-exe -I../../ -DSHMOO

~$ ./nbody-orig-sycl-pvc-exe 65536

65536, 239.385

(CPU 보다 ???배 큰 숫자)

2. Vtune 및 mkl을 활용한 최적화

튜토리얼 자료

https://cdrdv2-public.intel.com/671192/mkl-2017-tutorial-fortran.pdf

git에서 소스 코드 받기

$ git clone https://github.com/kth018/mkl_fortran_samples

< 코드 확인 >

두개의 Matrix 곱하기 프로그램

오리지날 코드 확인

~$ cat src/matrix_multiplication.f

< compile >

~$ cd /mkl_fortran_samples/matrix_multiplication

<Makefile 편집>

FC := ifort -> FC := ifx

LIBFLAGS := -mkl -static-intel -> LIBFLAGS := -qmkl

~$ make

< 프로그램 실행 및 Vtune 실행 >

~$ release/matrix_multiplication

…..

== Matrix multiplication using triple nested loop ==

== completed at   143.83056 milliseconds ==

Example completed.

~$ vtune -collect hotspot release/matrix_multiplication

* -collect [hpc-performance | memory-access | hotspot ….]

vtune 수행 (Hotspots)

- Vtune에서 소스코드를 연결해서 보려면 '-g' 옵션을 넣고 컴파일을 해야 한다.

```
tornado@tornado-linux:~/WORK/KSC23$
tornado@tornado-linux:~/WORK/KSC23$ ifx -O3 -g -qmkl -o matrix_multiplication  matrix_multiplication.f
tornado@tornado-linux:~/WORK/KSC23$
```

- vtune –collect *hotspots matrix_multiplication*

```
tornado@tornado-linux:~/WORK/KSC23$
tornado@tornado-linux:~/WORK/KSC23$ vtune -collect hotspot ./matrix_multiplication
vtune: Collection started. To stop the collection, either press CTRL-C or enter from another co
nsole window: vtune -r /home/tornado/WORK/KSC23/r000hs -command stop.
 This example measures performance of computing the real
 matrix product C=alpha*A*B+beta*C using
 a triple nested loop, where A, B, and C are matrices
 and alpha and beta are double precision scalars
```

- 실행 후 아래처럼 100% 라고 나오면, vtune 작업이 끝나며, (hotspot 경우) r001hs 라는 디렉토리가 생성됨

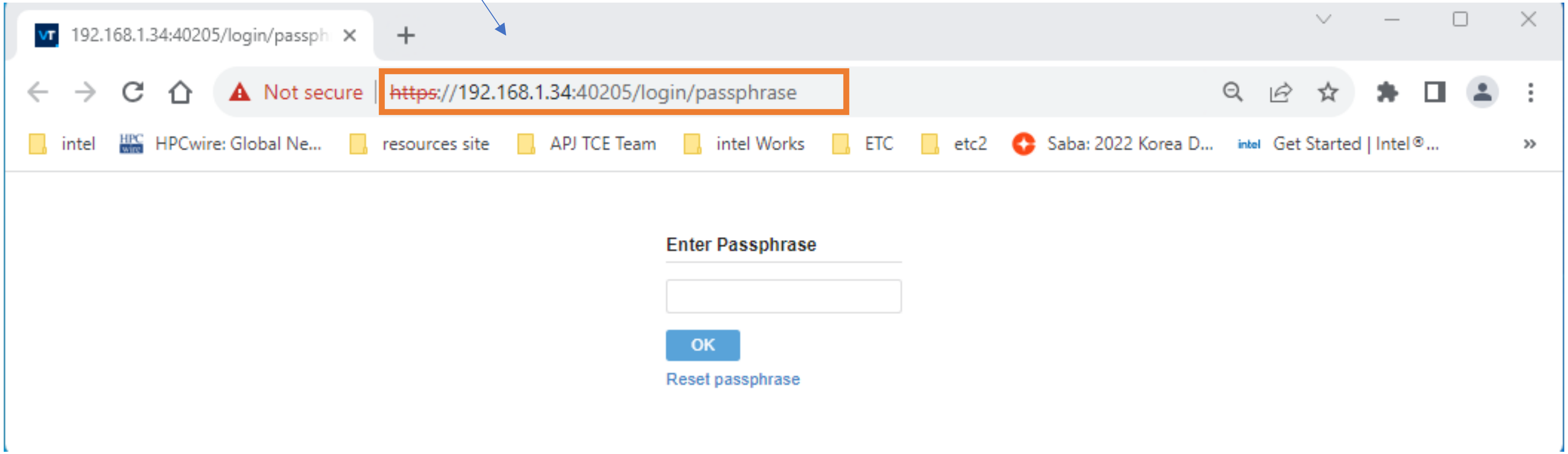
```
vtune: Executing actions 100 % done
tornado@tornado-linux:~/WORK/KSC23$ ls
compile.txt  matrix_multiplication  matrix_multiplication.f  r000hs
tornado@tornado-linux:~/WORK/KSC23$
```

서버에서 vtune-backend 실행

명령어 : vtune-backend -–allow-remote-access –-data-directory (프로파일 결과 디렉토리)

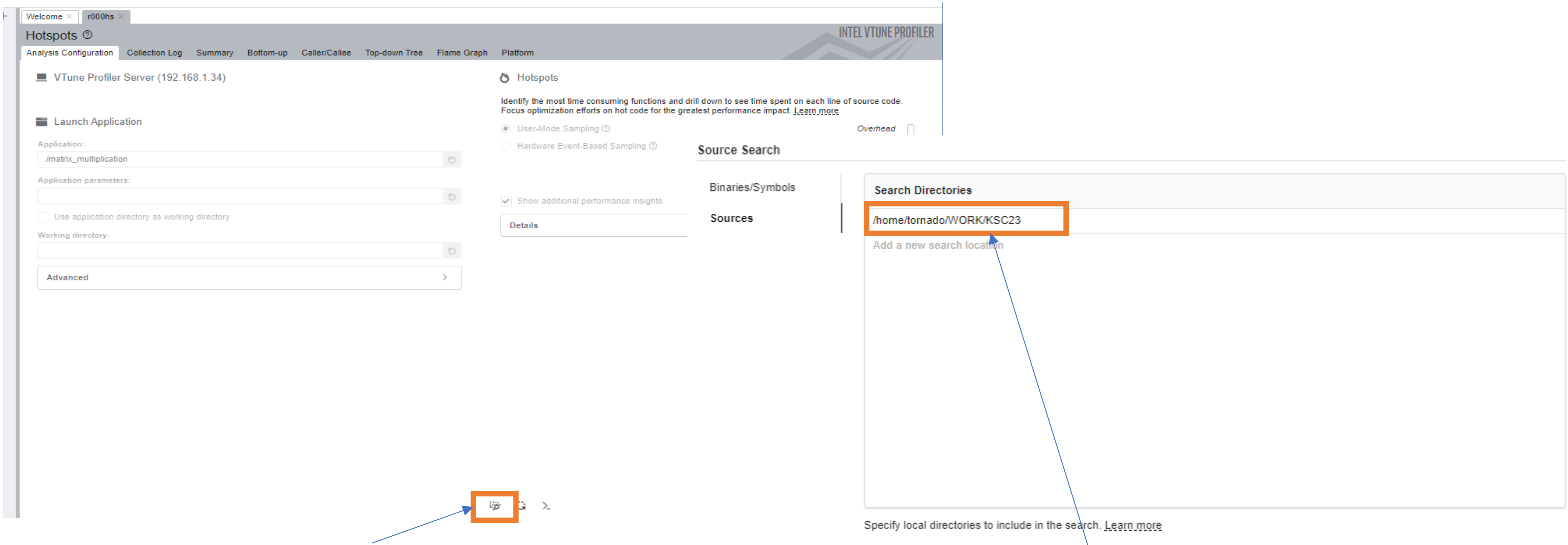
```
tornado@tornado-linux:~/WORK/KSC23$ vtune-backend --allow-remote-access --data-directory /home/
tornado/WORK/KSC23/r00hs
No TLS certificate was provided as a --tls-certificate command-line argument thus a self-signed
 certificate is generated to enable secure HTTPS transport for the web server: /home/tornado/.i
ntel/amplxe/settings/certificates/middleware.crt.
Serving GUI at https://192.168.1.34:40205/
```

보여지는 URL로 접속



* password 요구시 계정 password를 치고 들러가면 됨

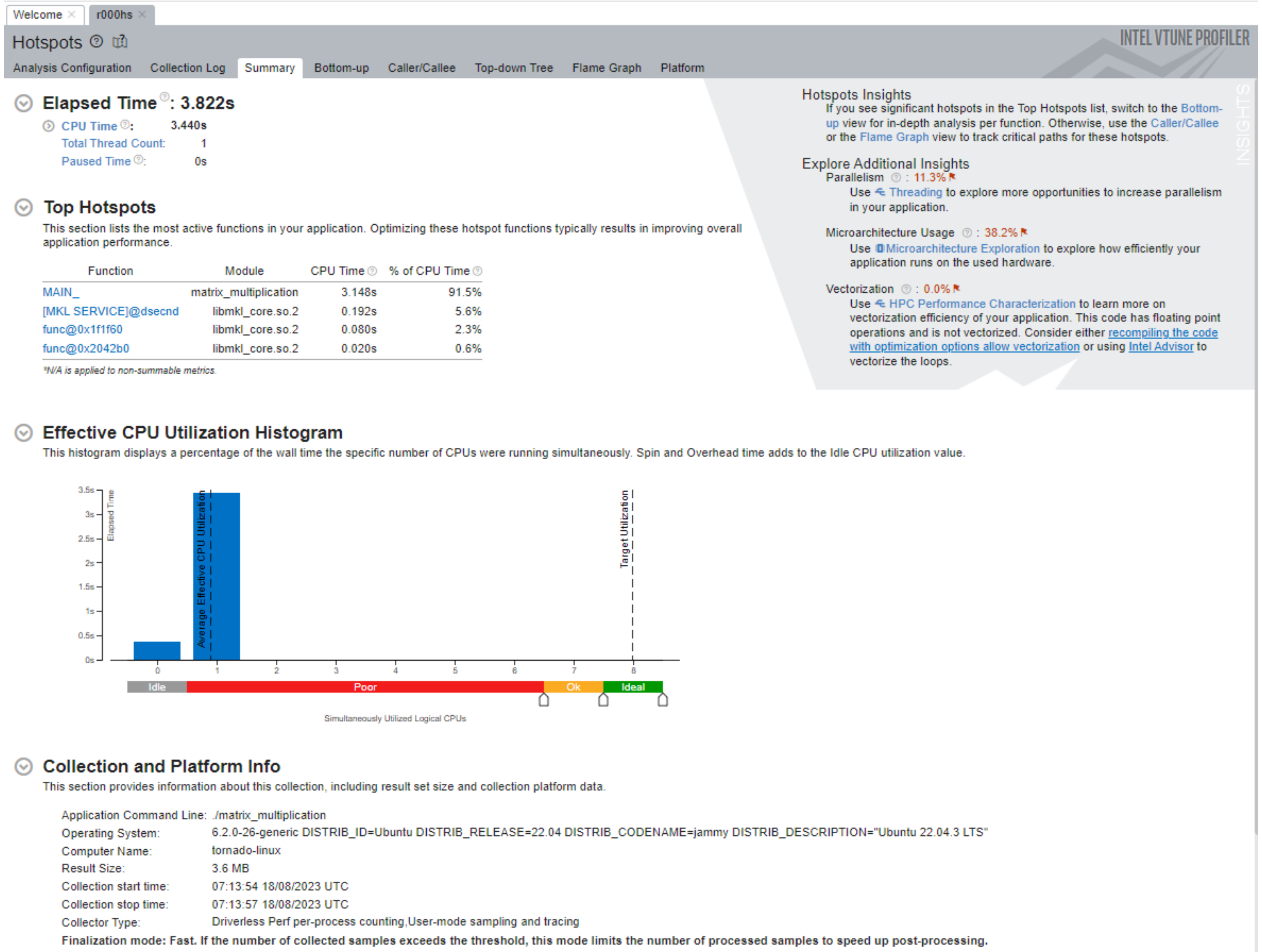
vtune에서 소스코드 연결



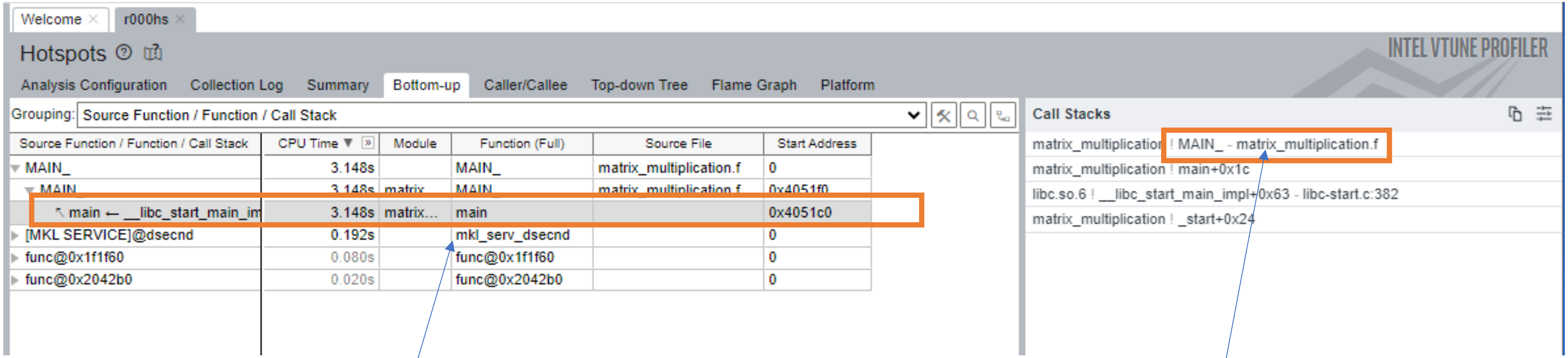
Analysis Configuration 탭에서 버튼 클릭

창이 나오면 Sources 탭을 클릭 후, 소스 디렉토리(전체 경로) 입력

결과 1


Welcome
r000hs
Hotspots
INTEL VTUNE PROFILER
Analysis Configuration
Collection Log
Summary
Bottom-up
Caller/Callee
Top-down Tree
Flame Graph
Platform
Elapsed Time: 3.822s
CPU Time: 3.440s
Total Thread Count: 1
Paused Time: 0s
Top Hotspots
This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.
Function | Module | CPU Time | % of CPU Time
MAIN_ | matrix_multiplication | 3.148s | 91.5%
[MKL SERVICE]@dsecnd | libmkl_core.so.2 | 0.192s | 5.6%
func@0x1f1f60 | libmkl_core.so.2 | 0.080s | 2.3%
func@0x2042b0 | libmkl_core.so.2 | 0.020s | 0.6%
*N/A is applied to non-summable metrics.
Hotspots Insights
If you see significant hotspots in the Top Hotspots list, switch to the Bottom-up view for in-depth analysis per function. Otherwise, use the Caller/Callee or the Flame Graph view to track critical paths for these hotspots.
Explore Additional Insights
Parallelism: 11.3%
Use Threading to explore more opportunities to increase parallelism in your application.
Microarchitecture Usage: 38.2%
Use Microarchitecture Exploration to explore how efficiently your application runs on the used hardware.
Vectorization: 0.0%
Use HPC Performance Characterization to learn more on vectorization efficiency of your application. This code has floating point operations and is not vectorized. Consider either recompiling the code with optimization options allow vectorization or using Intel Advisor to vectorize the loops.
INSIGHTS
Effective CPU Utilization Histogram
This histogram displays a percentage of the wall time the specific number of CPUs were running simultaneously. Spin and Overhead time adds to the Idle CPU utilization value.
Elapsed Time
3.5s
3s
2.5s
2s
1.5s
1s
0.5s
0s
Average Effective CPU Utilization
Target Utilization
0 1 2 3 4 5 6 7 8
Idle
Poor
Ok
Ideal
Simultaneously Utilized Logical CPUs
Collection and Platform Info
This section provides information about this collection, including result set size and collection platform data.
Application Command Line: ./matrix_multiplication
Operating System: 6.2.0-26-generic DISTRIB_ID=Ubuntu DISTRIB_RELEASE=22.04 DISTRIB_CODENAME=jammy DISTRIB_DESCRIPTION="Ubuntu 22.04.3 LTS"
Computer Name: tornado-linux
Result Size: 3.6 MB
Collection start time: 07:13:54 18/08/2023 UTC
Collection stop time: 07:13:57 18/08/2023 UTC
Collector Type: Driverless Perf per-process counting,User-mode sampling and tracing
Finalization mode: Fast. If the number of collected samples exceeds the threshold, this mode limits the number of processed samples to speed up post-processing.

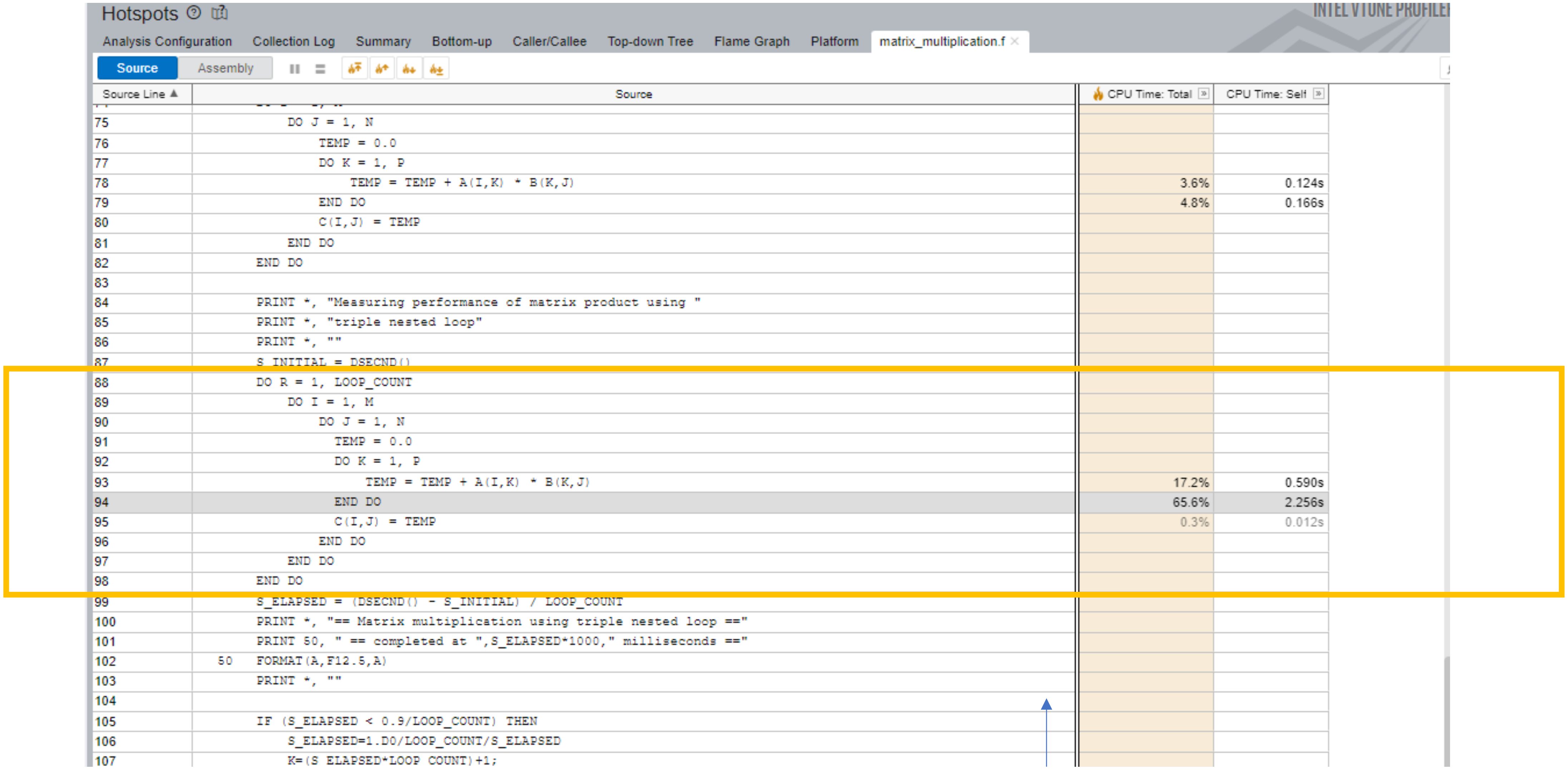
결과 2



Bottom-up 탭에서 가장 오래 걸리는 부분 보기

클릭하면 소스코드 탭이 생기며,
가장 시간이 많이 걸리는 라인으로 이동

결과 3



소스코드 탭이 생기며, 94 주변 라인이 65.6% 소요됨

< mkl 코드 확인 >

~$ cat src/dgemm_with_timing.f

```
      PRINT *, "Making the first run of matrix product using "
      PRINT *, "Intel(R) MKL DGEMM subroutine to get stable "
      PRINT *, "run time measurements"
      PRINT *, ""
      CALL DGEMM('N','N',M,N,P,ALPHA,A,M,B,P,BETA,C,M)

      PRINT *, "Measuring performance of matrix product using "
      PRINT *, "Intel(R) MKL DGEMM subroutine"
      PRINT *, ""
      S_INITIAL = DSECND()
      DO R = 1, LOOP_COUNT
          CALL DGEMM('N','N',M,N,P,ALPHA,A,M,B,P,BETA,C,M)
      END DO
      S_ELAPSED = (DSECND() - S_INITIAL) / LOOP_COUNT
      PRINT *, "== Matrix multiplication using Intel(R) MKL DGEMM =="
      PRINT 50, " == completed at ",S_ELAPSED*1000," milliseconds =="
 50   FORMAT(A,F12.5,A)
      PRINT *, ""
```

< MKL 적용 코드 성능 확인 >

~$ release/dgemm_with_timing

```
….
 == Matrix multiplication using Intel(R) MKL DGEMM ==
 == completed at      0.31979 milliseconds ==

 It is highly recommended to set parameter LOOP_COUNT
 for this example on your computer as        3128  to have
 total execution time about 1 second for reliability
 of measurements

 Example completed.
```

<참고> oneMKL openMP GPU offload

- openMP GPU offload 설명

https://www.intel.com/content/www/us/en/docs/oneapi/optimization-guide-gpu/2023-0/offloading-onemkl-computations-onto-the-gpu.html

Notices and Disclaimers

- You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.
- The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.