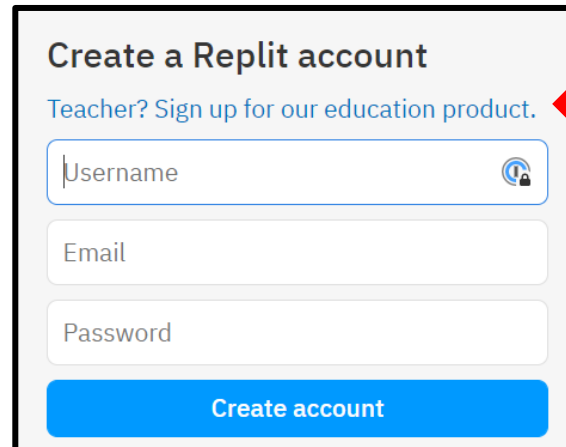


C++ / JAVA 프로그래밍


01주차 실습
replit 사용법

실습 홈페이지 참여

- 아래 링크를 클릭하여 replit teams에 참여한다.
 - <https://replit.com/teams/join/ohgawxdihnejxyqvpzxtzujpobkgxlej-INE2023-20212>
 - 이미 아이디와 비밀번호가 있는 경우 ①에 입력 후 로그인
 - 아이디가 없는 경우 ②에 있는 "Sign up" 문구 클릭
 - Google, Github, Facebook을 이용해 로그인 또는 가입하려는 경우 ③에서 원하는 방식 선택



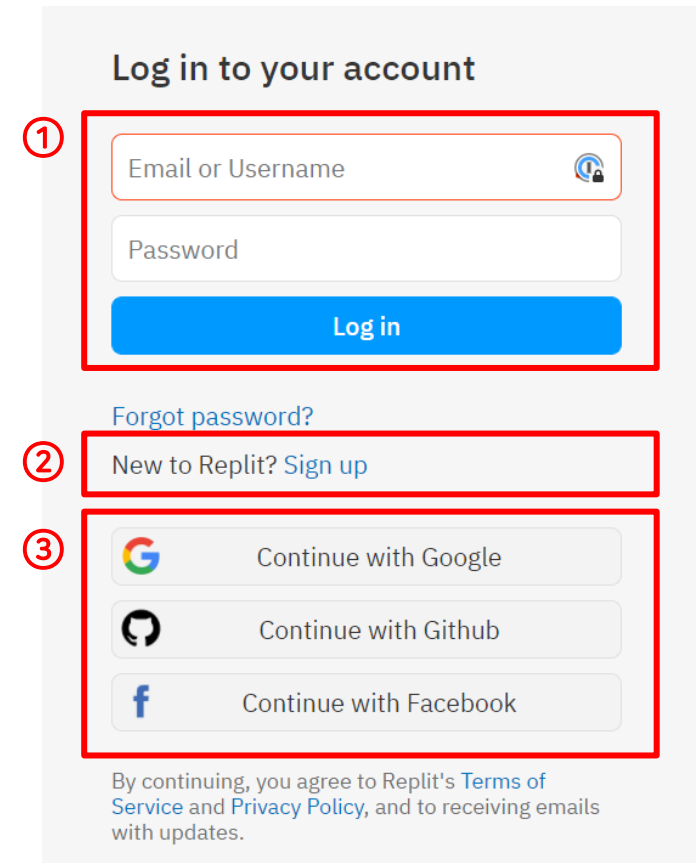
Create a Replit account
Teacher? Sign up for our education product.

Username 


Email

Password

Create account



Log in to your account

① Email or Username 


Password


Log in


Forgot password?

② New to Replit? Sign up

③

 Continue with Google

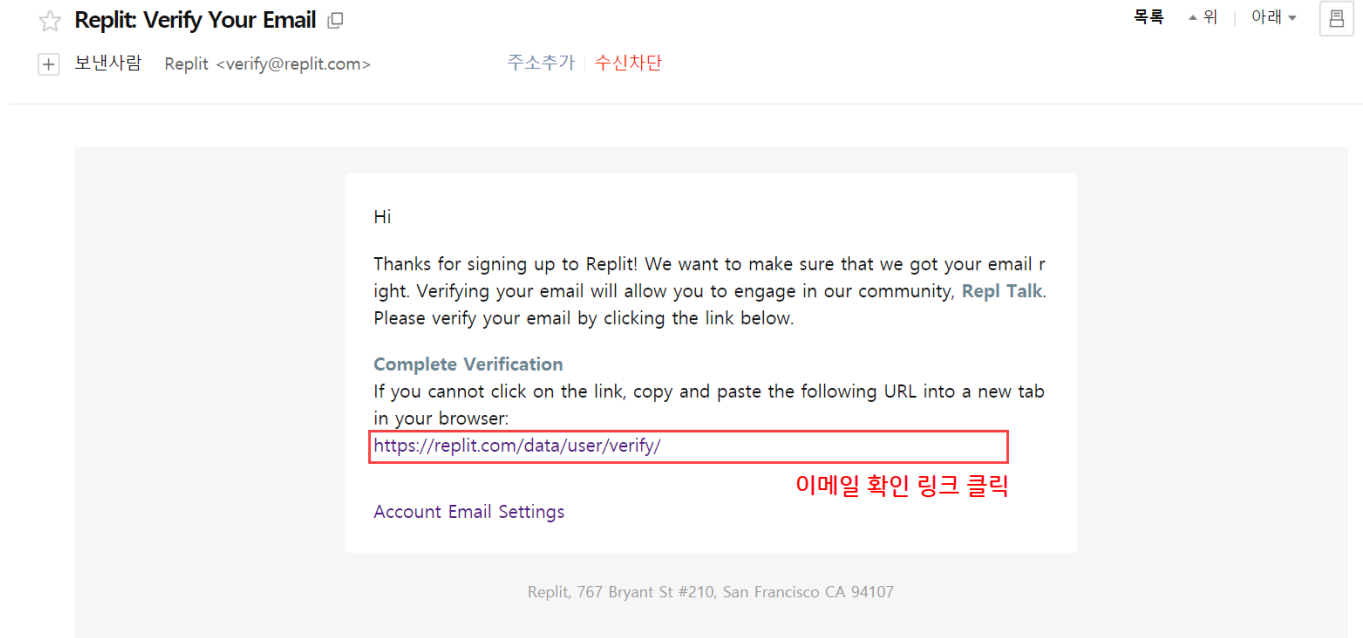
 Continue with Github

 Continue with Facebook

By continuing, you agree to Replit's [Terms of Service](#) and [Privacy Policy](#), and to receiving emails with updates.

새로 가입한 경우 주의사항

- 새로 가입했다면 반드시 이메일에서 **이메일 확인 링크**를 클릭해야 함
 - 링크를 클릭하지 않으면 과제 제출이 완료되지 않음



- 화면에 아래와 같은 문구가 나오면 이메일 확인이 되지 않은 것

Please verify your email. [Resend verification link](#)

초기 화면 레이아웃



- ① 과제 이름
- ② 마감일
- ③ 과제 시작



C++ / Java 프로그래밍 2021

@INE2023-20212

산업경영공학과 C++ / JAVA 프로그래밍 실습 페이지 입니다.



Team Projects

Export submissions

Search for projects...

1주차 실습

TITLE ①

DUE DATE ②

③ SUBMISSION



Hello World

화면에 "Hello World"라는 문구를 출력해...

9/12/2021 12:00 PM

Start project

과제 페이지 레이아웃



실행

제출

INE2023-20212 / 1주차 실습/01/Hello ...

Run ▶

Submit

Files

main.cpp

README.md

파일 탐색기

main.cpp x

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Hello World!\n";
5     return 0;
6 }
```

Console Shell Markdown

Preview of README.md

Hello World 출력하기

화면에 Hello World라는 문자열을 출력해보자.

- 문자열을 출력하기 위해 `std::cout` 을 사용한다.
- 문자열은 `" "` 으로 둘러싸야 한다.
- 문자열 출력 후 줄바꿈을 위해, 줄바꿈 문자 `\n` 을 문자열 끝에 붙인다.

This project has 1 input/output test.
Run tests to check your work!

Dismiss

과제 제출 시 테스트해볼 내용이 하나 있음

과제에 대한 설명

설정

① 출력창 위치 설정

- side-by-side: 좌우로 놓기
- stacked: 위아래로 놓기

② 개발환경 색상 설정

③ 코드 글자 크기

④ 들여쓰기 형식

- spaces: 공백문자로 들여쓰기
- tabs: 탭으로 들여쓰기

⑤ 들여쓰기 크기

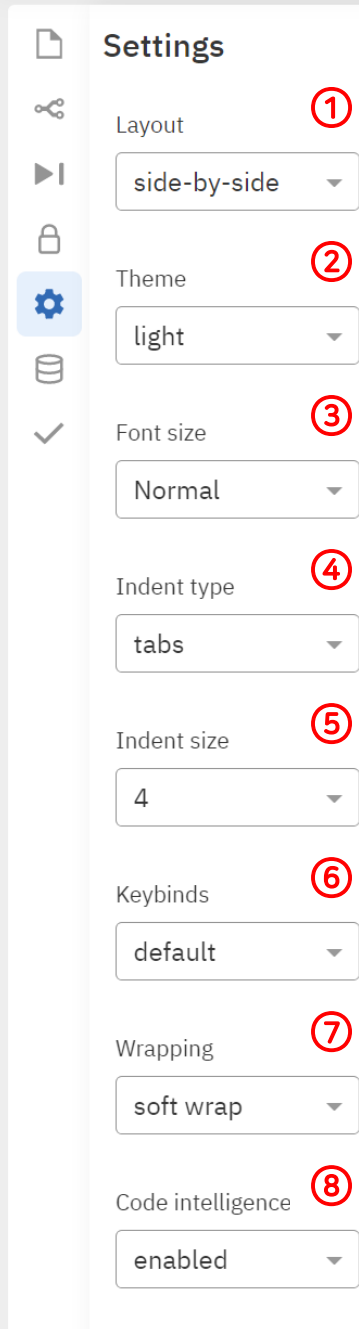
- 공백문자 크기 x 설정한 숫자 크기

⑥ 단축키 설정

- emacs 또는 vim 단축키 사용

⑦ 자동 줄 바꿈 기능 사용여부

⑧ 코드 자동완성 기능 사용여부



도움 요청

- ① 도움이 필요한 곳을 선택한 후, "Start thread" 버튼을 누릅니다.

```

2
3
4 std::cout << "Hello World!\n";
5 return 0;
6 }
    
```

Start thread

- ② 도움이 필요한 내용을 입력합니다.

@JaehyungEun
 여기 좀 도와주세요

▶

- ③ 완료되었다면 Resolve 버튼을 누릅니다.

Resolve

✓ ✕

TestHasoo 1m

@JaehyungEun
여기 좀 도와주세요

JaehyungEun now

@TestHasoo 안녕하세요

@TestHasoo 제출하기 버튼을 눌러주세요

Reply to @JaehyungEun ▶

과제 테스트 - 실패한 경우

INE2023-20212 / 1주차 실습/01/Hello ...

Run

Input/Output Tests

+ Create test

Run tests

test

Failed

Results

main.cpp

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Jello World!\n";
5     return 0;
6 }
```

Console

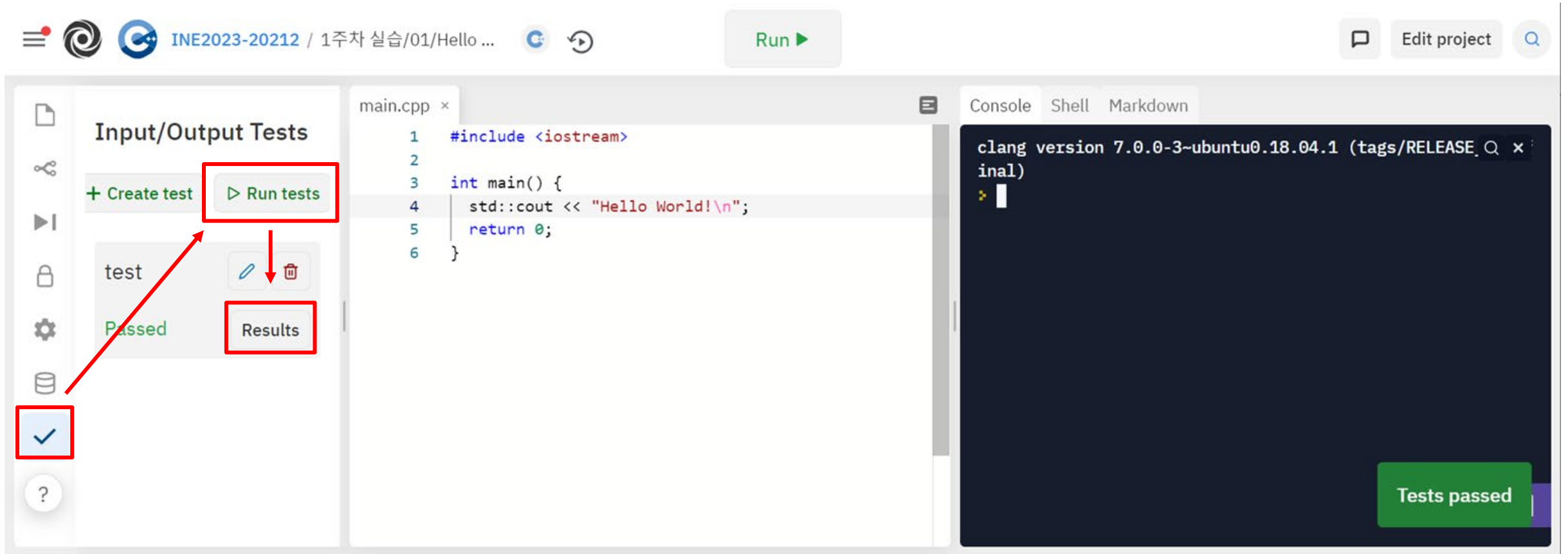
Shell

Markdown

clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_7.0.0-3~ubuntu0.18.04.1)

Tests failed

과제 테스트 - 성공한 경우



The screenshot displays a code editor interface for a project named "INE2023-20212 / 1주차 실습/01/Hello ...". The main editor shows a C++ file named "main.cpp" with the following code:

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Hello World!\n";
5     return 0;
6 }
```

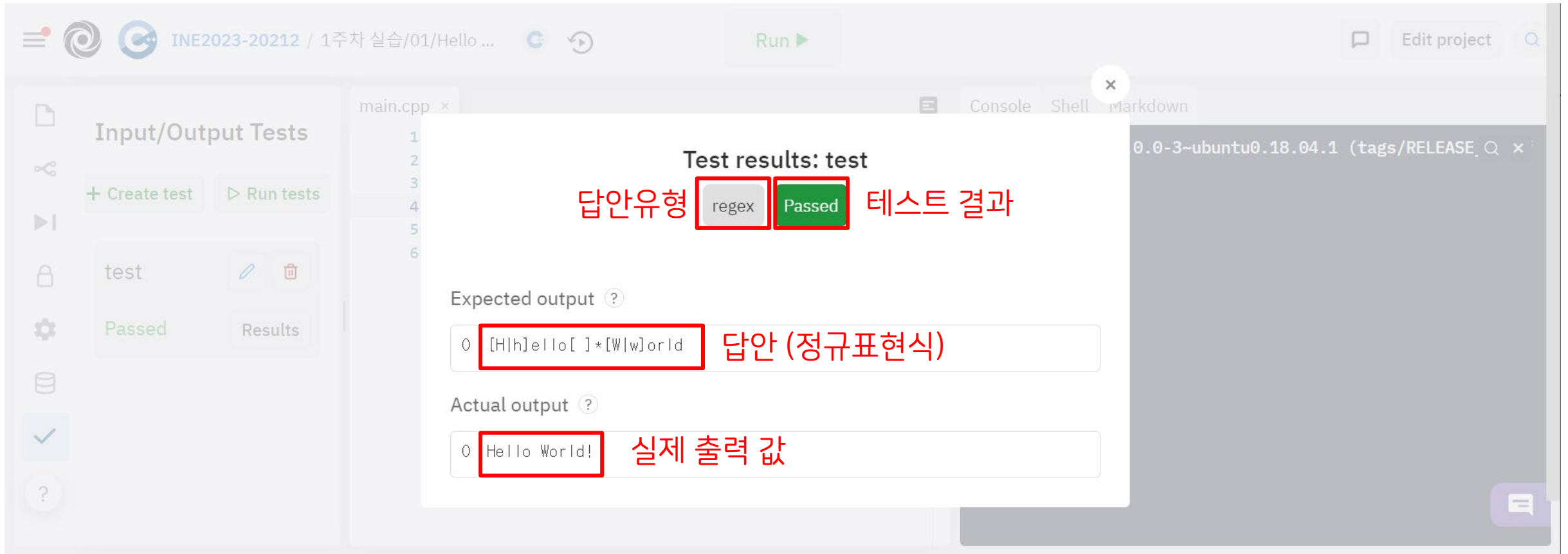
On the left sidebar, under "Input/Output Tests", there is a "test" entry with a status of "Passed". A red box highlights the "Run tests" button, and another red box highlights the "Results" button. A red arrow points from the "test" entry to the "Run tests" button. A blue checkmark icon is also visible in the sidebar.

The right sidebar shows the "Console" tab with the output:

```
clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_7.0.0-3~ubuntu0.18.04.1)
>
```

A green button labeled "Tests passed" is visible in the bottom right corner of the console area.

과제 테스트 - 세부 내용 확인



The screenshot shows a code editor interface with a sidebar on the left containing 'Input/Output Tests'. The main area displays a 'Test results: test' dialog box. The dialog box has a title bar with a close button. Inside, it shows the test name 'test' and its status 'Passed'. Below this, it displays the 'Expected output' and 'Actual output' sections. The 'Expected output' section shows a regex pattern '[H|h]ello[]*[W|w]orld' and the 'Actual output' section shows the string 'Hello World!'. Both the regex pattern and the string 'Hello World!' are highlighted with red boxes. The background shows a code editor with a file named 'main.cpp' and a 'Run' button.

Test results: test

답안유형 regex Passed 테스트 결과

Expected output ?

0 [H|h]ello[]*[W|w]orld 답안 (정규표현식)

Actual output ?

0 Hello World! 실제 출력 값

프로그램을 만들 때의 약속

- 원칙적으로 로마자를 이용해 작성합니다.
 - 주석 및 "" (큰따옴표) 안에는 한글이나 한자 등을 사용할 수 있습니다.
- 전각 공백 문자 사용에 주의합니다.
 - 전각 공백 문자(␣ -> "한자 키" -> 1)를 큰따옴표 밖에서 사용하면 오류가 생깁니다.
 - 발견하기 어려우므로 주의합시다.
- 대문자와 소문자를 구별해서 사용합니다.
 - 컴파일러는 `std::cout`과 `Std::cout`를 완전히 다른 의미로 해석합니다.
- 주석문은 `/* */`안에 작성하거나, `//` 뒤에 작성합니다.
 - 주석은 해당 부분에 대한 설명을 적는 것으로서, 프로그램에 반영되지 않습니다.
- 예약어 사용에 주의합니다.
 - 예약어는 프로그램에서 사용하기로 예약된 단어들을 말합니다. 예약어는 마음대로 사용할 수 없습니다.

실습1 Hello World 출력하기

- 화면에 Hello World라는 문자열을 출력해보자.
 - 문자열을 출력하기 위해 `std::cout`을 사용합니다.
 - 문자열은 `""`으로 둘러싸야 합니다.
 - 문자열 출력 후 줄바꿈을 위해, 줄바꿈 문자 `\n`을 문자열 끝에 붙입니다.
 - 문자열 안에 줄바꿈 문자를 넣는 대신, 문자열 뒤에 `<< std::endl`을 사용할 수 있습니다.

```
Console Shell Markdown
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
Hello World!
> 
```

실습2 학번과 이름 출력하기

- 화면에 학번과 이름을 출력해보자
 - 학번이라 쓰인 부분에 10자리 학번을 입력합니다.
 - 이름이라 쓰인 부분에 이름을 한글로 입력합니다.
 - 학번과 이름 사이에는 띄어쓰기를 넣습니다.
 - 이름은 띄어쓰기 없이 2글자 이상 입력합니다.

```
Console Shell Markdown
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
2021123456 은 재 형
> 
```

실습3 두 줄 출력하기

- 화면에 본인의 ****키****와 ****학점****을 출력해봅시다.
 - 출력할 문장은 `std::cout <<` 뒤에 작성해야 합니다.
 - 출력할 문장은 `"`로 둘러 싸야 합니다.
 - 한 줄에 여러 개의 `<<`을 사용할 수 있습니다.
 - 키와 학점은 서로 다른 줄에 출력해야 합니다.
 - 다음 줄에 출력하려면 그 줄의 끝에 `std::endl`을 붙여야 합니다.
 - `std::endl` 대신 `\n`을 사용할 수 있지만, 화면에 바로 출력되지 않을 수도 있습니다.

```
Console Shell
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
제 키는 185cm 입니다.
이번 학기에는 A+ 학점을 받을 예정입니다.
```

[과제] 삼각형 출력하기

- * 문자를 이용해서 삼각형을 출력해봅시다.
 - 제일 첫 줄에는 한 개의 * 문자를 출력합니다.
 - 두 번째 줄에는 두 개의 * 문자를 출력합니다.
 - 줄이 증가할 때마다 * 문자를 하나 씩 추가합니다.
 - 가장 아랫줄의 * 문자 수가 다섯 개가 될 때까지 반복합니다.

```
Console Shell Markdown
clang++-7 -pthread -std=c++17 -o main main.cpp
./main
*
**
***
****
*****
```

