1-2 : Java & Git 기초

들어가기 전에

이번 시간에는…

IntelliJ 및 JDK 구축
Java 기초
Git 기초

자바

JVM, JRE, JDK

JVM: 자바 프로그램을 실행할 수 있는 환경 제공

JRE: JVM + 자바 프로그램을 동작 시킬 때 필요한 라이브러리, 파일 포함

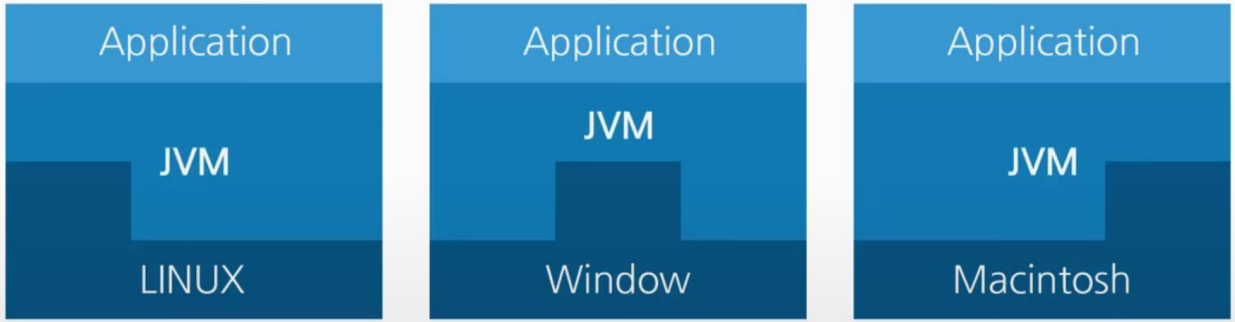
JDK: JRE + 개발을 위해 필요한 도구 포함

자바

자바(Java)의 특징

플랫폼에 독립적이다.

자바는 자바 실행환경(JRE)이 설치되어 있는 모든 운영체제에서 실행이 가능하다.
서로 다른 실행환경을 가진 시스템에서 동일하게 자바를 실행할 수 있다.(이식성)
JVM을 거쳐야하기 때문에 다른 언어에 비해 실행 속도가 느리다.



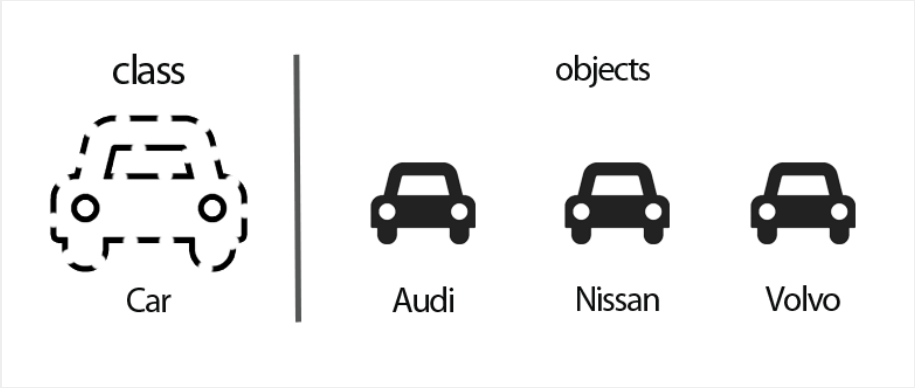
자바

자바(Java)의 특징

객체지향 프로그래밍언어

객체지향 프로그래밍은 프로그래밍을 개발하는 기법

부품에 해당하는 객체를 만들고, 이것들을 하나씩 조립 및 연결하여 개발(재활용)



자바

자바(Java)의 특징

메모리를 자동으로 관리한다.

C, C++은 메모리를 직접 할당하고, 삭제하여 관리한다.

자바는 개발자가 직접 메모리에 접근할 수 없으며, Garbage Collector를 통해 자바가 직접 관리한다.

C/C++

Java

개발환경구축

필수 요소

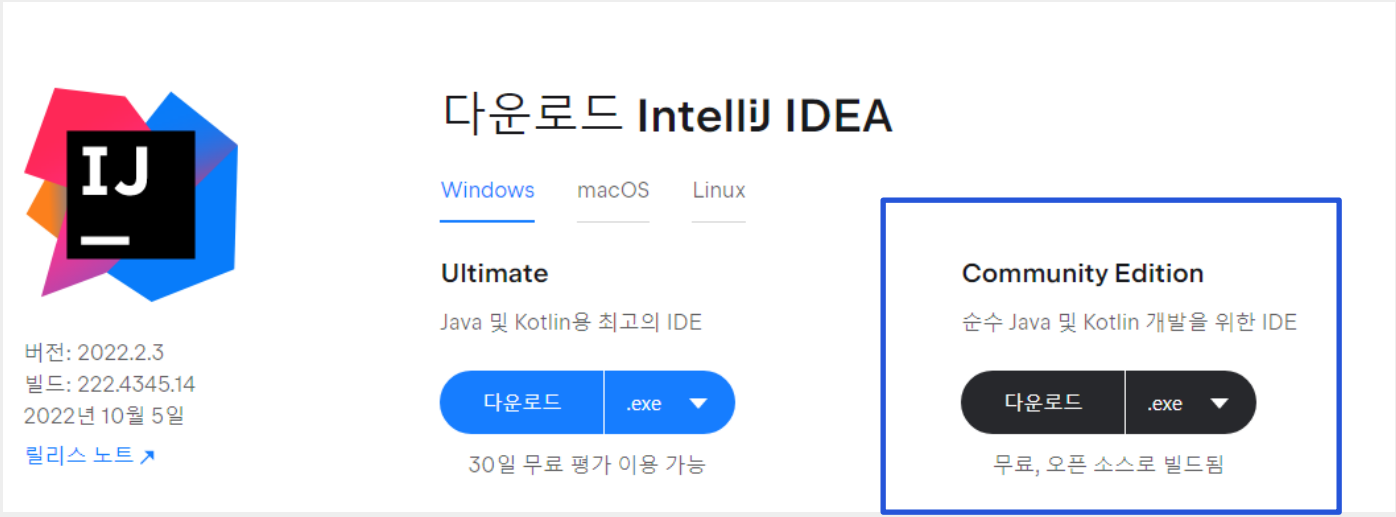
JDK(Java) / IDE(IntelliJ)

JDK(Java Development Kit): 자바 개발도구

IDE(Integrated Development Environment): 통합 개발 환경 ex) Eclipse, IntelliJ

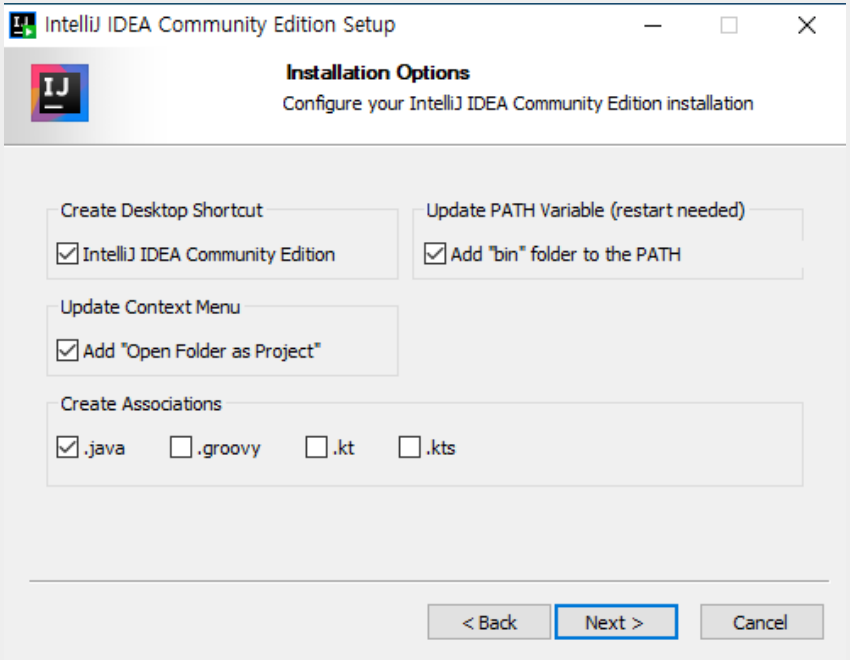
개발환경구축

IntelliJ 다운로드



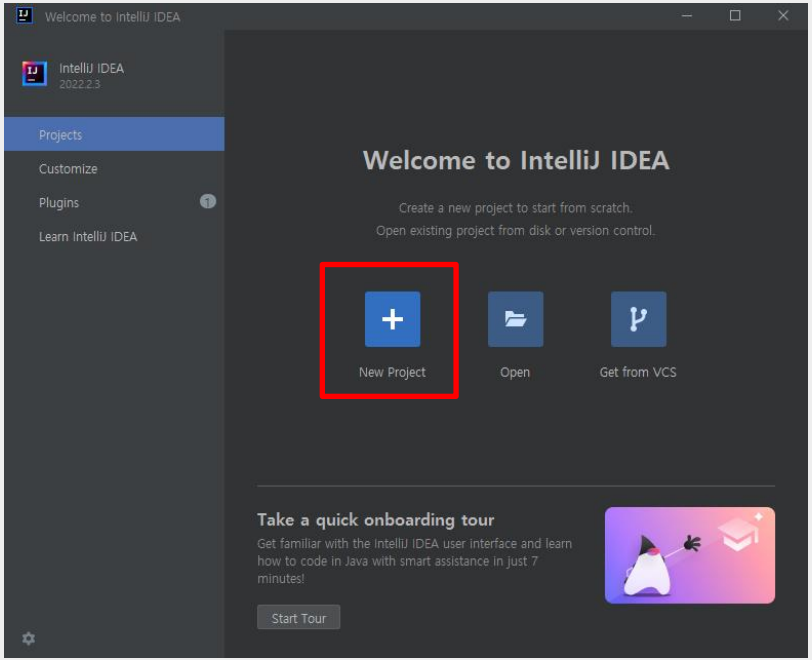
개발환경구축

IntelliJ 다운로드



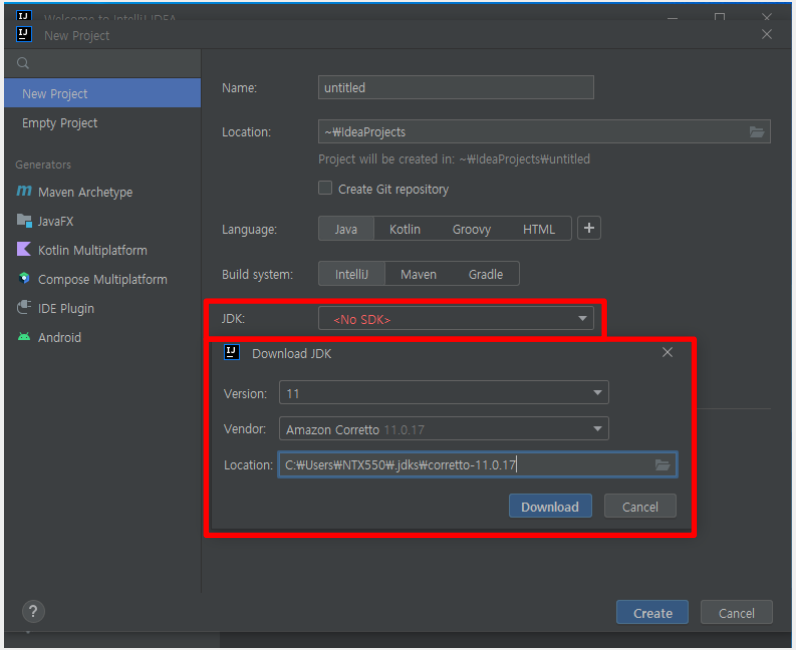
개발환경구축

JDK 다운로드



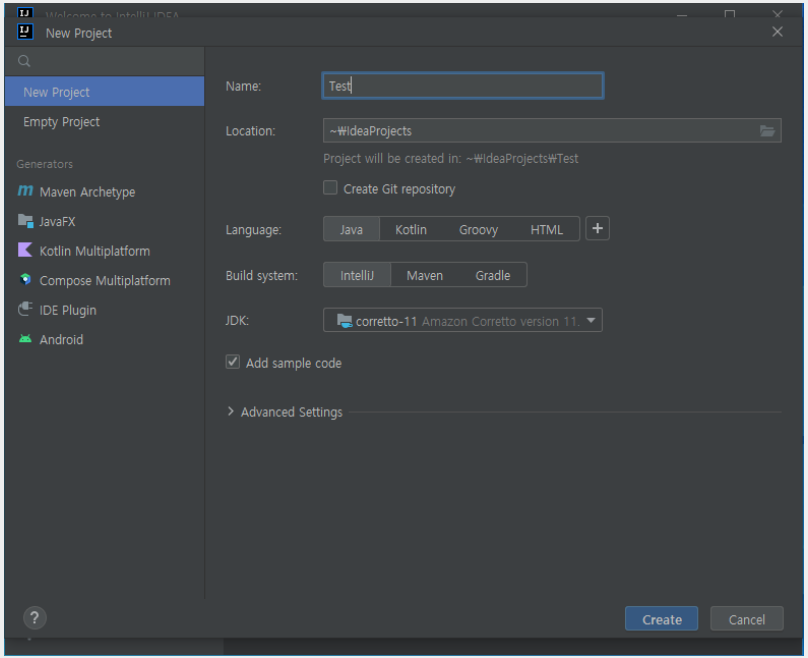
개발환경구축

JDK 다운로드



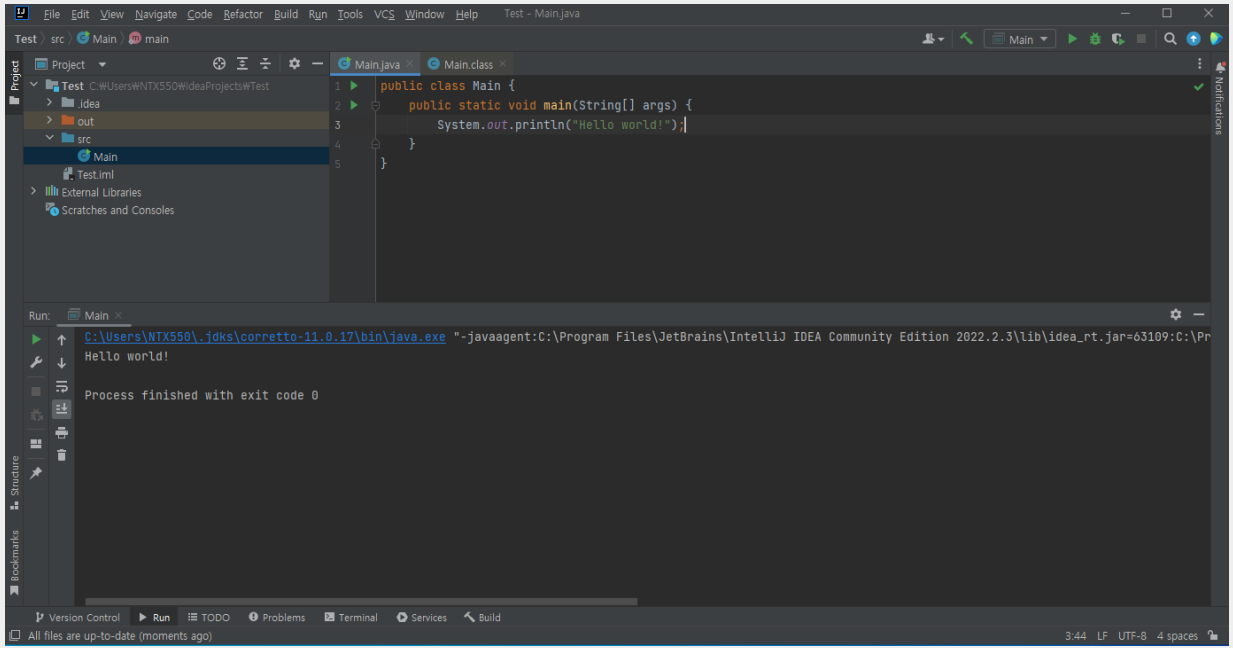
개발환경구축

JDK 다운로드



개발환경구축

Java 동작 확인



자바 기초 문법

자바 변수 - 기본형

```
데이터타입 변수명 = 값;
```

```
int number = 100;
```

분류	타입
논리형	Boolean
	true와 false 중 하나를 값으로 갖으며 조건식과 논리적 계산에 사용된다.
문자형	char
	문자를 저장하는데 사용되며, 변수에 하나의 문자만 저장이 가능하다.
정수형	short, byte, int, long
	정수를 저장하는데 사용되며, 주로 int가 사용된다. byte는 이진 데이터를 다룰 때 사용되며 short는 C언어와의 호환을 위해서 추가되었다.
실수형	float, double
	실수를 저장하는데 사용되며, 주로 double이 사용된다.

자바 기초 문법

자바 변수 - 참조형

변수에 객체의 주소를 저장한다.
기본형을 제외한 나머지 타입

자바 클래스를 참조형이라고 할 수 있다.
대표적으로 String 클래스 또한 참조형이다.

```
참조형타입 변수명 = new 생성자메서드();
```

```
String abc = new String("abc");
```

변수 선언 실습

변수 선언 실습

다음 조건의 변수를 만들고 출력해보자

자바 출력문

```
System.out.println(출력값);
```

조건

참, 거짓을 저장하는 변수에 true 값 저장

정수를 저장하는 변수에 1234 값 저장

소수를 저장하는 변수에 3.14 값 저장

문자열을 저장하는 변수에 Hello World 값 저장

자바 기초 문법

자바 조건문

```
if(조건식){
    실행문;
}else if(조건식){
    실행문;
}else{
    실행문;
}
```

변수 선언 실습

조건문 실습

다음 상황을 조건문으로 만들고 출력해보자

```
if(조건식){
    실행문;
}else if(조건식){
    실행문;
}else{
    실행문;
}
```

조건

시험 점수 변수를 하나 선언하고,
시험 점수가

90점 이상이면 A
80점 이상이면 B
70점 이상이면 C
60점 이상이면 D
그 외에는 F

를 출력하는 조건문을 만드시오

자바 기초 문법

자바 반복문

```
for(초기화식; 조건식; 증감식){
    실행문;
    실행문;
}
```

```
int total = 0;
for(int i = 1; i <= 100; i++){
    total = total + i;
}
System.out.println(total);
```

1.초기화식은 최초 한 번만 수행합니다.

2.조건식을 수행해서 수행결과가 false라면 for 반복문을 빠져 나갑니다.

3.수행 결과가 true라면 실행문을 수행한다.

4.증감식을 수행한다.

5. 2번부터 4번까지 반복적으로 수행한다

변수 선언 실습

반복문 실습

다음 상황을 조건문으로 만들고 출력해보자

```
for(초기화식; 조건식; 증감식){
    실행문;
    실행문;
}
```

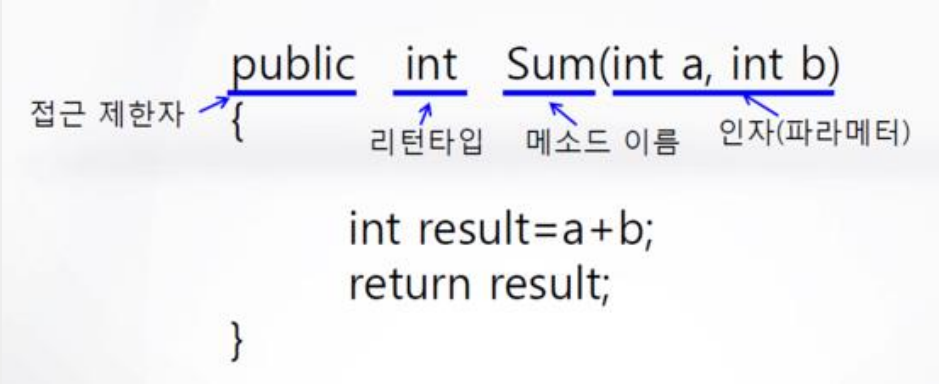
조건

*를 하나씩 더해 나가며, 50개까지 출력하시오.

[예시]
*
**
***
****
*****
******
…

자바 기초 문법

자바 메소드

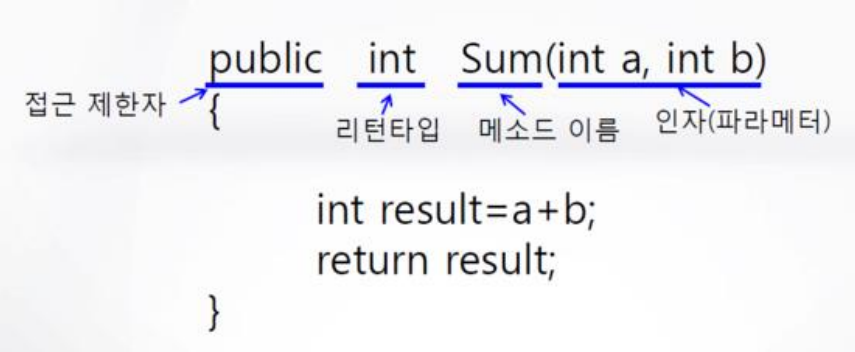


- 접근제어자: 클래스 외부에서 접근이 가능한지 여부를 지정.
  (public – default – protected – private 순으로 외부 접근에 열려 있음)
- 리턴 타입: 메소드 수행이 끝난 후 호출한 곳으로 넘겨줄 값의 타입
- 메소드 이름: 호출할 때 사용할 이름
- 인자: 메소드에 넘겨줄 값

메소드 선언 실습

메소드 실습

다음 상황을 메소드로 만들고 출력해보자



조건

문자 2개를 받아서 더해서 리턴해주며,
모든 Class에서 사용할 수 있는
함수를 만들어주세요.

Git

Git이란?

형상 관리 도구(버전 관리 시스템) 중 하나

소스코드를 효과적으로 관리할 수 있게 해주는 무료 공개 소프트웨어

3대 목적

- 버전 관리: 스냅샷을 통해 파일을 버전 별로 관리
- 백업: 로컬, 원격 저장소에 파일 백업
- 협업: 원격 저장소를 통해 협업 가능

GitHub

GitHub란?

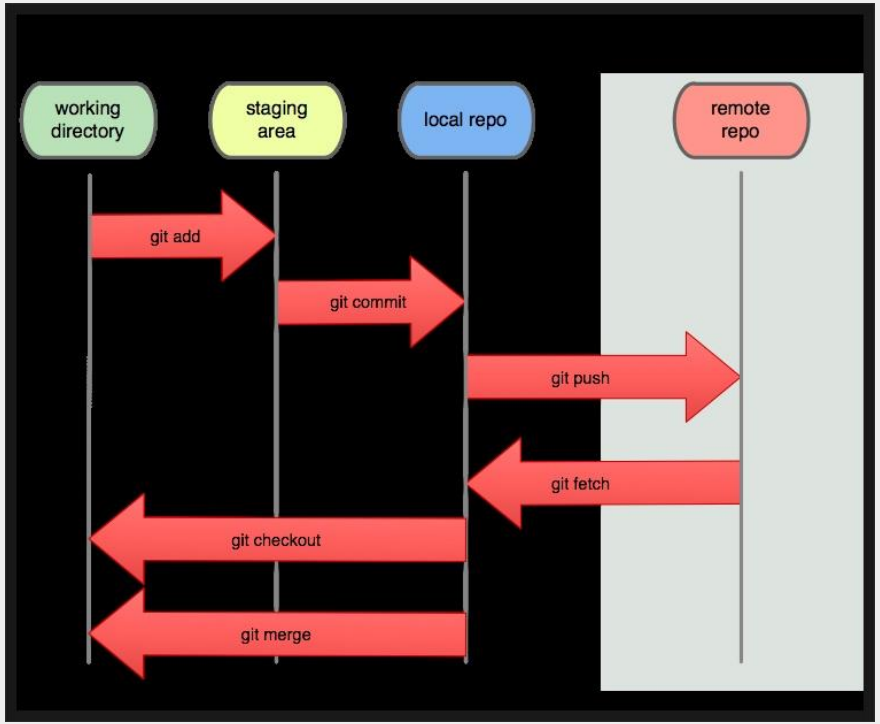
Git 웹 호스팅 시스템

협업하고 있는 코드를 저장하는 서버

ex) GitHub, GitLab, BitBucket

Git 활용법

Git 프로세스



(사진 출처: http://pismute.github.io/whygitisbetter/images/local-remote.png)

Git 명령어

대표적인 Git 명령어(로컬 활용)

```
git config --global user.name "내 이름" #깃에 이름 등록 (한 번만 등록)
git config --global user.email "내 메일" #깃에 내 메일 등록 (한 번만 등록)
git init #현재 파일에 git 폴더 생성
git status #현재 git 관리 현황 출력
git add -A #모든 파일 commit list에 올림
git commit -m "메세지" #commit
git log #깃 히스토리 출력(일련번호 확인 가능)
git reset 일련번호 --hard #되돌리기.
git revert 일련번호 #취소할 위치로 되돌아감. 수정된 새로운 커밋 생성
git branch 브랜치명 #브랜치 생성
git branch #브랜치 출력
git checkout 브랜치명 #브랜치 이동
git merge 브랜치명 #현재 브랜치와 지정한 브랜치 merge
git rebase 브랜치명 # merge와 같은 역할. + 재배치효과(한 줄로 정리)
```

Git 명령어

대표적인 Git 명령어(원격 활용)

```
git remote #현 폴더 원격 저장소 현황 확인
git remote add 원격저장소명 url #원격 저장소 url 클론
git push 원격저장소명 브랜치명 #원격 저장소에 브랜치 푸쉬 작업
git pull 원격저장소명 브랜치명 #원격 저장소에 브랜치 풀 작업
git fetch # 원격저장소 정보로 깃 새로고침
git fetch -all # 원격저장소 정보로 깃 새로고침(모든 브랜치)
git checkout -b 브랜치명 #브랜치 생성과 이동 동시에
git branch -a #원격, 로컬 생성된 브랜치 출력하여 확인. -a 빼면 로컬 브랜치만 출력
git checkout -b 브랜치명 원격저장소명/원격브랜치명 #원격저장소 정보를 가져온 브랜치를 생성, 이동
git branch -D 브랜치명 #브랜치 삭제
git push -d 원격명 브랜치명 #원격 브랜치 삭제
```

과제 안내

자바 강의 수강

필수 수강 강의(총 약 10시간 분량)

Section8. 자바 객체 지향 프로그래밍 소개
Section12. 자바 프로그래밍의 참조 자료형
Section14. 자바 객체 지향 프로그래밍으로 되돌아가기
Section15. 자바 프로그래밍의 컬렉션
Section16. 자바 프로그래밍의 제네릭
Section19. 자바 예외처리 소개

심화 선택 강의(총 약 4시간 분량)

Section17. 함수형 프로그래밍
Section18. 쓰레드와 동시성
Section21. 동시 컬렉션의 동시성과 원자 연산

과제 안내

백준 문제 풀이

자바 기본 문법을 익히기 위해 다음 범위까지 문제 풀이

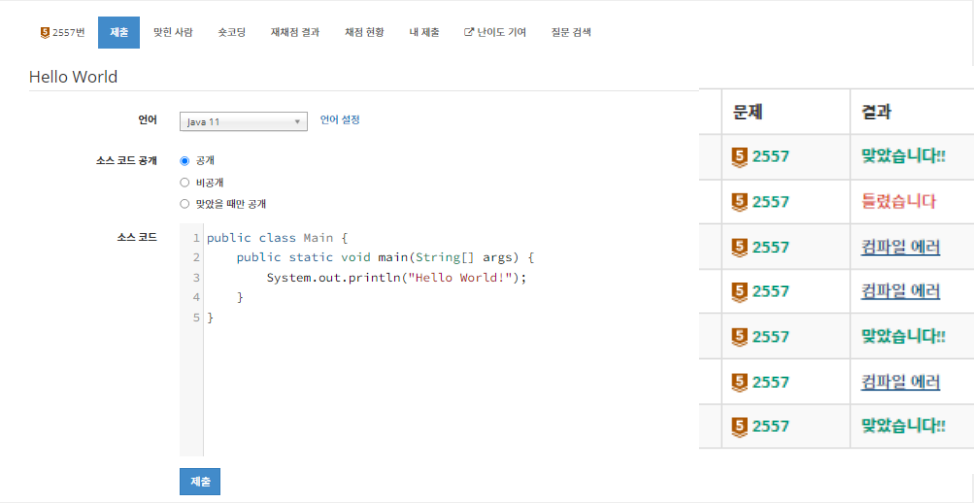
https://www.acmicpc.net/step

단계	제목	설명	정보	총 문제
1	입출력과 사칙연산	입력, 출력과 사칙연산을 연습해 봅시다. Hello World!		14
2	조건문	if 등의 조건문을 사용해 봅시다.		7
3	반복문	for, while 등의 반복문을 사용해 봅시다.		12
4	1차원 배열	배열을 사용해 봅시다.		9
5	함수	함수를 정의하면 코드가 깔끔해지고 관리하기 쉬워집니다.		3
6	문자열	문자열을 다루는 문제들을 해결해 봅시다.		10

과제 안내

백준 문제 풀이

문제 풀이 후, 각 문제의 제출 페이지에서 채점 가능



문제	결과	메모리	시간	언어	코드 길이
2557	맞았습니다!!	13824 KB	120 ms	Java 11 / 수정	116 B
2557	틀렸습니다			Java 11 / 수정	115 B
2557	컴파일 에러			Java 11 / 수정	121 B
2557	컴파일 에러			Java 11 / 수정	120 B
2557	맞았습니다!!	28776 KB	72 ms	Python 3 / 수정	21 B
2557	컴파일 에러			Python 3 / 수정	19 B
2557	맞았습니다!!	12652 KB	76 ms	Java 8 / 수정	108 B

과제 안내

요구사항 개발 및 Git 활용

자바의 객체지향 프로그래밍을 최대한 활용하여 해당 요구사항을 개발 후 깃헙 저장소에 업로드

- 최상위 고객 객체를 추상클래스로 만들고, 공통 멤버변수(이름, 등급, 할인방식)와 가격을 계산하는 계산 메서드, 유저 정보를 출력하는 메서드 2가지를 만든다.
- 유저 정보를 출력하는 메서드를 추상메서드로 선언하고, 웹에서 접속하는 유저와 모바일에서 접속하는 유저를 구분하여 정보가 출력되는 상단에 접속 경로를 출력해야 한다.
- 등급은 enum을 활용해서 BRONZE, SILVER, GOLD 등급으로 관리하고, 각 등급마다 할인 비율인 0.1, 0.2, 0.3을 적용하도록 한다.
- 할인방식은 금액을 받으면 계산 결과를 반환하는 메서드를 가진 인터페이스로 설계하고 보너스 방식과, 할인 방식이 있다.
- 보너스 방식은 할인 비율만큼 보너스 포인트에 더하고, 할인 방식은 지불 금액에서 곧바로 할인 비율만큼 금액을 차감해주는 방식이다.
- 계산 메서드는 고객 객체에 있는 할인방식을 활용하여 계산되도록 한다.
- - 고객의 속성은 Getter, Setter로만 접근 가능하게 해야 한다.
- - 고객은 이름 속성으로 동일한지 비교가 가능하다. (equals, hashcode 활용)

과제 안내

요구사항 개발 및 Git 활용

자바의 객체지향 프로그래밍을 최대한 활용하여 해당 요구사항을 개발 후 깃헙 저장소에 업로드

실행부 Main 메서드는 다음과 같은 흐름으로 구성

```java
public class Main {

    public static void main(String[] args) {

        // 세일 서비스 객체 생성

        // 고객 객체 생성

        // 고객 리스트 생성, 추가

        // 고객 정보 조회 & 가격 계산

    }

}
```

main 함수 실행시 각 회원의 결과 출력 화면은 다음과 같이 출력되어야 한다.

```
======================
접속 경로: Web
이름: 이름6
등급: GOLD
결제 방식: 할인 결제 방식
결제 금액: 7000
======================
접속 경로: Mobile
이름: 이름7
등급: BRONZE
결제 방식: 보너스 결제 방식
보너스: 1000
결제 금액: 10000
======================
```

Hello World!