



개요



- 객체지향 프로그램에 대한 이해를 한다.
- 자바 프로그래밍을 위한 설치 프로그램을 알고, 개발환경을 만들 수 있다.
- 자바 프로그램의 특징을 이해하고, 환경에 따른 프로그램 설치의 장.단점을 안다.



# 생각해봅시다 :

- 프로그래밍 언어의 수행 과정은 어떤 프로세스를 거치는가?
  - 사람의 언어, 기계어(**0 - on ,1 - off**)
  - 사람의 언어 → 기계어로 변환 과정 : **compile**
  - 기계어 → **memory(RAM)** → **CPU** → 원하는 결과를 처리.
- **Binary data - 2진 데이터**
- 플랫폼(**OS**) 독립적이란 무엇일까?
  - **Operating System**
  - **Visual C++ : windows OS** 지원
  - 고객의 사용 운영체제
  - **JVM (java virtual machine)** : 바이트 코드가 운영체제에 맞게 수행.



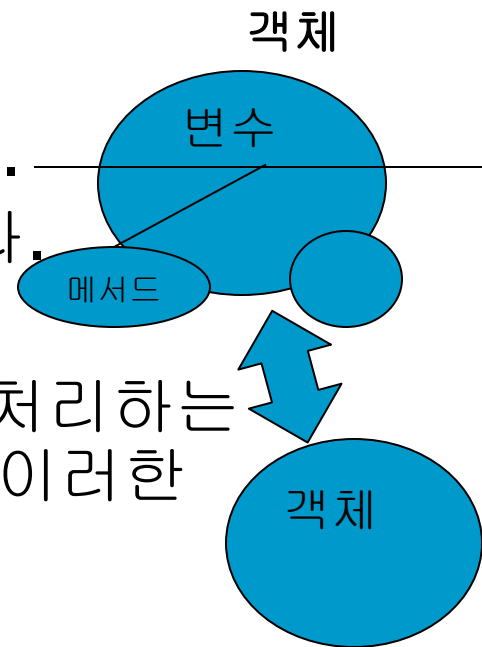
# 자바란? :

## ■ 자바의 시작

- 현재는 오라클로 통합된 썬 마이크로 시스템즈의 제임스 고스링에 의해 개발된 객체 지향 프로그래밍 언어이다. **1995**년에 발표되어 현재 많은 **Sever** 프로그램이나 웹프로그래밍에 활용되고 있다.

## ■ 특징

- 플랫폼 독립적(이식성인 높은) 언어이다.
  - 운영체제 상관없이 수행되는 언어이다.
- 객체지향 프로그래밍 언어이다.
  - 가장 설계나 수행과정을 효과적으로 처리하는 것이 **Object**개념(사물, 물건)이기에 이러한 기초로 시작되었음.





## ■ 특징

- 분산 네트워크 기술을 지원
  - 원격 메소드 호출 <RMI : (Remote Method Invocation), CORBA(Common Request Broker Architecture)> : 네트워크상 다른 computer(서버)를 java 프로그래밍으로 연결해서 처리하는 것.
- 다중 쓰레드를 지원한다. - ex) 네트워크상에 한번에 여러 클라이언트 접속. 하나의 프로세스 안에 여러 단위의 요청 내용을 처리.
- 보안 기능이 지원한다.
  - 접근에 대한 권한을 변수로 처리 - Access Modifier(접근제어자)
- 메모리를 자동 관리
  - 실제 메모리 주소까지 관리하지 않고, 자동으로 메모리를 관리하게끔 설계, 쓰레기 수집기(Garbage collector)를 통해 사용하지 않은 메모리 자동관리.



## 자바란? :

- 개발 플랫폼에 따른 3가지 형태의 **java**
  - **java SE(Standard Edition) :**
    - 데스크탑에서 실행되는 응용 프로그램 개발 사용
  - **java EE(Enterprise Edition)**
    - 대규모 엔터프라이즈 기반 애플리케이션 개발
    - 서버와 클라이언트 분산에서 개발 시 활용
  - **java ME(Micro Edition)**
    - 모바일 및 임베디드(특정한 기기에 포함된 프로그램) 기반 애플리케이션 개발환경에서 활용



## 자바 개발 환경 구축 :

- 자바 개발 도구(JDK) 설치 : JRE+개발을 위한 도구
  - java development kit
    - jvm(자바가상 기계), 라이브러리 API(손쉽게 프로그램을 할 수 있는 클래스모음), 컴파일러 등 개발 도구가 포함.
- 자바개발 환경(JRE)
  - java runtime environment
    - jvm + 표준 클래스 라이브러리.
- 오라클(<http://www.oracle.com>) 설치 사이트
  - 운영체제에 맞는 jdk를 다운로드( 64/32 CPU)

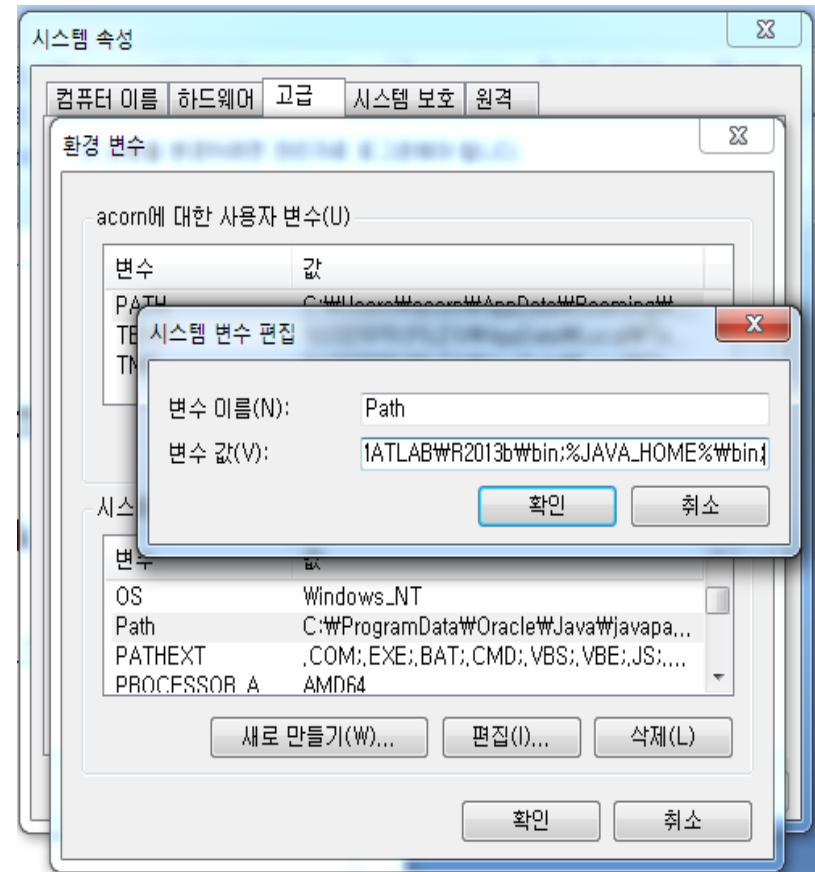
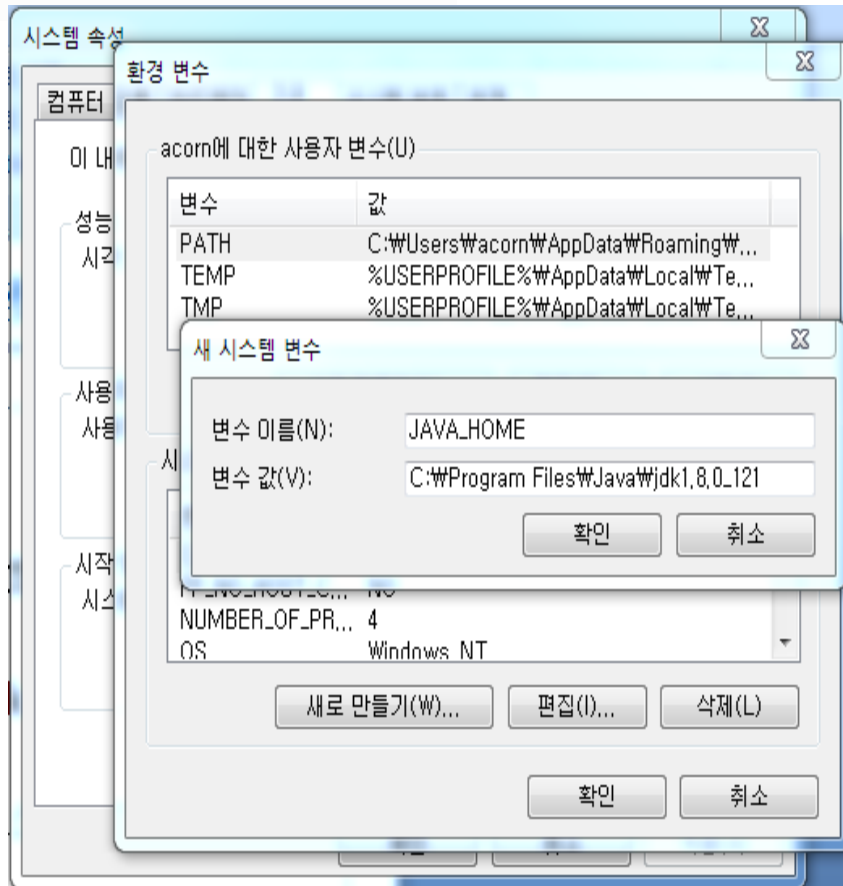


## 실행 환경 setting :

- java 설치
    - cmd 창에서 java 실행.
  - JAVA\_HOME 환경변수로 setting하기.
    - java 설치된 곳 확인..
      - C:\Program Files\Java\jdk1.8.0\_121
    - 시스템변수에 등록.
      - 윈도우 시작 → 제어판 → 시스템 및 보안 → 시스템 → 고급시스템 설정
  - 시스템 변수 Path에서 추가
    - ;%JAVA\_HOME%\bin;
-



# 환경변수 setting



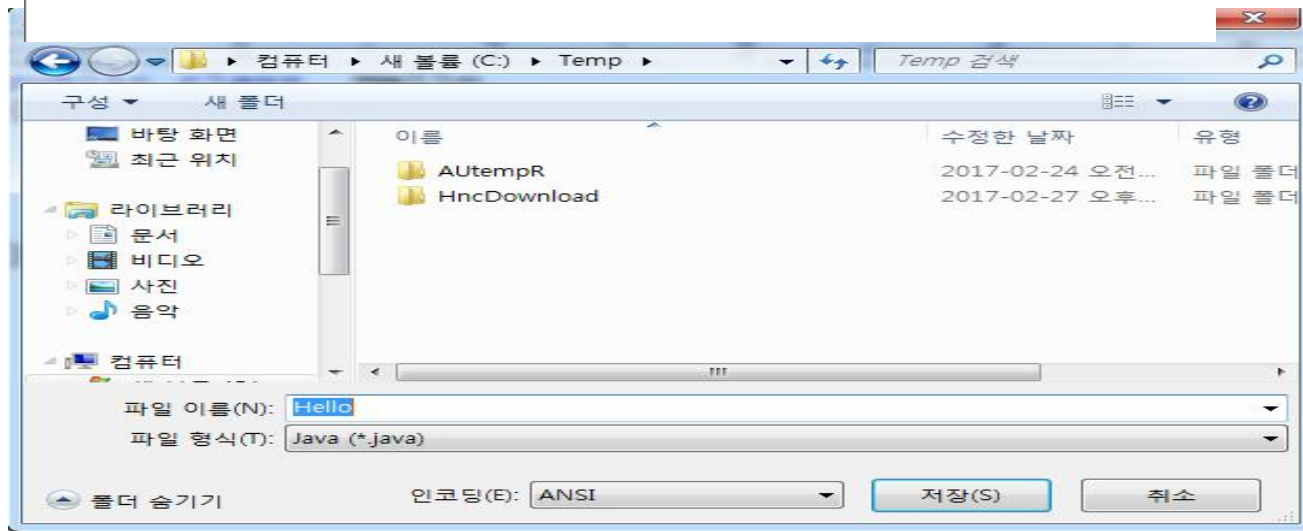


# 자바프로그램 개발 순서 :

- 소스 작성에서 실행
  - java 소스 파일 작성
    - Hello.java : main안에  
System.out.println("안녕하세요");
  - 컴파일러(javac.exe)로 바이트 코드(.class-기계어)생성
    - javac Hello.java → Hello.class
  - JVM 구동 명령어(java.exe)로 실행
    - java Hello

# 소스 작성 :

```
1 public class Hello
2 {
3     public static void main(String[] args)
4     {
5         System.out.println("Hello World!");
6     }
7 }
8
```





## 정리 및 남은 과제 :

- jdk와 jre이 어떤 부분에 대해서 차이가 있는가?
  - 자바에서 말하는 플랫폼 독립적이라는 말은 무엇을 의미하는가?
  - 가베지 컬렉션을 통해서 처리되는 것은 무엇인가?
  - 프로그램 설치에서 32/64는 무엇을 의미하고, jdk와 eclipse 버전에 어떤 영향을 미치는가?
  - eclipse에서 workspace는 어떤 역할을 하는가?
  - java프로그램의 실행 단계별 내용을 기술하세요!!
-



**감사합니다 !**

---