<수업 목록>

​

-> 컴파일 / JDK, JRE / JVM / 인코딩 / 프로그램, 소프트웨어 / CLI, GUI / 메인 메소드 / 메소드 호출 / 프로그램의 수행 순서 / 문자열, 정수

​

● 자바(JAVA)

: 컴퓨터 프로그래밍 언어, 정적 자원을 동적으로 변화

​

1) 특징

① 단순(simple)함, 강력함

②객체 지향(Object-Oriented) 언어 ↔ 절차(방법) 지향 언어

: 객체 별로 코드를 작성하고 객체들을 조합하여 전체 프로그램들을 완성하는 프로그램 설계 방법

▷ + : 코드 재사용 편리, 빠르고 신뢰도 높아짐

-절차 지향 언어: 작업 도구에 더 신경써야함,

③ 분산 환경 지원

④ 견고함

⑤ 안전함

⑥ 컴퓨터 구조에 중립적

⑦ 이식성 존재

⑧ 고성능

⑨ 멀티스레딩 지원

⑩ 동적이다(dynamic)

⑪ 기타 장점: 프로그램 크기 감소, 자동 쓰레기 수집 기능 등등

​

2) 에디션(Edition)

① Java SE(Standard Edition)

② Java ME(Mobile Edition)

③ Java EE(Enterprise Edition)

④ Java ME(Micro Edition)

⑤ JavaFX

​

● 프로그램 (Program)

: 컴퓨터를 위한 작업지시서, 컴퓨터가 특정한 작업을 하기 위한 명령어들의 리스트

- 프로그래머(programmer) : 프로그램을 개발하는 사람

-프로그래밍 : 프로그램을 짜는 것, 이를 통해 만들어진 것을 소프트웨어라고 함

▶ 명령어(intstruction)

: CPU가 수행하는 기초적인 0연산(ex. 정수끼리 덧셈, CPU로 정산 가져오기)

​

1) 프로그래밍 언어

: 기계어 대신 인간이 편리하게 지시를 내리기 위한 언어

-기계어 : 컴퓨터가 이해하는 단 하나의 언어, 0과 1로만 구성 되어있음

▶ 소스 코드(source code) : 인간이 프로그램이 언어를 사용하여 원하는 작업을 텍스트로 기술한 것

▶ 소스 파일(source file) : 소스 코드가 파일에 저장된 것

▶ 오브젝트 파일(object file) : 컴파일러가 소스파일을 기계어로 변환해 저장한 파일

​

▷ 언어

: 전기 흐름을 숫자로 표현(이진법), 0 / 1 + 논리적인 연산을 통해 모든 것을 표현, 저급 언어에 서 고급 언어로 발전

-번역기 : 저급 언어와 고급 언어를 번역, 영문자(라틴어)를 대개 사용

-저급 언어 : 기계에 가까운 숫자로 표현한 언어

-고급 언어 : 인간이 좀 더 친화적으로 사용 가능한 언어(java, c#, C++)

-컴파일 언어 : 미리 소스코드를 작성하고, 실행 전에 번역, 번역 후 실행

-스크립트 언어 : 코드를 바로 실행, 코드를 쓸 때마다 바로 번역

​

2) 프로그램의 수행순서

① 소스 파일의 생성

: 에디터를 사용해서 소스 코드를 입력해 파일에 저장

② 컴파일

③ 클래스 적재

④ 바이트 코드 검증

⑤ 실행

​

● 소프트웨어

: 컴퓨터에게 동작 방법을 지시하는 명령어 집합의 모임

​

● 컴파일(Compile)

1) 컴파일러

: 통역을 담당하는 소프트웨어, 인간이 프로그래밍 언어로 작성한 프로그램을 기계어로 바꾸어줌

-소스 파일을 자바 컴파일러 이용하여 바이트 코드로 변환

▶ 바이트 코드(byte code) : 자바 컴파일러가 특정한 컴퓨터가 아닌 가상적인 기계를 위해 생성한 코드

​

● 자바 가상 기계(JVM, Java virtual machine)

: 바이트코드를 해석하고 실행하는 소프트웨어, 작성한 코드를 번역한 이후 동작들을 읽어서 각 컴퓨터에 맞게 서로 다른 동작 방식으로 실행시켜줌

\* 1byte = 8bit

→ 두 단계의 컴파일(자바 컴파일러, 자바가상기계)을 통해 재 컴파일 없이 여러 운영 체제에서 변경없이 실행 가능

= 이식성이 높음

​

● 자바 개발 도구

1) JDK(Java Development Kit)

: 자바 개발 도구, JRE에 추가로 자바 프로그램을 개발하는데 필요한 컴파일러, 디버거와 같은 명령어행 개발 도구를 추가한 것

​

2) JRE(Java Runtime Environment)

: 자바 프로그램을 실행하기 위한 라이브러리, 자바 가상 기계, 기타 컴포넌트 - 실행을 위한 환경

​

3) 다운로드

(1) Java 설치

① 내 pc에서 시스템 종류 확인 후 oracle - resource에서 컴퓨터에 알맞은 파일을 찾아 회원가입 + 라이선스 동의

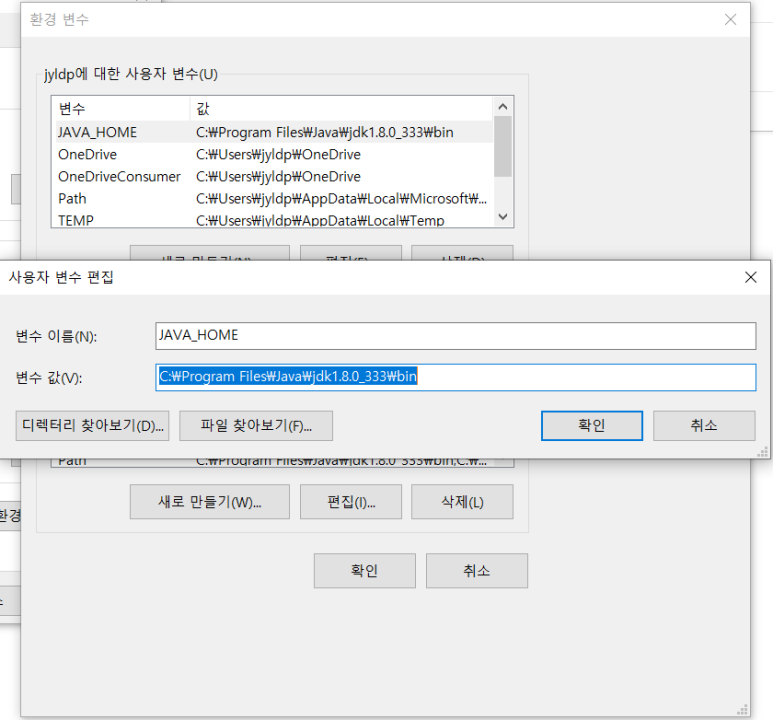
② 실행파일 다운 후 실행 + next 실행 + 기본 설정 유지하고 next + close

(2) 환경 변수 설정

① 검색창에 ' 시스템 환경 변수 편' 검색

② 환경 변수 클릭

③ 사용자 변수에서 JAVA\_HOME 변수에 경로 설정해서 추가해주기

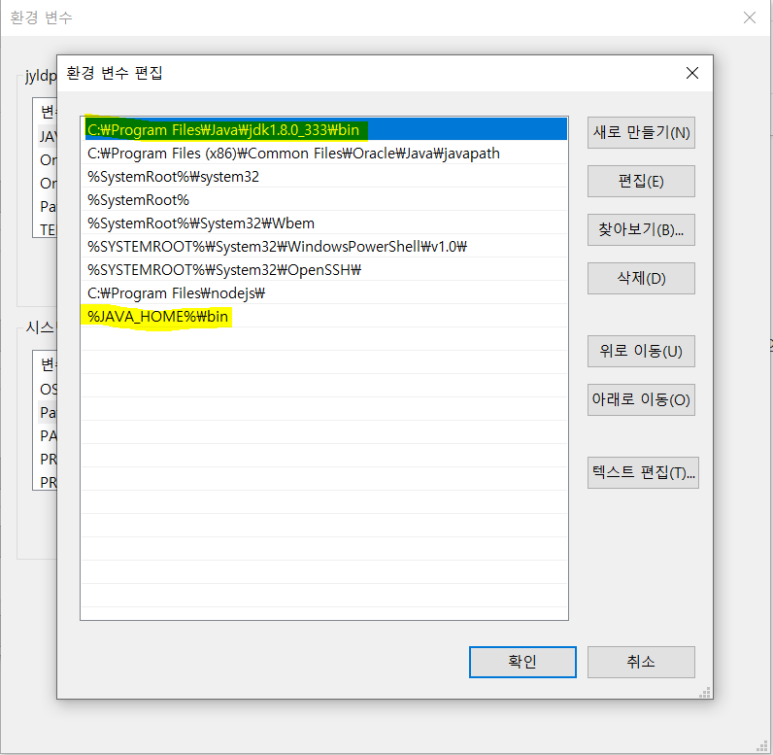


사용자 변수에 경로 추가

④ 시스템 변수에서 Path라는 변수 값에 경로 추가

→ path 원하는 곳으로 이동하는 명령 바로 지정 가능

\* jdk.1.8.0\_(버전 이름)이랑 bin 사이 번호는 다를 수 있음



2가지 경로 추가

○ 통합 개발 환경(intetgrated development environment)

: 소스 에디터, 컴파일러, 디버거 + 비주얼 개발도구

-ex. 이클립스(Eclipse), 넷빈(NetBeans) from Sun

​

● 인코딩(Encoding)

: 코드화, 암호화

어떤 정보를 정해진 규칙(code)에 따라 변환하는 것

▶ 문자 인코딩 : 문자 코드를 전산기기 안에서 0,1로 저장하는 방식, 컴퓨터는 8bit의 이진법 숫자에 대응해 문자를 읽음

-ASCII(American Standard Code for Information Interchange) : 미국정보교환표준부호

-유니코드 : 전세계의 모든 문자를 컴에서 일관되게 표현할 수 있음

\*1byte = 8bit

​

● Command 명령

① cd : change directory

-cd.. : 한 단계 위로

-cd 폴더명 : 폴더로 들어갈 수 있음

-cd\ : 가장 위 폴더로 한 번에 이동

② dir : 폴더명을 찾을 때, 파일과 폴더 목록 확인 가능

③ d : d 폴더로 이동

④ mkdir 이름 : 폴더 생성

⑤ notepad 파일명 / mspaint 파일명 : 메모 파일, 그림판 파일 생성

⑥ javac 파일명 : 원하는 파일 번역 → 파일 생성

​

​

● GUI, CLI

1) GUI(Grafiic User Interace) : 사용자가 편리하게 사용할 수 있도록 입츌력 등의 기능을 알기 쉬운 아이콘 따위의 그래픽 기반으로 나타낸 것

2) CLI(Command Line Interface) : 명령어로 제어, 가상 터미널 또는 터미널을 통해 사용자와 컴퓨터가 상호 작용하는 방식을 뜻함

​

​

\* 확장자명 : 파일을 나타내줌

\* 파일명 = class(단위) 이름

​

​

● 메소드

1) 메소드 : 기능, 명령을 통해 행동하게 하는 것

2) 호출 : 동작을 하게끔 이름을 불러주는 것

3) 메인메소드 : 프로그램의 시작점, 메소드는 프로그램이 해야하는 작업을 받아 수행하고 결과를 내보내는 곳

▶ 절차 수행 : 메소드 안의 문장 순차진행함, 절차대로 수행 → java는 절차적인 프로그래밍

​

​

● 연산자

-수학 연산자 순서와 동일, () → \* / → + - 순으로 적용

▶ +

① 문자열 : 문자끼리 결합

② 숫자 : 덧셈

③ 문자, 숫자 : 결합(-, / 는 문법에서 어긋남, 컴파일 번역 안해줌)

​

● 문자열(String)

: 문자를 대표하는 일련의 자료값을 저장하고 있는 자료형

● 정수(Integer)

: n이 자연수일 때, n + x를 만족하는 모든 x, 모든 n, 0