

- Window > 설정에 관하여
 - 보안 : MS 대부분 백신사 구매, V3 는 개발자 비권장, 기업체 실시간 모니터링 주력
 - 개인정보 : 개인컴퓨터가 아니면 전부 off
 - 앱 삭제
 - Delete : 일반적 삭제는 파일의 위치 및 경로만 지우는 것
 - Shift + Del : 목록을 지움, 경로를 찾기 어렵게 만듦
 - 포렌식 : 목록을 찾는것이나 덮어쓰기시 목록 및 내용 불가
 - 자석 or 전자렌지시 복구 불가
 - 시간 및 언어 : 동기화(Synchronized) 및 비동기화(Asynchronized)
 - 동기화 : 요청과 결과가 동시에 일어나는 것
 - 함수를 호출(요청)하고 그 결과를 기다려 직접 처리(마무리)함
 - 객체 또는 함수에서 다른 함수를 호출시 함수의 결과를 호출한 쪽에서 처리하는 경우
 - 장점 : 쓰레드간 경쟁상황이 발생하지 않아 안정적, 직관적 결과 도출
 - 단점 : 결과가 처리될 때까지 대기해야 함
 - ex) 은행 입출금 작업
 - 비동기화 : 요청과 결과가 동시에 일어나지 않음
 - 함수를 호출하고(요청) 결과를 기다리지 않고 다른 곳에서 처리함
 - 객체 또는 함수에서 다른 함수를 호출시 함수의 결과를 호출한 쪽에서 처리하지 않는 경우
 - callback¹ : 호출한 함수를 받아(call) 특정 조건(이벤트)가 완료되면 되돌려 주는(back) 함수
 - 장점 : 기다리지 않으므로 다른 작업 진행 가능(효율적)
 - 단점 : 복잡성 증가, 충돌발생시 디버깅이 어려워짐
 - Node.js 의 기본 함수들의 호출은 대부분 비동기식
- IntelliJ
 - ARM64.exe 다운은 애플 계열, 모바일폰용으로 CPU사용의 저전력이 장점(Kotlin)
 - PHP : 현존 가장 많이 web상에 구현된(육교)지만 낮은 성능 및 높은 복잡도로 불편하다. ex) 육교보다 지하차도 같은 신기술이 좋지만 비용측면에서 상존한다.
- 경로(Path) : 저장된 곳을 표시 or URL(uniform/universal resource locator, 인터넷의 파일주소)
 - ex) C:\work\bin
- 드라이브(Drive) : 디렉토리의 집합, 물리적,논리적 집합 단위
 - ex) C: or D:
- 디렉토리(Directory) : 폴더, 파일의 집합, 하위 디렉토리를 포함
 - ex) 이때 C: 는 root로 최상위 디렉토리이며, work는 C: 의 하위 디렉토리이다.
- Port : 항구, 자료를 주고 받는 외부 연결 통로(13page 참고)
 - com : 구식, 직렬, 저렴, 느림
 - parallel : 구식, 병렬, 비쌈, 빠름, 24pin
 - PS/2 : 키보드, 마우스
 - USB(Universal Serial Bus) : 범용 직렬 전송
- BUS : 데이터 전송 통로, 주로 컴퓨터 내부 CPU to RAM,GPU,SSD (주로 64bit)
- Deadlock : 처리중 스톱 / Buffering : 처리중 지연 / Error : 처리 불가

¹ 비동기화는 자바에서 멀티스레딩, Runnable, Callable, 콜백(callback), 프라미스(promise), CompletableFuture 등의 방법을 사용하여 구현할 수 있다.

- Embedded : 내부형 컴퓨터, 필요한 기능만 존재
- 부팅(Booting)²³⁴ : OS가 메모리에 로드되기전에 필수 관련 작업을 완료이후 OS를 운용하는 것
 - Booting > Power Supply > MainBoard > ROM(BIOS) > Boot Loader > User Interface
 - Booting : 전원의 인가
 - Power Supply : 전력을 MainBoard에 안정적으로 공급
 - ROM(Read only Memory) : ROM에 저장된 BIOS 프로그램을 실행한다. 읽기전용, 비휘발성 저장, 새 정보 저장 불가
 - RAM(Random Access Memory) : CRUD⁵가 되나 전력 미공급시 휘발됨
 - DRAM : 재충전
 - RDRAM : 빠른 읽기, 휘발성
 - DDRAM : 송수신 동시가능, 128 bit와 유사한 효과
 - BIOS(Basic Input Output System) : 입출력 정상 점검
 - POST(Power On Self Test) : ROM BIOS(CMOS) 에 저장된 부팅 정보로 하드웨어 정상 검사
 - 전원공급확인 > 메모리 검사 > CPU 검사 > 장치 검사(HDD, GPU, 키보드, 마우스등)
 - CPU(Central Process Unit)가 ROM의 메모리 정보를 RAM 에 로드(적재)
 - Memory Heap : 저장공간(SDD, HDD) 정상 점검, 운영체제(OS) 및 필수프로그램들을 메모리에 적재한다.
 - 이때 작업관리자에서 Process 부분에서 적재된(사용중인) 것을 확인 가능
 - 리눅스는 1Gb, 윈도우는 4Gb 로 차이가 존재하는 편
 - Boot Loader : 운영체제를 주 기억장치(RAM) 에 로드하는 것
 - User Interface : 화면에 운영체제의 사용자 인터페이스(로그인 창 등)가 표시
- 8Bit = 1Byte : Bit 는 하나의 공간으로 있으면 1, 없으면 0으로 2진수로 표현
 - $2^{16}=512$: Unicode UTF
 - $2^8=256$: 표현할 수 있는 경우의 수는 최대 256이지만 실사용은 255까지, 1Byte로 Extended BCD 라고도한다. 영어는 표현되나 한글,일어,중어등은 불가능
 - $2^7=128$: ASIC ii , BIS, 데이터통신용, 영문 기준
 - $2^6=128$: Basic Code(BCD)
 - $2^4=32$: Word 0 부터 31개까지, 알파벳은 26개로 처리가능
- 2Byte 는 UTF - 0 (영어), 8(범용), 16(회귀어), 32 등이 있다. 주로 UTF-8이 한글용
- 63342 : 내 컴퓨터에서만 보이는 local host 이자, HTML 코드가 IntelliJ 와 Chrome 간의 포트
 - Java - Spring - IntelliJ 의 연계가 좋은 편
- Ctrl + Shift + i : 도구더보기 > 개발자도구를 보는 단축키
 - Console Programing : 데이터 값이 잘 오는지 확인하기 용이
 - Javascript : 클릭시 작업되는 내용
- 실습
 - 금일도 2시간 가량 기획회의 및 요구사항 분석-> 만들고자하는 내용의 선정(목차) -> ... -> 디자인(맨나중)

² [컴퓨터, 이렇게 부팅된다](#)

³ [컴퓨터 부팅 과정을 메인보드, 롬, 램을 통해 설명해 보자](#)

⁴ [#13 운영체제 | 간단한 부팅 과정](#)

⁵ CRUD : Create(생성), Read(읽기), Update(갱신), Delete(삭제)