java	se
	ee
	me

:0110	2진수
양 -> 2진수 -> 메모리 저장	

*정수를 2진수로 표현하는 방법

1.Sign - Maginetude - 부호 절대값 표시 방식

양수	: +		0	1	0	0	0
음수	: -		**부호 비트**		**절대값**		

10진수 -> 2진수 -> 음/양수

4-2 = 4+(-2)=2 의 개념이 생기

	0	1	0	0	4
: +	1	1	1	0	-2
	1	1	1	0	(-6)
	음수	4	2		

2. 1's 보수

4	0	1	0	0		0	0	1	0	2
-4	1	1	0	1		1	0	1	1	(-2)
						0	0	0	1	

3. 2's 보수

4.Exess k

K값 bias	**비트수-1**			**4-1**					
4비트	**2**	**-1**	**: =**	**2**	**-1**	**: =**	**8 -1**	**: = 7**	

***부동 소숫점 IEEE 규칙**

287				1	**0**	**1**	**1**	**1**	**1**	**2**	**3**
2^10^2	8^10^1	7^10^0		**2^8**	**2^7**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**

106	**: = 01101010**						
128	**64**	**32**	**16**	**8**	**4**	**2**	**1**
0	1	1	0	1	0	1	0

12.375

12	1100	**1100.011**	->	1.100011 * 2^3		
0.375	0.011			1	8	23
0.375 *2	0.75	0		부호비트	지수부	가수부
0.75*2	1.5	1				
0.5*2	: 1.0	1				

*문자를 2진수

MS949 문자 규칙에 따라 2진수로 변환한 것. 그러나 너가 받아서 이 규칙에 따라 읽어라.
내가 나에게 MS949로 encoding 한 date를 보냈다.
규칙에 따라 변환(Encoding) <- > 원래 상태로 복원 (Decoding)

정수값	2의 보수	메모리 저장
부동소숫점	IEEE754번 규칙	메모리 저장
VS CODE	UTF-8 규칙으로 encoding 해서 저장	

VS CODE - UTF8로 저장
javac -encoding UTF-8 " "

고려보

github.com		bitcamp-study	Repo				
깃로드 (git push)		clone					
Local	백업 동기화시킨다.	bitcamp-study		Hello.java			

1 $(cmd에서) git clone https://github.com/아이디/bitcamp-study
2 파일 편집 및 백업 local 저장소에 백업하기
2-1. git add Hello.java <- 백업할 파일 명단을 만든다. (staged 상태)
2-1-1 git add . <-새로 추가하거나 변경하거나 한 모든 파일의 명단을 만들기
2-2 git commit -m " 문자" <-백업 명단에 있는 동록된 파일이나 폴더를 로컬 저장소에 백업한다. (snap shot)
: 깃 메세지 컨벤션
파일작성
- 파일은 저장소 폴더 안에서 작업하기
git config --global user.email "siiy10@naver.com"
git config --global user.name "chunderella"

파일 명단 만들기	git add .
로컬 저장소에 백업	git commit -m " "
깃로드	git push
복제	git clone https://github.com/Chunderella/bitcamp-study.git

	bitcamp-study				
clone-push		clone - pull			
bitcamp-study		bitcamp-study			
commit					
파일					

자바 명령문을 변환한 바이트코드(bytecode) 기계어 비 순한 것(p-code)

Hello.java	컴파일	Hello.class	장기(loading)	JVM		OS
source(소스)	javac.exe	bytecode				
			java.exe			

실행 x (JVM 필요)

bytecode(java 명령) = p-code = portable code
소프트웨어에 의해 처리되기 때문에
보통의 기계어보다 덜 구체적이다.(추상적이다.)

패키지 = Domain 폭넓 ~ 부분

.class	목적파일
.java	소스파일

*S/W

Data		**가공**		**Data Information**

정보를 만들기 위한 재료
가공된 Data로 만든 정보

ex) 연도별 매출 상승 곡선(차트),

literal
데이터를 명령프로그램에서 표현하는 방법

메서드	**클래스**

*S/W 명령문 관리

basic

	함수				
		m1 ()			
		m2 ()	**class A**	m1 ()	
				m2 ()	
				m3 ()	
		m3 ()			**"클래스는 데이터를 분리"**
					- 재사용
					- 네임스페이스
		m4 ()			
		m5 ()	**class B**	m4 ()	
				m5 ()	

"클래스"는 메서드를 관리하기 편도록 분류하는 문법 : C ++ / JAVA

			micro - service	node.js

monolotic 작은 단위를 모아 프로그램을 제작 (메서드 단위로도 충분)

furtional program

```
/*클래스 - 메서드를 물적 단위로 분리하는 기능*/
class Exam84 {
/*메서드 */
public static void main(String[] args) {
System.out.println(100);
System.out.println(0b0110);
System.out.println(012);
```

100 (10진수)

```
System.out.println(0x12);
}
}
```

+자바 소스파일과 클래스 파일 분리

bitcamp-study/		javac -d bin -encoding UTF-8 src/hello.java	**경로를 지정해야함**
	src/hello.java	bin : .class 파일을 둘 폴더를 지정하는 옵션	
	bin/hello.class	-d 폴더가 없으면 만들어라	
		java Exam09를 그냥 하면 경로를 찾을 수 없기때문에 경로를 지정해 줘야함.	
		$ java -classpath bin Hello	
		$ java -cp bin Hello	

+프로젝트 분리

bitcamp-study/				
	java-lang/			
		src/hello.java		
			big.hello.class	

한개의 저장소에 여러개의 프로젝트 만들기

+자바 호환(compatible) 언어

	언어		생성
	.java	컴파일	.class
	.kt	컴파일	.class (문법이 간결함)
	.grooby	컴파일	.class
	main		
		java	
			설정파일
			resources