자바1 수업차시

		문제풀이 - 체계적인 문제 풀이 , 순서도 작성하기
	오전 : 자바설치, 자바로 코드 작성법	public class Hello{
I.	순서도 작성법 : 변수의 역할 , 약속된 기호로 절차표현	public static void main(String[] args) {
	(수 맞추기게임, 1~10까지 합 - 체계적으로 풀어보기)	System.out.println("hello java");
	(X · · · · · · · · · · · · · · · · · ·	}
	체계적인 문제 풀이 방법 - 자료명세 , 처리과정, 순서도	} 작성하기 (하루 바 정도 무제 푸이 시가은 가느다)
2.		TOTAL OF EVEN LEXE
3.	오전 : 문제풀이 발표	변수, 연산자, 제어구조 학습
J.		문제풀이를 코드로 구현해보자
4.	문제풀이 코딩 (풀이한 문제를 코딩해보기)	
		배열의 사용하는 이유
		배열관련문제 풀이
5.	배열	내가 좋아하는 숫자 6개 배열로 저장하자!
		int[] number =new int[3];
		int[] number = new int[] { 7,8,9} ;
		C:메모리관리를 직접한다
		java: 메모리관리를 직접하지 않는다
6.	배열2	
0.	-new의미 (동적메모리 , 정적메모리)	I차원배열
		2차원배열
		3차원배열
		함수:
		명령의 집합 (코드 덩어리)
	함수	함수뽀개기
7.		-청소년 판단 함수 만들기
		-I~IO까지합 함수 만들기
		-오므라이스만드는 함수 만들기
8.	함수2	라이브러리 만들기
	클래스	고객정보를 저장할 구조화된 변수가 필요해 !!
9.	-type으로써의 클래스	
	-절차지향과 객체지향 코드	구조화된 데이터를 처리하는 변수가 없다면?
10	클래스 , 캡슐화 -생성자, this, 패키지 ,접근제어자	개체지하고 C 자사하기
10.	(생성자오버로딩)	객체지향코드 작성하기
		-this 파헤치지

		-생성자는 어떤 용도인가?
		-setter getter는 어떤 용도인가 ?
		-toString 매서드로 객체정보 반환하기
		객체배열을 이용한 간단한 프로그램 작성하기
		일정 정보를 담을 수 있는 Day 클래스 만들기
	객체배열 다루기	class Day{
II.	일정 관리프로그램 작성	
		}
	클래스 – static, nonstatic 이해하고 다루기	매서드에 붙은 static 의미 이해하기
 2 *⊦^	상속(물려받기)	
12	업캐스팅과 다운캐스팅	static 과 static아닌 매서드 사용법 알기
		~~ 객체코딩의 참 맛
		모든 종류의 자동차를 자동차로 다룰 수 있다(부를 수있다)
		자동차로 참조할 수 있다
		모든 종류의 동물을 동물로 다룰 수 있다. (부를 수 있다)
		동물로 참조할 수 있다
	3380 01031 51400	우리반 모든 학생을 학생으로 다룰 수 있다
3 * ^	클래스 - 상속과 다형성 (오버라이드)	
	(2-1-1-1)	모든 세상의 물건을 객체로 다룰 수 있다
		맛있는 다형성 예제 만나기
		나의 다형성에제 만들기
		모든 객체의 조상 Object
		모든 객체는 Object로 다룰 수 있다
	Object, String, StringBuilder,	
4 *ŀ^	Wrrapper클래스	String (char[] + 관련매서드 모음)
14:111	ArrayList	문자열을 처리하는 클래스 String
	(제네릭 맛보기)	
		문자열은 무엇인가?
		MyArrays만들어 보기
5 * t^	추상클래스, 인터페이스 , 객체배열 정렬하기	-MyCompartor 인터페이스
	- 객체는 정렬을 기준을 구현해야 한다	-MyCompare 인터페이스

		Comparable 인터페이스
		Comparator 인터페이스
		예외를 처리한다는 것은?
1/ +601	예외처리하기	- checked Exception
16 * t^l	파일다루기	- unchkecked RuntimeException
프로젝트	4일 , 발표 하루	
		- 채팅프로그램 만들어보기
2 * ^	멀티쓰레드와 네트워크 프로그램	

자바2 수업차시

		세가지 매서드 라이브러리 만들기
		- 별을 3번 출력하는 매서드 만들기
	라이브러리를 만들 때 인터페이스 활용	- 원하는 문자를 3번 출력하는 매서드 만들기
		- 원하는 코드를 3번 실행해 주는 매서드 만들기 (팁! 인터페이스가 필요함)
	제네릭 클래스 복습	- ArrayList , GList, OList
l일차	제네릭 매서드 만들기	
		- ? => 같은 의미 extends Object
	매서드인자에서 와일드카드 사용하기	- extends T T와 T를 상속받은 클래스 의미
		- super T T와 T이상 (부모들)
	인터페이스를 활용한 요리사 찾기프로그램	중식이가능한,일식이가능한, 한식이가능한
	함수형인터페이스란	
		- 이름있는 클래스로 구현
	함수형인터페이스를 구현하는 3가지 방법	- 익명 클래스로 구현
		- 람다식으로 구현
		함수형인터페이스는 주로 매서드의 매개변수로 사용된다 .
2일차	- Runnable : void run()	ArrayList <string> list = new ArrayList<>();</string>
	- Consumer <t> : void accept(T t)</t>	list.add("장미");
	- Supplier <t>: T get()</t>	
	- Predicate <t> : Boolean test(T t)</t>	list.add("후리지야");
	- Function <t,r> : R apply(T t)</t,r>	
		list.forEach(item -> System.out.println(item));
		- List 인터페이스를 구현한 ArrayList
		(배열처럼 쓰고싶을 때, 순서있음, 중복허용)
	Collection – List, Set, Map	- Set 인터페이스를 구현한 HashSet
		(중복을 허용하지 않고 저장할 때)
		-equals 와 hash 를 오버라이딩해야함
	Concenter Liet, Get, Flap	(같은객체로 인식되어야 하나만 저장됨)
		(불근꼭깨도 한격적하지 하시한 제명됨)
		- Map 인터페이스를 구현한 HashMap
		(key, value)로 값을 저장할 때 사용
		String[] stringArray = {"딸기", "포도", "사과"};
3일차		Stream <string> stream2 = Stream.of(stringArray);</string>
		//정렬하고 출력하기 !!
		stream2.sorted().forEach(s -> System.out.println(s));
	스트림	ArrayList <string> stringList = new ArrayList<>();</string>
	· (데이터의 종류에 상관없이	stringList.add("딸기");
	같은방식으로 데이터를 다룰 수 있게 하는 것 !)	stringList.add("포도");
		stringList.add("사과");
		//정렬하고 출력하기 !!
		Stream <string> stream = stringList.stream();</string>
		stream.sorted().forEach(s -> System.out.println(s));

		-reflect
		reflect
	변경에 유리한 코드 작성	(클래스로더를 통해 읽어온 클래스 정보를 사용하는 기술)
	(인터페이스와 reflect가 사용됨)	런타임에 클래스, 인터페이스, 매서드 ,필드 분석 및 조작
	-인터페이스사용으로 느슨한 결합	Class clazz = Class.forName("com.acorn.Acorn"); Acorn obj=
4일차	-reflect로 객체를 동적으로 생성	(Acorn)clazz.getDeclaredConstructor().newInstance();
	객체를 생성하는법	
	- new	
	- 클래스정보를 얻어서 객체를 생성할 수 있다	
		-dependency (의존성)
	DI개념 (의존성 주입)	-Injection (주입)
	애너테이션 유지범위	재귀매서드 공부법 - 작은범위의 예제로 만들고 재귀과정을 그리면서 이해한다
	게 하다 한 휴 이용파	- 별5개 출력 재귀매서드
E 01 ★ŀ	사용자애너테이션 만들기	- I~IO까지 합 재귀매서드
5일차	enum 작성하기	- 배열의 최대값 구하기를 재귀매서드
	재귀매서드 작성하기	- 하노이탑
		- 퀵소트 작성하기