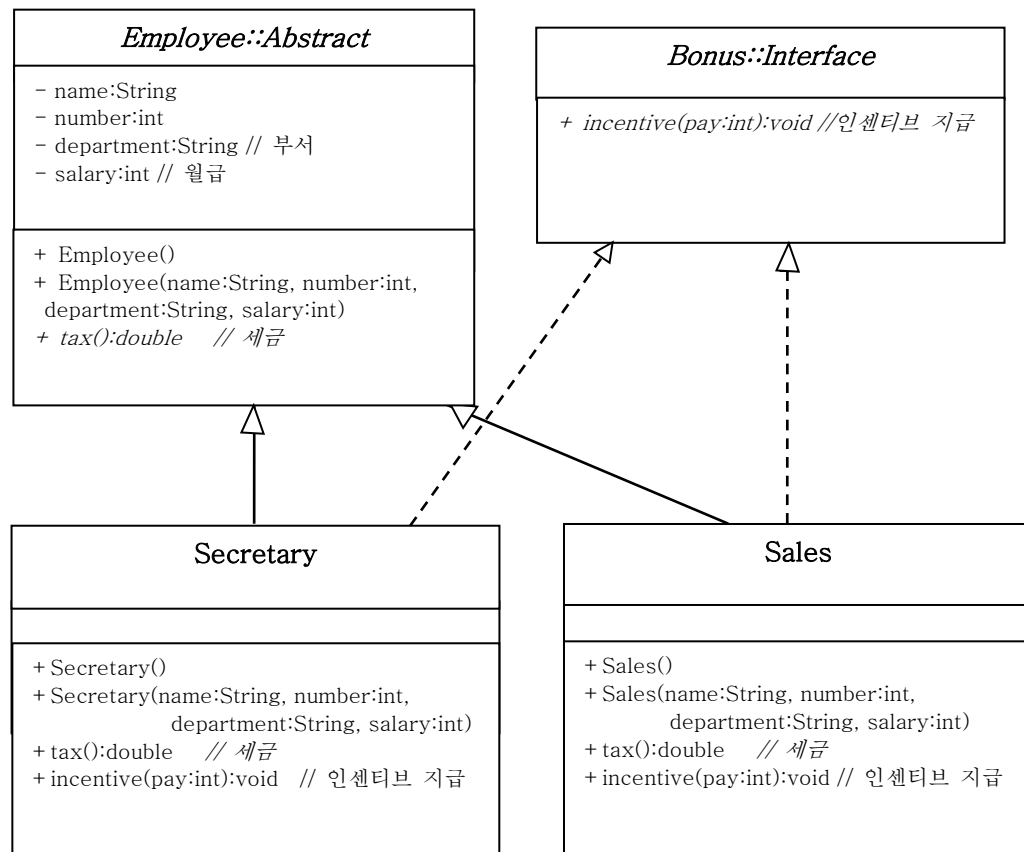


Abstract 클래스와 Interface를 상속하는 두 개의 클래스를 구현하고 실행 결과의 예와 같이 나오도록 프로그램을 작성한다.

1. 사용 데이터

name	number	department	salary
Duke	1	secretary	800
Tuxi	2	sales	1200

2. 클래스 다이어그램



3. 구현 클래스

패키지명	클래스명	메소드	설명
emp	Employee	+Employee()	기본 생성자
		+Employee(name:String, number:int, department:String, salary:int)	4개의 멤버 변수를 받는 생성자
		+tax():double	세금을 리턴한다.
	Bonus	+incentive(pay:int):void	인센티브를 지급한다.
	Secretary	+ Secretary()	기본 생성자
		+ Secretary(name:String, number:int, department:String, salary:int)	4개의 멤버 변수를 받는 생성자
		+tax():double	세금을 리턴한다. salary에 10% 징수
		+incentive(pay:int):void	salary에 계산된 인센티브를 가산한다. 인센티브는 pay의 80% 이다. (pay가100이면 80을 가산)
	Sales	+ Sales()	기본 생성자
		+ Sales(name:String, number:int, department:String, salary:int)	4개의 멤버 변수를 받는 생성자
		+tax():double	세금을 리턴한다. salary에 13% 징수
		+incentive(pay:int):void	salary에 계산된 인센티브를 가산한다. 인센티브는 pay의 120% 이다.(pay가100이면 120을 가산)
	Company	<u>+main(args:String[]):void</u>	main 메서드
		<u>+ printEmployee(emp:Employee[], isTax:boolean):void</u>	Employee 객체의 정보 (직원이름소속 부서명)를 출력한다. 다. isTax 변수에 전달되는 값의 참/거짓 여부에 따라서 텍스트 출력 여부를 결정하여 처리한다.

* 필요 시 getter, setter 메서드를 구현한다.

4. Company 클래스의 구현 내용

```
public class Company {  
  
    public static void main(String args[]) {  
        // Employee 배열에 객체들을 생성하여 저장한다.  
        // printEmployee() 메서드를 호출하여 Employee 배열의 객체들 정보를 출력한다.  
        // 이 때는 세금을 출력하지 않는다.  
        // 모든 직원에게 인센티브로 100씩 지급한다.(직원에 따라 적용되는 인센티브 금액은  
        // 다르다.  
        // printEmployee() 메서드를 호출하여 Employee 배열의 객체들 정보  
        // 와 세금을 출력한다.  
    }  
  
    public static void printEmployee(Employee[] emp, boolean isTax) {  
        // 객체의 정보를 출력한다.  
        // isTax 의 참/거짓 여부에 따라서 텍스 출력 여부를 결정하여 처리한다.  
    }  
}
```

5. 실행 결과

name	department	salary
Duke	secretary	800
Tuxi	sales	1200

인센티브 100 지급

name	department	salary	tax
Duke	secretary	880	88.0
Tuxi	sales	1320	171.6