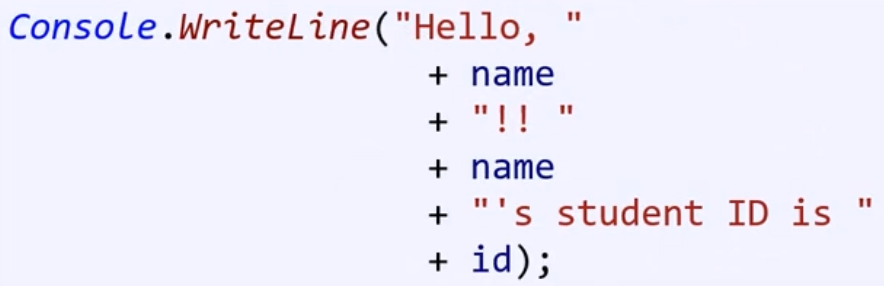
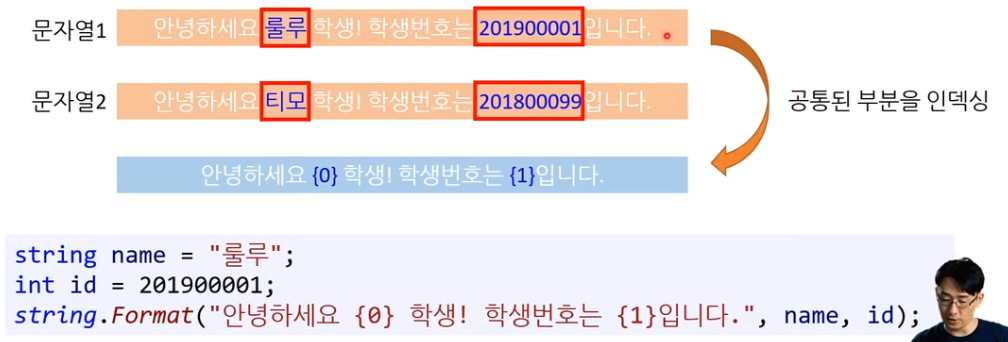
**1. 문자열 포맷팅 1**

* **문자열 결합 연산자 ‘+’를 이용해서 문자열을 만들 수 있음**
* 문자열 + 문자열
* 정수형 + 문자열 / 문자열 + 정수형
* 부동소수형 + 문자열 / 문자열 + 부동소수형
* **결합해야 할 문자열이 많아지면 성능 저하와 가독성이 하락함**
* ex)  
  
* 임시 문자열과 결합되는 다른 피연산자는 새로운 임시 string을 만듦  
  - 다른 피연산자를 만나면 다시 새로운 임시 string을 만듦  
  - 이 과정에서 생긴 이전의 임시 string은 버려지고, 이 과정이 계속되면 성능의 저하를 초래함
* **좀 더 나은 방법**
* ex)텍스트, 테이블이(가) 표시된 사진

  자동 생성된 설명
* **format**
* ex)  
  
* 여기서는 소괄호 안에 있는 데이터를 특정 서식에 맞춰 결합시킴
* 결합한 결과는 문자열로 반환됨  
  - 즉, 소괄호 안에 있는 데이터를 서식에 맞춰 문자열로 바꾸는 기능을 수행
* **string.Format**
* 소괄호 안에 있는 데이터를 서식에 맞춰 문자열로 바꾸는 함수
* 문자열을 반환

**2. 문자열 포맷팅 2-1**

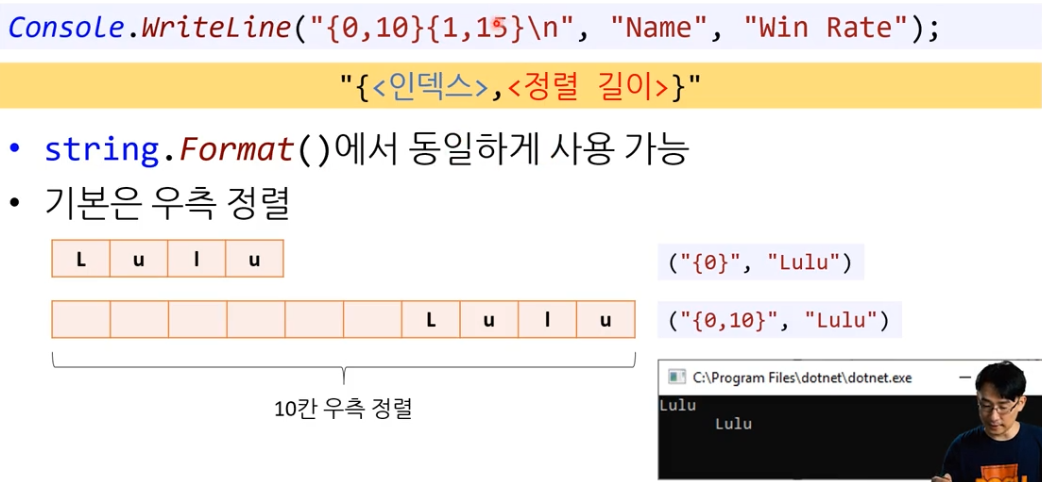
* **문장에 {0}, {1}, … {n}과 같은 인덱싱을 이용해서 원하는 값을 대입할 수 있도록 함**
* ex)  
  
* 가독성의 차이  
  - ex)  
  텍스트이(가) 표시된 사진

  자동 생성된 설명
* **인덱스의 중복?**
* 인덱스는 매개변수의 순서이므로, 각 순서와 인덱스에 맞게 대입됨  
  텍스트이(가) 표시된 사진

  자동 생성된 설명

**3. 문자열 포맷팅 2-2**

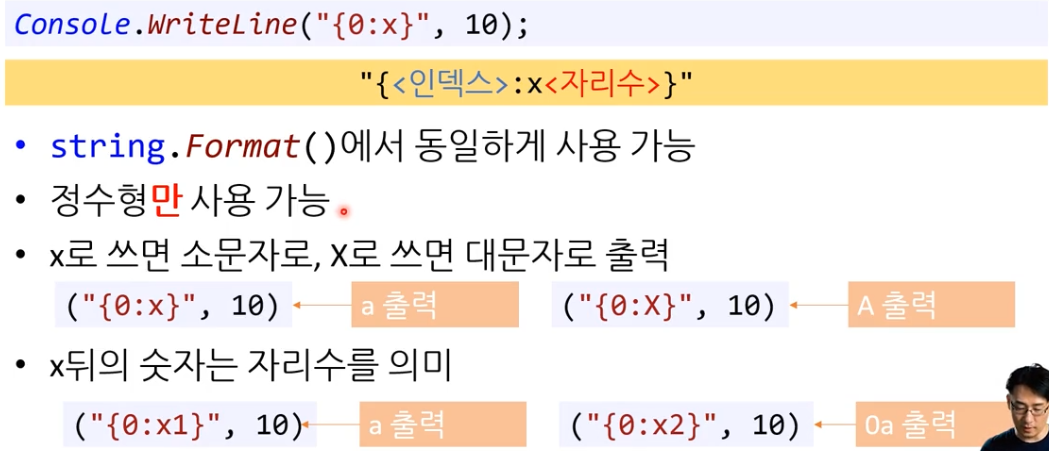
* **스페이스 바를 이용하지 않은 정렬**
* ex)  
  텍스트이(가) 표시된 사진

  자동 생성된 설명
* **문자열 정렬**
* ex)  
  
* 왼쪽 정렬을 하려면 음수를 사용하면 됨
* **소수점을 한자리까지만 출력**
* ex)  
  텍스트이(가) 표시된 사진

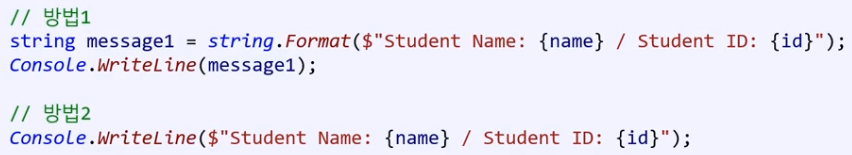
  자동 생성된 설명  
  텍스트이(가) 표시된 사진

  자동 생성된 설명
* **10진수를 16진수로 출력하기**
* ex)  
  텍스트이(가) 표시된 사진

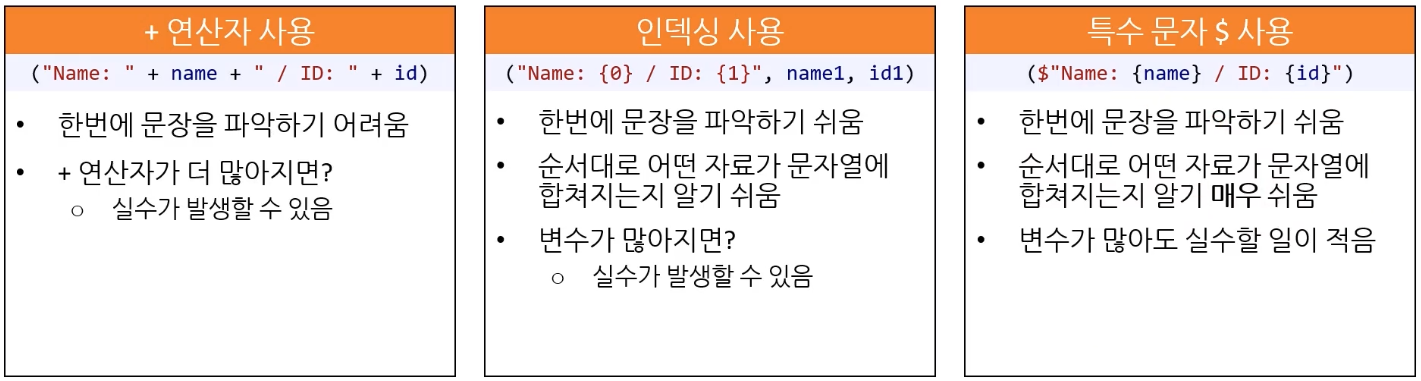
  자동 생성된 설명텍스트이(가) 표시된 사진

  자동 생성된 설명
* ex)  
  

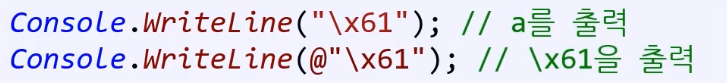
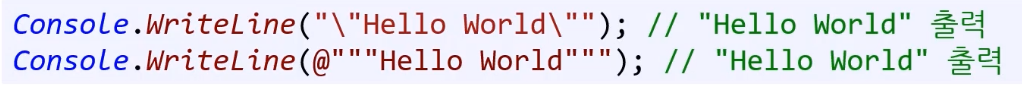
**4. 스트링 포맷팅 3 (문자열 보간)**

* **string.Format**
* ex)  
  
* **문자열 앞에 “$”를 붙임**
* ex)  
  
* **대체될 문자열을 “{(변수명)}”으로 표현**
* ex)  
  

**5. 세 개의 포맷팅 방식의 비교**

* **세 개의 포맷팅 방식의 비교**
* ex)  
  

**6. 적은 그대로 출력하기**

* **문자열에서 특수 문자를 출력할 때 역슬래시를 사용했음**
* **이것을 그대로 출력하고 싶다면 “@” 기호를 사용하면 됨**
* ex) 주어진 문자열을 있는 그대로 출력하는 코드  
  
* **단, 큰따옴표는 조금 다르게 동작함**
* ex)  
  

**6. 키보드 입력 – Console.ReadLine();**

* **enter 키가 입력되기 전까지의 값을 반환**
* **키보드로부터 받은 값은 반드시 문자열 형으로 반환**
* 문자열은 문자도, 숫자도 담을 수 있는 가장 포괄적인 형태
* **키보드로부터 받은 값은 반드시 문자열 형으로 반환**
* 문자열은 문자도, 숫자도 담을 수 있는 가장 포괄적인 형태

**7. 문자열을 정수형으로 변환 – int.Parse();**

* **소괄호 안에 든 문자열을 다른 형식(int)으로 변환(해석)하는 기능을 가짐**
* **문자열이 숫자 외의 값을 가질 경우 예외가 발생함**
* ex)  
  