**s 정답 해설**

(1)

* const xmasDay = new Date(`${new Date().getFullYear()}-12-25:00:00:00+0900`);
* 크리스마스 날짜를 얻기 위한 코드입니다.
* 올해의 연도를 얻기 위해 getFullYear()함수를 사용했습니다. Date 함수는 기본적으로 미리 초로 계산되지만, Date의 인수로 문자열을 전달하면 지정된 형태로 날짜와 시간을 반환합니다. 여기선 [템플릿 리터럴](https://developer.mozilla.org/ko/docs/Web/JavaScript/Reference/Template_literals)을 사용했습니다.
* 위 코드는 Sat Dec 25 2021 00:00:00 GMT+0900 (한국 표준시)를 생성합니다.

(2)

* const now = new Date();
* Date의 인수에 아무것도 적지 않으면 현재 시간을 생성해 줍니다.

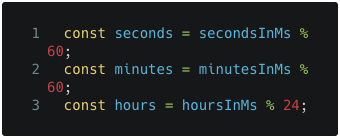
(3)

* const difference = new Date(xmasDay - now);
* Date 함수로 생성한 Date 객체는 연산이 가능합니다. 즉 xmasDay와 now를 연산할 수 있습니다.
* 크리스마스까지 얼마나 남았는지를 알기 위해 xmasDay에서 now를 빼주고 그 값을 difference로 선언해 줍니다.

(4)

* 1초는 1000미리 초입니다. 미리 초 단위의 값을 가지고 있는 difference변수를 1000으로 나눠주어 secondsInMs에 저장해 줍니다.
* 같은 방법으로 secondsInMs를 이용해 minutesInMs를 minutesInMs를 이용해 hoursInMs를 구해줍니다. 1시간은 60분, 1분은 60초, 하루는 24시간으로 나눠줍니다.
* Math.floor()은 나누기 후 생기는 소수점을 없애기 위해 사용합니다.

(5)

* 화면에 출력될 값을 가지고 있는 변수들입니다.
* 출력되는 값들은 크리스마스까지 남은 일수, 남은 시간, 남은 분, 남은 초이므로 나머지 값을 얻을 수 있는 모듈러(%)연산을 사용해야 합니다.
* 모듈러(%)연산을 할 때 시간, 분, 초의 단위에 맞춰 60 또는 24로 나눠주면 됩니다.

(6)

* 남은 시간이 10보다 작아질 경우 앞에 0을 붙여 두 자리로 나타내기 위한 코드입니다. ex) 9h -> 09h
* 위 코드에선 [삼항연산자](https://developer.mozilla.org/ko/docs/Web/JavaScript/Reference/Operators/Conditional_Operator)를 사용하고 있습니다. if 문처럼 생각하면 됩니다.
* 가장 먼저 daysStr을 살펴보면 days < 10이 if 문의 조건과 같은 역할을 합니다.
* 0${days}` : days}부분의 왼쪽은 참일 때 실행할 식이고 오른쪽은 거짓일 때 실행할 식입니다.
* 즉, days가 10보다 작을 때 0${days}가 클 때 days가 출력됩니다.
* 같은 방법으로 hoursStr, minutesStr, secondsStr도 작성해 주면 됩니다.

(7)

* clockTitle.innerHTML = `${daysStr} ${hoursStr} ${minutesStr} ${secondsStr}`;
* clockTitle의 innerHTML프로퍼티에 앞에서 구한 변수들을 넣어 화면에 출력해 주기 위한 코드입니다.

(8)

* setInterval(getTime, 1000);
* setInterval함수를 사용해 getTime 함수를 1000미리 초(1초)마다 실행되게 해줍니다.
* 하지만 이렇게 되면 브라우저가 시작하고 1초간은 시간이 화면에 나타나지 않습니다. 그러므로 setInterval()전에 getTime()을 먼저 한번 실행시켜 줍니다.

**결론**

미리 초 단위가 사용되다 보니 처음에 헷갈리는 부분이 있겠지만, 강의에서 배운 내용과 거의 동일하다 보니 충분히 해결할 수 있었습니다. Date()를 사용하는 방법을 확실하게 연습해 볼 수 있는 챌린지였습니다.