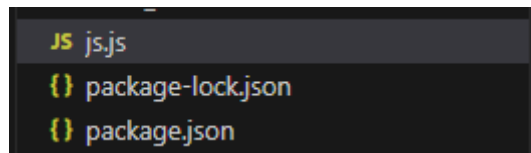


- 기반 환경 설치
- NVM(Node version Manager) 설치 #Node 의 시작  
<https://github.com/coreybutler/nvm-windows#installation--upgrades>  
<https://github.com/coreybutler/nvm-windows/releases>

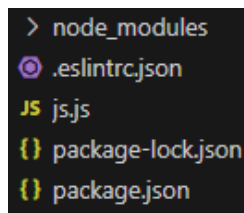
- [nvm-setup.exe](#) 실행
  - C:\Users\student02\AppData\Roaming\nvm
  - CMD : nvm > Running version 1.1.11.
  - 여러 버전 확인 : nvm list available
  - LTS 중 하나 설치 : nvm install 18.14.1
  - LTS 최신버전 설치 : nvm install --lts
  - 설치된 목록 확인 : nvm ls
  - node 사용 : nvm use <version>
  - 사용중인 버전 확인 : nvm current
  - 설치 경로 확인 : which node
  - 버전 삭제 : nvm uninstall <version>
  - 설치 후 환경설정 : npx eslint --init.
  - ESLint / local / npm install eslint
  - global / npm install -g eslint

- npm fund : package를 통한 연결 및 의존성 확인



VS Code Terminal 에서 입력

- npx : 실행 파일 설정
  - 명령어 실행 : npm eslint --init
  - To check syntax and find problems > JavaScript modules (import/export) > None of these > No > Node > JSON > YES > NPM
  - 완료시 아래와 같이 설정됨



- VS code Extensions
  - CMD : c:\work > npx eslint --init
- Node.JS Command
- 예시1)

```
let message = "hello world";  
console.log(message);
```

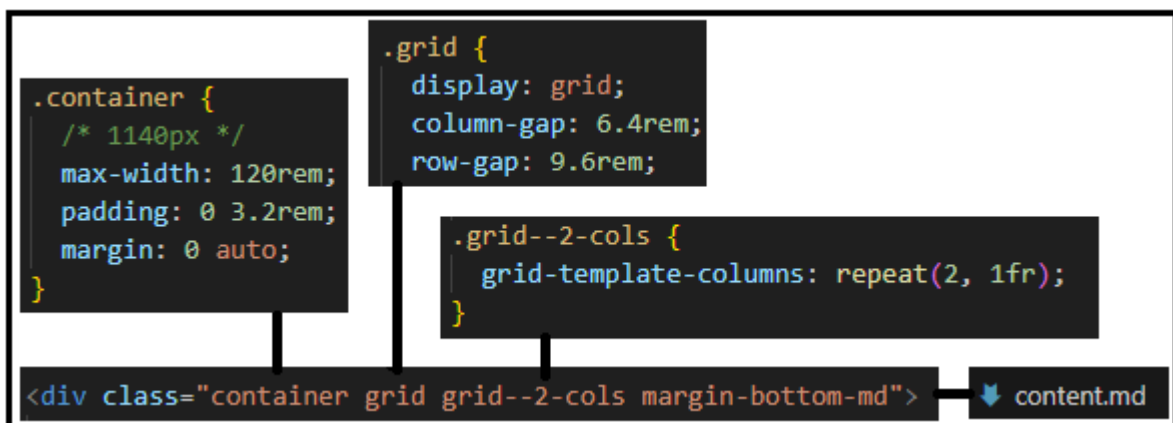
- let A : A를 변수로 설정
- console.log(message) : 화면 출력(=print)

## < CSS - Reverse Engineering >

- 학습목표 : 구현 완료된 HTML/CSS/Javascript 파일을 기반으로 구성, 작성 의도, 새 문법등을 파악하여 기술적인 원리나 구조분석을 통해 발견하는 과정을 경험하기 위함
- Class 미지정



- HTML에서 Class="testimonial"이 확인되나, CSS에는 보이지 않음
- 하위 클래스인 `testimonial - img, text, name` 는 CSS에서 확인됨
  - 우선 `testimonial - img, text, name` 를 구현하기에 앞서 미리 `testimonial`을 선언
  - 의도적이거나 비의도적으로 `testimonial`에 대한 CSS 지정을 생략
    - 의도치않게 생략한 경우를 방지하기 위하여 기술명세서가 필요함
- Class 의 메모화



- 각각 `container / grid / grid--2-cols`에 대하여 CSS에서 지정된 것이 확인됨
- 메모장 역할 : `margin-bottom-md`는 Content.md 파일로 연결됨을 의미
  - 즉 `Class="container grid grid--2-cols"`만 쓰인 것과 동일
  - `margin-bottom-md`는 편의상 작성된 것

- Multi-Device를 위한 호환성 준비

```

/*****
/* BELOW 544px (Phones) */
*****/

@media (max-width: 34em) {
  .grid {
    row-gap: 4.8rem;
  }

  .gallery {
    grid-template-columns: repeat(4, 1fr);
    gap: 1.2rem;
  }
}

/*****
/* BELOW 944px (Tablets) */
*****/

@media (max-width: 59em) {
  html {
    /* 8px / 16px = 0.5 = 50% */
    font-size: 50%;
  }

  .gallery {
    grid-template-columns: repeat(6, 1fr);
  }
}

```

- **.gallery** 가 여러 선언되었으며, 내용도 `repeat(4, 1fr)`과 `repeat(6, 1fr)`로 다름
  - 이는 주석에서 확인되듯이 **Phones** 와 **Tablets** 등 다양한 전자기기마다 적용되기 위함

- 그 외 시간관계상 미정리 목록 (추후 업데이트)

```

transition: all 0.3s;
filter: brightness(0);
opacity: 50%;
align-self: end;
flex-direction: column;

transform: translateY(2.4rem);
grid-template-columns: repeat(4, 1fr);
transform: rotate(45deg);

.cta-img-box {
  background-image: linear-gradient(
    to right bottom,
    rgba(235, 151, 78, 0.35),
    rgba(230, 125, 34, 0.35)
  ),
  url("../img/eating.jpg");
  background-size: cover;
}

z-index: 999;

```

- google font-family

```

<link
  href="https://fonts.googleapis.com/css2?family=Rubik:wght@400;500;600;700&display=swap"
  rel="stylesheet"/>

```

- <https://fonts.googleapis.com/css2?family=Rubik:wght@400;500;600;700&display=swap>

- Display : Flex
  - <https://narup.tistory.com/210>

- **overflow** : 주어진 공간이 모자라 넘치는 콘텐츠에 대한 제어
  - Tip : 초기 디자인 단계에서부터 콘텐츠가 넘치지 않도록 구성하는 것이 권장됨, 스크롤바는 접근성(accessibility) 측면에서 불편한 점이 많은 편이다.
  - `visible : overflow` 속성을 별도로 명시하지 않을 때, 기본 값

```
<div class="lyrics">
  동해 물과 백두산이 마르고 닳도록
  하느님이 보우하사 우리나라 만세.
  무궁화 삼천리 화려 강산 대한 사람,
  대한으로 길이 보전하세.
</div>
```

동해 물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세. 무궁화 삼천리 화려 강산 대한 사람, 대한으로 길이 보전하세.

```
.lyrics { width: 200px; height: 50px; border: 3px solid blue; }
```

- 외곽으로 돌출되어 좋은 디자인이 아닌 편

- `hidden` : HTML 요소 밖으로 돌출된 것 제거, 일부 미학적 용도로 활용

```
.lyrics {
  overflow: hidden;
}
```

동해 물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세. 무궁화 삼천리 화려 강산 대한 사람, 대한으로 길이 보전하세.

- `scroll` : 돌출 및 누락 방지를 위해 스크롤링 기능 적용

```
.lyrics {
  overflow: scroll;
}
```

동해 물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세. 무궁화 삼천리 화려 강산 대한 사람, 대한으로 길이 보전하세.

- `overflow-x` : 가로축만 선택으로 보여줌
- `overflow-y` : 세로축만 선택으로 보여줌

- `auto` : 설정하기 어려울시, 콘텐츠의 크기가 주어진 공간만큼만 스크롤바 적용

```
.lyrics {
  overflow: auto;
}
```

동해 물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세. 무궁화 삼천리 화려 강산 대한 사람, 대한으로 길이 보전하세.

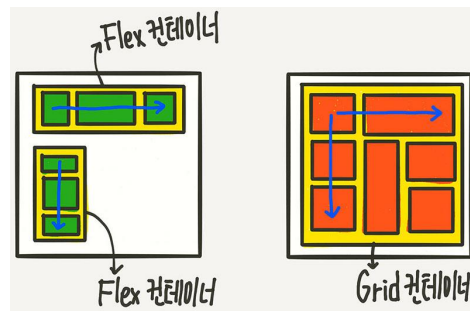
- 콘텐츠에 따라 스크롤바가 생기거나, 충분할 경우 생기지 않음

- `overflow-x : hidden`

```
html {
  overflow-x: hidden;
}
```

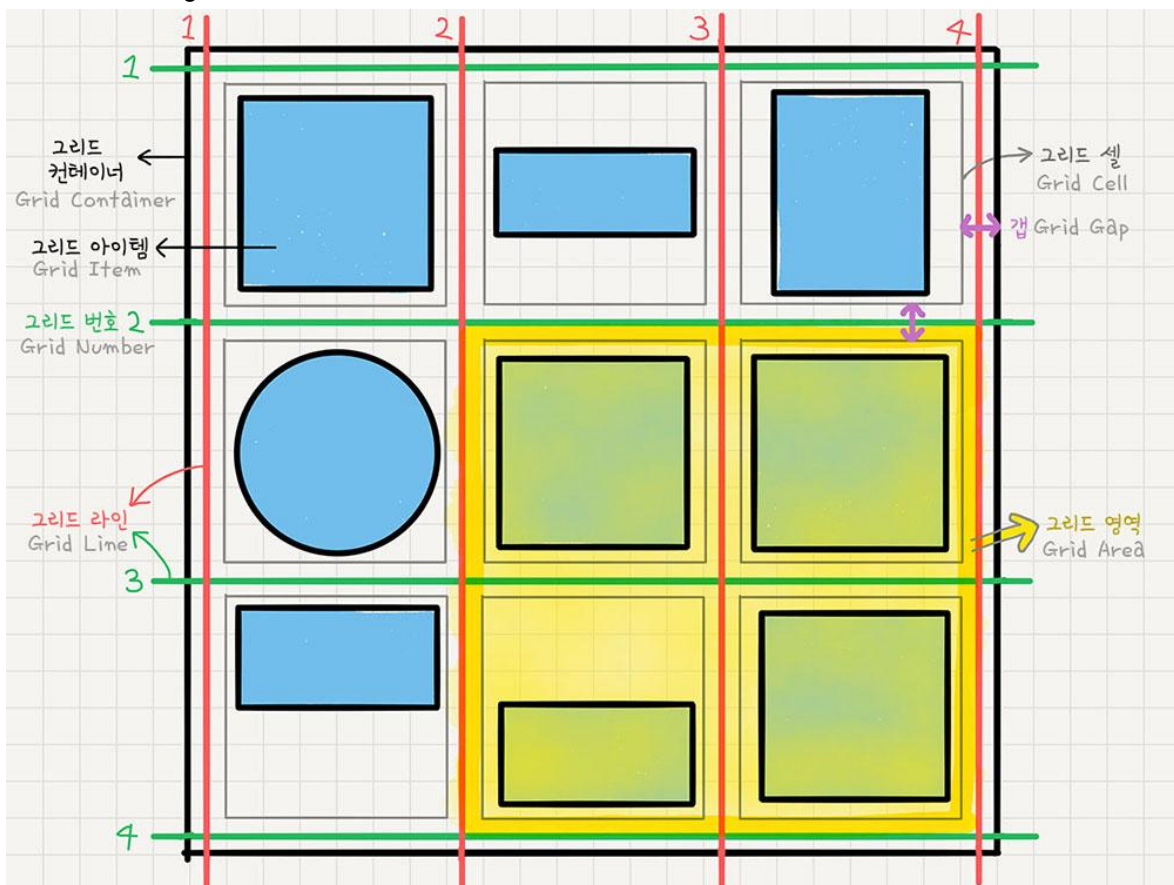
- RE: (의도적으로) 세로의 긴 스크롤만 제공하도록 디자인 됨

- display : grid - HTML 요소를 브라우저의 원하는 위치에 배치하기 위한 속성<sup>1</sup>



*Flex*(1차원) - 한 방향의 레이아웃 / *Grid*(2차원) - 두 방향(가로/세로)의 레이아웃

- grid의 구성 요소




- Grid Container : 전체영역 `<div class="container"></div>`
- Grid Item : Grid 컨테이너의 자식 요소 `<div class="item"></div>`
- Grid Track : Grid의 행(Row) 또는 열(Column)
- Grid Cell : Grid의 한 칸, Grid 아이템 하나가 들어가는 “가상의 칸(들)”
- Grid Line : Grid 셀을 구분하는 선
- Grid Number : Grid 라인의 각 번호
- Grid Gap : Grid 셀 사이의 간격
- Grid Area : Grid 라인으로 둘러싸인 사각형 영역으로, 그리드 셀의 집합

<sup>1</sup> [CSS grid 속성 정리](#)

- display : grid


```
.container{
  display: grid;
  grid-template-columns: repeat(5, 20%);
  grid-template-rows: 200px 200px;
}
```



- 부모 요소인 div.container를 Grid Container(그리드 컨테이너)
- 자식 요소인 div.item들을 Grid Item(그리드 아이템)
- grid-template-columns : grid 아이템들의 가로열 크기 지정
- grid-template-rows : grid 아이템들의 세로열 크기 지정
- repeat(반복횟수, 크기) : 보통 grid 속성을 사용하는 경우 반응형 웹이기 때문에 사용가능한 화면을 기준으로 몇 %씩 크기를 배치할것인지 정할 수 있다. 또는 비율로 결정해주는 fr(fraction) 속성을 사용한다.
  - repeat(5, 20%) 와 repeat(5, 1fr)은 동일하게 20%씩 차지

- grid-gap

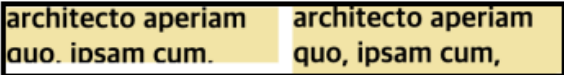
```
.container{
  display: grid;
  grid-template-columns: repeat(5, 1fr);
  grid-auto-rows: minmax(200, auto);
  grid-gap: 10px;
}
```



- grid-row-gap, grid-column-gap, grid-gap 지정 가능
- grid 를 뺀 row-gap, column-gap, gap 이 설계상 편리(IE 구조 통일)

- grid-auto

```
.container{
  display: grid;
  grid-template-columns: repeat(5, 1fr);
  grid-auto-rows: minmax(200, auto);
}
```



- grid-auto-rows : 세로 줄 갯수에 상관없이 지정한 크기대로 설정이 편리
  - grid-auto-rows : 200px == 200px 초과시 이미지 잘림
- minmax : 최소 크기 지정, 아이템 크기가 늘어나면 자동으로 크기 증가

- grid-template-columns: repeat(auto-fill) == 설정된 너비의 허용치까지 채움



- 셀의 폭이 최대 5개지만 셀이 최대 4개라면 최대한 채운 모습

- grid-template-columns: repeat(auto-fit) == 설정된 너비의 허용치에 맞게 채움



- auto-fill 과 auto-fit은 용도에 따른 차이가 존재

- 이외의 grid-auto-flow / align-items<sup>2</sup> / align-self / justify-items / justify-self 등 다양한 기능들이 존재함<sup>3</sup>

<sup>2</sup> [align-items](#) 속성은 콘텐츠 아이템의 내부 상하관계 정렬 상태를 설정합니다.

<sup>3</sup> [이번에야말로 CSS Grid를 익혀보자](#) - 1분 코딩