

# 박정모 VanillaJS - HTML 생성하기 연습문제

---

## 문제1.

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>06-collapse</title>
    <style type="text/css">
      .collapsible-title {
        background-color: #777;
        color: white;
        cursor: pointer;
        padding: 18px;
        width: 100%;
        border: none;
        font-weight: normal;
        margin: 0;
        text-align: left;
        outline: none;
        font-size: 15px;
      }

      .active,
      .collapsible-title:hover {
        background-color: #555;
      }

      .content {
        padding: 0 18px;
        max-height: 0;
        overflow: hidden;
        transition: max-height 0.2s ease-out;
        background-color: #f1f1f1;
      }
    </style>
  </head>
  <body>
    <h2>Animated Collapsibles</h2>
    <p>
      A Collapsible:
      <button type="button" id="btn">절대 누르지 마시오</button>
    </p>

    <hr />

    <div class="collapse">
```

```

    <h1 class="collapsible-title">Open Collapsible</h1>
    <div class="content">
      <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
        do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
        veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
        consequat.
      </p>
    </div>
  </div>
  <div class="collapse">
    <h1 class="collapsible-title">Open Collapsible</h1>
    <div class="content">
      <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
        do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
        veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
        consequat.
      </p>
    </div>
  </div>
</div>

<script>
  function toggle(target) {
    target.classList.toggle("active");
    const parent = target.closest(".collapse");
    const content = parent.querySelector(".content");
    if (content.style.maxHeight) {
      content.style.maxHeight = null;
    } else {
      content.style.maxHeight = content.scrollHeight + "px";
    }
  }

  const target = document.querySelector("body");

  document.querySelector("#btn").addEventListener("click", (e) => {
    target.insertAdjacentHTML(
      "beforeend",
      "<div class='collapse'><h1 class='collapsible-title'
onclick='toggle(this)'>이거라도 누르지 마시오.</h1><div class='content'><p>왜눌렀
대.....</p></div></div>"
    );
  });

  document.querySelectorAll(".collapsible-title").forEach((v, i) => {
    v.addEventListener("click", (e) => {
      // 클릭된 자기 자신
      const current = e.currentTarget;
      // 스스로에게 active클래스에 대한 적용 여부 변경
      current.classList.toggle("active");

      // 클릭된 자신의 상위 요소인 <div class="collapse">
      const parent = current.closest(".collapse");

```

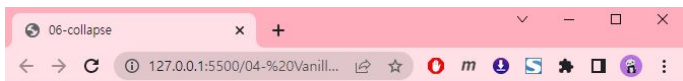
```

// 제어할 대상을 탐색
const content = parent.querySelector(".content");

// content에 maxHeight속성이 있다면 (혹은 0이 아니라면)
if (content.style.maxHeight) {
    content.style.maxHeight = null;
} else {
    content.style.maxHeight = content.scrollHeight + "px";
}
});
});
</script>
</body>
</html>

```

## 실행결과 스크린샷



### Animated Collapsibles

A Collapsible:

