









타깃사이트: 디저트39

 <p>[24년 8월 신메뉴]저칼로리 & 라이스 슬림 요거트...</p> <p>24.08.02</p>	 <p>[24년 8월 신메뉴]저칼로리 딸기 음료 출시!</p> <p>24.08.02</p>	 <p>[24년 8월 신메뉴] 피스타치오크림 음료 3종..</p> <p>24.08.02</p>	 <p>[24년 7월 신메뉴]오레오 셰이크 3종 출시!</p> <p>24.07.29</p>
---	---	--	---

벤치마킹: 메가박스

 <p>12 딸에 대하여</p> <p>예매율 0.7% 개봉일 2024.09.04</p> <p>♡ 99 예매</p>	 <p>12 트위스터스</p> <p>예매율 0.7% 개봉일 2024.08.14</p> <p>♡ 770 예매</p>	 <p>ALL 사랑의 하츠피</p> <p>예매율 0.6% 개봉일 2024.08.07</p> <p>♡ 519 예매</p>	 <p>15 극장판 : 기본_히이라...</p> <p>예매율 0.5% 개봉일 2024.08.28</p> <p>♡ 1.1k 예매</p>
---	---	---	--

웹브라우저 실행결과



[24년 8월 신메뉴]저칼로리 & 라이스 슬림 요거트 출시!

판매 개시일: 24.08.02

♡ 487 구매



[24년 8월 신메뉴]저칼로리 딸기 음료 출시!

판매 개시일: 2024.08.14

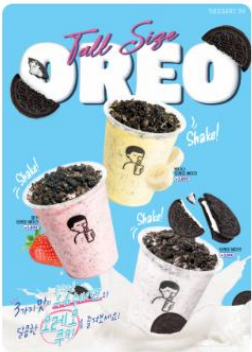
♡ 960 구매



[24년 8월 신메뉴]피스타치오 크림 음료 3종 출시!

판매 개시일: 2024.08.28

♡ 900 구매



[24년 7월 신메뉴]오레오 웨이크 3종 출시!

판매 개시일: 2024.08.07

♡ 990 구매

index.html ㉓

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link rel="shortcut icon" href="https://cdn.dessert39.com/www/img/
common/dessert39_favicon.ico">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>안내사항 1 페이지 | 디저트39</title>
  <link rel="stylesheet" href="../../src/index.css">

  <script src="https://unpkg.com/react@18/umd/react.production.min.js"
crossorigin></script>
  <script src="https://unpkg.com/react-dom@18/umd/react-
dom.production.min.js" crossorigin></script>
  <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>

  <script type="text/babel" src="../../src/card_data.js"></script>
  <script type="text/babel" src="../../src/Card.js"></script>
  <script type="text/babel" src="../../src/App.js"></script>
  <script type="text/babel" src="../../src/index.js"></script>

</head>
<body>
  <div id="root"></div>
</body>

</html>
```

index.css ㉓

```
@charset "utf-8";

@import url("https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.9.0/css/all.min.css");
@font-face {
  font-family: "NanumBarunGothic";
  src: url("https://fastly.jsdelivrivr.net/font-nanumlight/1.0/NanumBarunGothicWebLight.ttf");
}

* {
  box-sizing: border-box;
}

body, ul {
  margin: 0;
}
ul {
  padding: 0;
  list-style: none;
}
body {
  font-family: "NanumBarunGothic", sans-serif;
}
img {
  width: 100%;
  height: 100%;
  object-fit: cover;
  vertical-align: top;
}
a {
  text-decoration: none;
  text-align: center;
  line-height: 35px;
  vertical-align: top;
}
button {
  border: none;
  cursor: pointer;
}

#root {
  display: flex;
```

index.css ⑥

```
    height: 100vh;
    justify-content: center;
    align-items: center;
}
.container {
    display: flex;
}
.card_list {
    width: 235px;
    height: 330px;
    margin: 30px;
}
.card_img {
    position: relative;
    border-radius: 10px;
    overflow: hidden;
}
.card_img::after {
    content: "";
    position: absolute;
    left: 0;
    top: 0;
    right: 0;
    bottom: 0;
    /* width: 100%;
    height: 100%; */
    transition: background 0.7s;
    padding: 20px;
    cursor: pointer;
}
.card_img:hover::after {
    content: "처음 본 순간, 한눈에 반해버렸어! 설레는 운명이 시작된 우리의 첫 만남! 인생의 소울메이트를 꼭 만나길 바라는 '로미' 우연한 기회에 '하츠픽'을 처음 본 뒤, 운명의 인연임을 알아본다. '로미'는 '하츠픽'을 찾기 위해 반대를 무릅쓰고 새로운 길을 떠나지만, 마침내 만나게 된 '하츠픽'은 인간과의 소통을 거부하는데...";
    background: rgba(0, 0, 0, 0.8);
    color: #fff;
    cursor: pointer;
}
.card_age {
    padding: 0 4px;
    border-radius: 3px;
```

index.css ©

```
background: #229C56;
color: #fff;
margin-right: 8px;
}

.card_tit {
display: inline-block;
margin: 0px;
margin: 16px 0;
}

.card_rate::after {
content: "|";
color: #D8D9DB;
padding: 0 8px;
vertical-align: top;
}

.card_btn{
margin-top:12px;
}

.card_btn button {
width: 70px;
height: 35px;
margin-right: 3px;
border-radius: 3px;
border: 1px solid #F3F3F3;
}

.card_btn button::before {
content: "\f004";
font-family: "Font Awesome 5 Free";
margin-right: 8px;
}

.card_btn button.on::before {
color: red;
}

.card_btn a {
display: inline-block;
width: 155px;
height: 35px;
background: palevioletred;
color: #fff;
border-radius: 3px;
}
```

card_data.js ㉠

```
// m: movie
// c: card

const cards = [
  {
    cimg: "https://dessert39.com/data/file/Information/
thumb-3543165506_KJlqgvMQ_66f720f95828fd00e6fe91425f482d0446226f
b2_300×420.png",
    ctit: "[24년 8월 신메뉴]저칼로리 & 라이스 슬림 요거트 출시!",
    cdate: "24.08.02",
    cheart: 487
  },
  {
    cimg: "https://dessert39.com/data/file/Information/
thumb-3543165506_dtqCPEjF_369ba1cb54467331e5b17e1dc2c0517f315ced
d9_300×420.png",
    ctit: "[24년 8월 신메뉴]저칼로리 딸기 음료 출시!",
    cdate: "2024.08.14",
    cheart: 960
  },
  {
    cimg: "https://dessert39.com/data/file/Information/
thumb-3543165506_yT5Q4zbd_ec24c5f7de07e85e0ef8d9040accb7ea94543
730_300×420.png",
    ctit: "[24년 8월 신메뉴]피스타치오크림 음료 3종 출시!",
    cdate: "2024.08.28",
    cheart: 900
  },
  {
    cimg: "https://dessert39.com/data/file/Information/
thumb-3543165506_hes5AWGj_256954940c95be8a5a28571938aa0aace04b
3d65_300×420.png",
    ctit: "[24년 7월 신메뉴]오레오 셰이크 3종 출시!",
    cdate: "2024.08.07",
    cheart: 990
  },
];

// export default cards;
```

Cade.js ㉠

```
function Card(props){
  return (
    <li className="card_list">
      <div className="card_img">
        <img src={props.mimg} alt={props.mtit}> </img>
      </div>
      <div>
        <h3 className="card_tit">{props.mtit}</h3>
      </div>
      <div>
        <span className="card_date">판매 개시일: {props.mdate}</span>
      </div>
      <div className="card_btn">
        <button>{props.mheart}</button>
        <a href="#">구매</a>
      </div>
    </li>
  );
}
```


App.js ㉠

```
// App 컴포넌트
function App(){
  return (
    <ul className="container">
      {cards.map((item, idx) => (
        <Card
          key={idx}
          mimg={item.cimg}
          mtit={item.ctit}
          mage={item.cage}
          mrate={item.crate}
          mdate={item.cdate}
          mheart={item.cheart}
        />
      ))}
    </ul>
  );
}
```

index.js ㉠

```
// import cards from "./card_data";  
// import Card from "./card";  
  
// index.html에 출력  
const container = document.getElementById("root");  
const root = ReactDOM.createRoot(container);  
root.render(<App />);
```