



UI/UX Design with Dash and Dash Mantine Components - 연습 시험 (1)

part 1	연습 시험 (1)
part 2	Answers & Solutions



Cramify turns lecture slides into mock exams and solutions.

연습 시험 (1)

학생 이름: _____

Exam Guidelines:

- Read all questions thoroughly and solve them carefully.
- For proof-based questions, provide complete step-by-step solutions. Answers without detailed proofs will not earn credit.
- Write your answers legibly in the designated spaces. Illegible answers will not be graded. If you write outside these spaces, clearly mark them as your answers.
- Check the point value of each question and manage your time effectively.

Believe in yourself and your preparation. Good luck!



1. Dash Mantine Components (DMC)를 사용하는 Dash 애플리케이션에서 `dmc.MantineProvider`의 역할과 이를 사용하지 않을 경우 발생할 수 있는 문제점을 설명하십시오.

2. Dash Mantine Components 애플리케이션에서 "brandOrange"라는 사용자 정의 색상 팔레트를 정의하고 싶습니다. 이 색상은 밝은 모드에서는 주황색 음영으로, 어두운 모드에서는 더 어두운 주황색 음영으로 표시되어야 합니다. `theme.colors` 및 `theme.primaryShade`를 사용하여 이를 구현하는 방법을 설명하고, 각 모드에 대해 적절한 대비를 보장하기 위해 어떤 음영을 선택해야 하는지 명시하십시오. 또한, 이 "brandOrange" 색상을 기본 강조 색상으로 사용하려면 어떻게 해야 하는지 설명하십시오.

3. 사용자 인터페이스의 특정 섹션에 대한 UI 요소 목록이 제공됩니다. `dmc.Title`, `dmc.Text`, `dmc.Button`, `dmc.Card`를 사용하여 이러한 요소를 시각적 계층 구조 원칙에 따라 정렬하는 방법을 설명하고, 각 컴포넌트에 적절한 속성(예: `order`, `size`, `variant`, `shadow`)을 사용하는 방법을 보여주는 코드 예시를 제공하십시오.

UI 요소:

1. 섹션 제목: "새로운 기능"
2. 소제목: "주요 업데이트"
3. 설명 단락: "이번 업데이트에는 성능 향상 및 버그 수정이 포함됩니다."
4. 액션 버튼: "자세히 알아보기"
5. 관련 정보 카드: "추가 리소스"

4. `dmc.Grid`와 `dmc.GridCol`을 사용하여 반응형 그리드 레이아웃을 구현하고자 합니다. 데스크톱 화면에서는 세 개의 열을 동일한 너비로 표시하고, 태블릿 화면에서는 두 개의 열을 동일한 너비로 표시하고, 모바일 화면에서는 하나의 열로 전체 너비를 차지하도록 하려면 어떻게 `span` 속성을 설정해야 할까요? 또한 두 번째 열에 `offset`을 적용하여 태블릿 화면에서 오른쪽으로 한 칸 이동시키려면 어떻게 `offset` 속성을 설정해야 할까요?

5. `dmc.Image` 컴포넌트를 사용하여 이미지를 표시하는 웹 애플리케이션을 개발 중입니다. 사용자 접근성과 이미지 로딩 실패 시의 사용자 경험을 고려하여 다음 두 가지 시나리오에 대한 코드를 작성하고 각 코드에 대한 설명을 제공하세요.

1. 이미지 로딩이 실패했을 경우, '이미지를 불러올 수 없습니다.'라는 텍스트와 함께 'tabler:photo-off' 아이콘을 표시하는 자리 표시자를 표시합니다.
2. 이미지가 성공적으로 로딩되었을 경우, 이미지 아래에 '출처: Unsplash'라는 캡션을 표시합니다. 이미지의 URL은 'https://unsplash.com/photos/example-image'이고, 대체 텍스트는 '예시 이미지'입니다.

데이터 유형	설명	적절한 차트 유형
A. 시간 경과에 따른 판매량 변화	특정 기간 동안의 판매량 추세	선 그래프
B. 각 제품의 시장 점유율	전체 시장에서 각 제품이 차지하는 비율	원형 차트
C. 여러 도시의 평균 기온 비교	서로 다른 도시의 평균 기온 차이	막대 그래프
D. 제품 가격과 고객 만족도 간의 관계	두 변수 간의 상관관계	분산형 차트

위 표에 제시된 데이터 유형 (A, B, C, D)에 대해 각각 어떤 차트 유형이 가장 적합한지 설명하고, 각 차트 유형을 선택한 이유와 함께 차트의 명확성, 접근성, 적절한 레이블 지정 등을 위한 구체적인 방법을 제시하세요.

7. 사용자가 양식을 제출한 후 성공적으로 저장되었다는 피드백을 제공하기 위해 `dmc.Alert`를 사용하려고 합니다. 성공적인 작업임을 나타내는 적절한 색상과 아이콘을 사용하여, 사용자에게 명확하고 직관적인 피드백을 제공하는 코드를 작성하세요. 또한 5초 후에 자동으로 사라지도록 하고, 사용자가 수동으로 닫을 수 있도록 `withCloseButton` 속성도 설정해야 합니다.



Detailed solution in cramify

- 1) dmc.MantineProvider는 DMC 테마의 핵심으로, 애플리케이션 전체에 테마를 전파하고 스타일 일관성을 유지하며, 색상 구성표 및 전역 스타일을 관리합니다. 이를 사용하지 않으면 일관성 부족, 중복 코드, 테마 적용 어려움, 접근성 문제가 발생할 수 있습니다.

```
2) APP_THEME = {
  "colors": {
    "brandOrange": [
      "#FFF5F0", "#FEE2D5", "#FDD7BC", "#FBBF91", "#FAA766",
      "#FF9140", "#F57C1A", "#D96704", "#BA5200", "#913D00"
    ],
  },
  "primaryColor": "brandOrange",
  "primaryShade": {"light": 4, "dark": 6},
}
```

```
3) import dash_mantine_components as dmc

dmc.Title("새로운 기능", order=1, size="h1")
dmc.Title("주요 업데이트", order=2, size="h2")
dmc.Text("이번 업데이트에는 성능 향상 및 버그 수정이 포함됩니다.")
dmc.Button("자세히 알아보기", variant="filled", color="primary")
dmc.Card(children=[
  dmc.Text("추가 리소스")
], shadow="sm")
```

```
4) dmc.Grid(
  gutter="md",
  children=[
    dmc.GridCol(span={"base": 12, "sm": 6, "md": 4}, children="첫 번째 열"),
    dmc.GridCol(span={"base": 12, "sm": 6, "md": 4}, offset={"base": 0, "sm": 0, "md": 4}, children="두 번째 열"),
    dmc.GridCol(span={"base": 12, "sm": 12, "md": 4}, children="세 번째 열"),
  ]
)
```

- 5) 시나리오 1:

```
dmc.Image(src="", alt="예시 이미지", withPlaceholder=True, placeholder=dmc.Center(style={"height": "100%"}, children=
```

- 시나리오 2:

```
dmc.Image(src="https://unsplash.com/photos/example-image", alt="예시 이미지", caption="출처: Unsplash")
```

- 6) A. 선 그래프 (시간 경과에 따른 추세 표현) B. 원형 차트 (전체 대비 부분 비율 표현) C. 막대 그래프 (여러 항목 비교) D. 분산형 차트 (두 변수 간의 상관관계 표현)

```
7) import dash_mantine_components as dmc
from dash_iconify import DashIconify

dmc.Alert(
  "데이터가 성공적으로 저장되었습니다.",
  title="성공",
  color="green",
  icon=DashIconify(icon="tabler:check-circle"),
  duration=5000,
  withCloseButton=True
)
```



Check detailed solutions in **cramify**