高品質インフォグラフィックHTMLスライド作成プロンプト

## 1. 全体設計

### 1.1. 基本仕様

\* \*\*スライド形式:\*\* 1枚スライド (16:9)

\* \*\*レスポンシブ対応:\*\* 横幅100%、最大幅1600px

\* \*\*レイアウト:\*\* 中央寄せ、情報量に応じて1, 2, 3カラムを自動切り替え

\* 1カラム：幅100%

\* 2カラム：各カラム50%

\* 3カラム：各カラム33%

\* \*\*カラム間隔:\*\* 24px

\* \*\*情報フロー:\*\* 上から下への自然な流れ

\* \*\*フッター:\*\* ページ下部に出典情報を明記

\* \*\*グラフ挿入:\*\* 適切な箇所にグラフを挿入できる設計

\* \*\*スクロールバー:\*\* 可能な限り非表示。やむを得ず表示する場合は最小限に。

### 1.2. デザイン指針

\* \*\*視覚的階層:\*\* 情報の重要度に応じた明確な階層構造

\* \*\*統一感:\*\* 全体を通して一貫性のあるスタイル

\* \*\*簡潔性:\*\* 過剰な装飾を避け、情報を効果的に伝える

\* \*\*訴求力:\*\* 記憶に残りやすく、説得力のあるビジュアル

\* \*\*余白:\*\* 適切に余白を設け、圧迫感を軽減

\* \*\*情報構造:\*\* 論理的で一貫性のある垂直方向の流れ

\* \*\*親しみやすさとプロフェッショナル感:\*\* 要素(罫線、矢印)で親しみやすさを演出しつつ、洗練されたデザインでプロフェッショナルな印象を与えること。

## 2. デザイン仕様

### 2.1. カラースキーム

\* \*\*メインテキスト:\*\* `#334155`

\* \*\*見出し:\*\* `#EE6C8A` (Flory Pink)

\* \*\*サブタイトル:\*\* `#475569`

\* \*\*背景:\*\* 白ベース (`#ffffff`) またはグラデーション (`linear-gradient(135deg, #f8fafc 0%, #f1f5f9 100%)`)

\* \*\*カード背景:\*\* `#ffffff` (白)

\* \*\*ヘッダー背景色:\*\* `#2c3e50`

\* \*\*ヘッダー文字色:\*\* `#ffffff`

\* \*\*フッター背景色:\*\* `#2c3e50`

\* \*\*フッター文字色:\*\* `#ffffff`

\* \*\*カードタイトル背景色:\*\* `#3498db`

\* \*\*カードタイトル文字色:\*\* `#ffffff`

\* \*\*追加カラーパレット:\*\*

```

<palette>

<color name='ビジネス-1' rgb='0A2463' r='10' g='36' b='99' />

<color name='ビジネス-2' rgb='1E56A0' r='30' g='86' b='160' />

<color name='ビジネス-3' rgb='3D84B8' r='61' g='132' b='184' />

<color name='ビジネス-4' rgb='78A6C8' r='120' g='166' b='200' />

<color name='ビジネス-5' rgb='E6F2F5' r='230' g='242' b='245' />

</palette>

```

### 2.2. タイポグラフィ

\* \*\*フォント:\*\* Noto Sans JP, M PLUS 1p, Shippori Minchoを使用

```

<style>

@import url('https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@400;500;700&family=M+PLUS+1p:wght@400;500;700&family=Shippori+Mincho:wght@400;500;700&display=swap');

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: 'Noto Sans JP', sans-serif;

}

body {

background-color: #f8f9fa;

display: flex;

justify-content: center;

align-items: center;

min-height: 100vh;

}

</style>

```

\* \*\*要素別スタイル:\*\*

\* \*\*タイトル:\*\* 36px, `#0A2463`, 太字, シャープな印象（または 28-32px）

\* \*\*サブタイトル:\*\* 22px, `#1E56A0`, セミボールド（または 14-16px）

\* \*\*セクション見出し:\*\* 24px, `#3D84B8`, アイコン付き（または 16-18px）

\* \*\*本文:\*\* 16px, `#333333`, 行間1.6（または 12-13px, 行間1.3-1.4）

### 2.3. レイアウト要素

\* \*\*ヘッダー:\*\* 中央揃えタイトル + 右揃え日付/出典

\* \*\*カラム構成:\*\* 情報量に応じて1, 2, 3 カラムを自動で切り替えること。

\* \*\*カード:\*\* 白背景 (`#ffffff`), 角丸10px, シャドウ

\* \*\*セクション区切り:\*\* 区切り線または背景色(`#E6F2F5`)

\* \*\*レスポンシブ対応:\*\* 横幅100%、最大幅1600pxを維持

## 3. グラフィックレコーディング表現

\* \*\*情報構造:\*\* 重要度に基づいた垂直方向の情報構造

\* \*\*セクション:\*\* 各セクションは独立したまとまりとして表現

\* \*\*強調:\*\*

\* キーワードを青系の色や太字で強調

\* `highlight` クラスを適用 (`#fdf2f4`の背景色)。`highlight-pink`（背景色`#fdf2f4`、文字色`#EE6C8A`）と `highlight-yellow`（背景色`#fff9c4`、文字色`#b59d00`）も使用可能。

\* ポイントや見出しに絵文字（⚡️、⭐️など）を使用

\* \*\*視覚化:\*\*

\* プロフェッショナルなアイコン (📊📈💼⚙️📱など)を使用

\* 幾何学的な図形、グラフ、図表でデータを表現

\* 要素（罫線、矢印）を使用

## 4. コンポーネント設計とCSS

### 4.1. 基本コンポーネント

\* \*\*カード:\*\* タイトル、コンテンツ（テキスト、リスト、テーブル、グラフなど）

### 4.2. レイアウト設計

\* \*\*ベースコンテナ:\*\*

```

.slide-container {

width: 1600px;

min-height: 900px;

height: auto;

background-color: white;

border-radius: 12px;

box-shadow: 0 4px 20px rgba(0, 0, 0, 0.08);

display: flex;

flex-direction: column;

padding: 32px;

position: relative;

overflow: hidden;

}

```

\* \*\*ヘッダー:\*\*

```

.header {

background-color: #2c3e50;

color: #ffffff;

padding: 10px 40px;

display: flex;

justify-content: space-between;

align-items: center;

}

.header-title {

font-size: 28px;

font-weight: bold;

}

.footer {

background: #2c3e50;

color: white;

padding: 10px 40px;

text-align: center;

font-size: 14px;

}

```

\* \*\*カード:\*\*

```

.card {

background-color: white;

border-radius: 10px;

box-shadow: 0 2px 10px rgba(0, 0, 0, 0.05);

height: auto;

overflow: hidden;

border: 1px solid #e2e8f0;

display: flex;

flex-direction: column;

}

.card-title {

background-color: #3498db;

color: #ffffff;

padding: 10px 15px;

font-size: 22px;

font-weight: bold;

margin-bottom: 12px;

display: flex;

align-items: center;

gap: 8px;

border-radius: 8px 8px 0 0;

overflow: hidden;

}

.card-content {

padding: 15px;

flex: 1;

display: flex;

flex-direction: column;

gap: 10px;

font-size: 13px;

overflow-wrap: break-word;

overflow: hidden;

max-height: 100%;

}

```

\* \*\*可変カラム構成 (1, 2, 3カラム対応):\*\*

```

.content {

display: flex;

flex-direction: column;

padding: 20px;

gap: 20px;

}

.column {

width: 100%;

height: 100%;

display: flex;

flex-direction: column;

gap: 16px;

overflow: hidden;

}

.one-column {

width: 100%;

}

.one-column .column {

width: 100%;

}

.two-column {

width: calc(100% + 24px);

display: flex;

flex-wrap: wrap;

}

.two-column .column {

width: calc(50% - 12px);

}

.three-column {

width: calc(100% + 24px);

display: flex;

flex-wrap: wrap;

}

.three-column .column {

width: calc(33.333% - 16px);

margin-right: 8px;

}

```

\* コンテンツの量に応じて、`content` クラスを持つ要素に `one-column`, `two-column`, `three-column` のいずれかのクラスを追加することで、カラム数を制御します。HTML側で、例えば`<div class="content two-column">`のように指定します。

### 4.3. スタイル調整

\* \*\*オーバーフロー対策:\*\*

\* スクロールバーを非表示にするために、`overflow: hidden;` を基本とし、必要に応じてコンテンツを調整すること。

\* どうしてもスクロールが必要な場合は、`overflow-x: auto;` または `overflow-y: auto;` を使用し、最小限にとどめること。文字のはみ出しを防ぐために、`text-overflow: ellipsis;` の使用も検討してください。

\* \*\*リスト要素最適化:\*\*

```

ul {

padding-left: 16px;

margin-bottom: 6px;

}

li {

margin-bottom: 4px;

font-size: 12px;

line-height: 1.3;

}

```

\* \*\*ハイライト要素:\*\*

```

.highlight-pink {

background-color: #fdf2f4;

color: #EE6C8A;

padding: 2px 4px;

border-radius: 4px;

font-weight: 500;

}

.highlight-yellow {

background-color: #fff9c4;

color: #b59d00;

padding: 2px 4px;

border-radius: 4px;

font-weight: 500;

}

```

### 4.4. 拡張コンポーネント

\* \*\*プロセスフロー:\*\*

```

.process-flow {

display: flex;

justify-content: space-between;

margin: 16px 0;

position: relative;

}

.process-step {

text-align: center;

width: 22%;

z-index: 2;

}

.process-icon {

background-color: #EE6C8A;

color: white;

width: 36px;

height: 36px;

border-radius: 50%;

display: flex;

justify-content: center;

align-items: center;

margin: 0 auto 8px;

font-size: 16px;

font-weight: bold;

}

.process-title {

font-size: 13px;

font-weight: 600;

margin-bottom: 4px;

}

.process-desc {

font-size: 11px;

}

```

\* \*\*グリッドレイアウト:\*\*

```

.grid-container{

display: grid;

grid-template-columns: repeat(2, 1fr);

gap: 16px;

}

.grid-item {

background-color: #ffffff;

border-radius: 8px;

padding: 15px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.05);

border-left: 4px solid #3D84B8;

display: flex;

flex-direction: column;

gap: 8px;

}

.grid-item-title {

font-weight: bold;

color: #1E56A0;

font-size: 18px;

margin-bottom: 5px;

}

.grid-item-icon {

font-size: 14px;

color: #EE6C8A;

min-width: 20px;

}

.grid-item-content {

font-size: 14px;

line-height: 1.5;

}

```

\* \*\*アイコンリスト:\*\*

```

.integration-list {

display: grid;

grid-template-columns: repeat(5, 1fr);

gap: 6px;

margin-top: 8px;

}

.integration-item {

background-color: #fdf2f4;

border-radius: 12px;

padding: 3px 6px;

font-size: 11px;

display: flex;

align-items: center;

gap: 2px;

justify-content: center;

white-space: nowrap;

overflow: hidden;

text-overflow: ellipsis;

}

```

\* \*\*テーブル:\*\*

```

.comparison-table {

width: 100%;

border-collapse: collapse;

margin-top: 10px;

font-size: 12px;

border: 2px solid #e2e8f0;

}

.comparison-table th,

.comparison-table td {

padding: 8px;

border: 1px solid #e2e8f0;

}

.comparison-table th {

background-color: #fdf2f4;

text-align: left;

border-bottom: 1px solid #e2e8f0;

}

```

\* \*\*グラフ:\*\*

```

.graph-container {

display: grid;

grid-template-columns: repeat(6, 1fr);

gap: 16px;

width: 100%;

padding: 20px;

}

.graph-box {

display: flex;

flex-direction: column;

align-items: center;

height: 300px;

position: relative;

}

.graph-title {

font-weight: bold;

margin-bottom: 30px;

font-size: 14px;

text-align: center;

position: absolute;

top: 0;

left: 50%;

transform: translateX(-50%);

width: 100%;

}

.graph-value {

font-size: 16px;

font-weight: bold;

margin-top: 5px;

}

.bar {

width: 40px;

position: absolute;

bottom: 40px;

border-radius: 4px 4px 0 0;

transition: height 1s ease-in-out;

}

.bar-2023 {

background-color: #3D84B8;

left: calc(50% - 45px);

}

.bar-2024 {

background-color: #EE6C8A;

left: calc(50% + 5px);

}

.bar-label {

position: absolute;

bottom: 10px;

font-size: 12px;

text-align: center;

width: 40px;

left: calc(50% - 45px);

}

.bar-label-2024{

left: calc(50% + 5px);

}

.bar-value {

position: absolute;

bottom: 100%;

width: 100%;

text-align: center;

font-size: 12px;

font-weight: bold;

margin-bottom: 5px;

}

```

## 5. 実装のベストプラクティス

### 5.1. 情報最適化

\* 長文は要点を絞って圧縮 (1文20-30字程度)

\* 箇条書きは簡潔に (1-2行)

\* 重要概念を選別し、不要な詳細は削除

### 5.2. テキスト調整

\* タイトルは簡潔かつ的確に内容を表す

\* 複合語は短縮

\* 体言止めを積極的に活用

\* 接続詞・副詞は削減

### 5.3. 視覚的階層

\* 情報の重要度に応じた視覚的なヒエラルキーを確立

\* 強調したいポイントには `highlight` クラスを適用

\* 補足情報はより小さなフォントサイズで表示

\* 視認性の高いアイコンを選定

## 6. 実装時の注意点

\* 文字のはみ出しは絶対に避ける。`text-overflow: ellipsis;` の適用を検討。

\* 特にリスト、テーブル、長文パラグラフに注意。

\* コンテンツ量が多い場合は、フォントサイズを12pxまで小さくする。

\* グリッド/テーブルのセル幅を調整し、均等配分より内容に合わせた配分を優先。

\* スクロールバーが必要な場合は、美観を損なわないよう最小限に。

\* カード間のバランスを考慮し、高さの極端な偏りを避ける。

## 7. HTML構造 (ひな形)

```

<!DOCTYPE html>

<html lang="ja">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>インフォグラフィックスライド</title>

<style>

/\* ここにスタイルを定義 \*/

</style>

</head>

<body>

<div class="slide-container">

<div class="header">

<h1 class="header-title">メインタイトル</h1>

<div class="header-date">YYYY年MM月</div>

</div>

<div class="content">

<!-- 左カラム -->

<div class="column">

<div class="card">

<div class="card-title">🔍 セクション1</div>

<div class="card-content">

コンテンツ

</div>

</div>

<!-- 他のカード -->

</div>

<!-- 右カラム -->

<div class="column">

<!-- カード -->

</div>

</div>

<div class="footer">フッター情報 | © YYYY</div>

</div>

</body>

</html>

```