

## (2)「コミュニケーションと情報デザイン」で学ぶこと

- 要するに
  - ◆ 情報をデジタル化してファイルに保存する方法
  - ◆ 情報を圧縮する方法
  - ◆ コミュニケーションの理解
  - ◆ 情報デザインの方法
  - ◆ デザイン思考による問題解決

学習9, 10

# 学習9 情報をデザインすることの意味

- (1) デザインとは何か
- (2) デザインの起源
- (3) モダンデザイン
- (4) デザインの役割
- (5) 情報デザイン
- (6) 情報デザインの歴史
- (7) 情報デザインの活用
- (8) 情報の構造化
- (9) 人間中心設計
- (10) デザインのプロセス

# デザインとは何か

- デザインの定義と役割
  - ◆ 目的を達成するための計画をたてること
  - ◆ デザインの根源的な目的は「人を助けること」
    - 具体的な目的は社会や身の回りの問題の解決



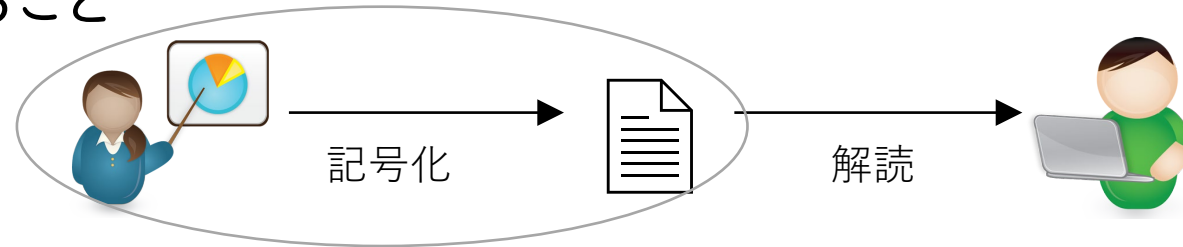
# デザインの起源とモダンデザイン

- デザインの起源：アーツ・アンド・クラフツ運動
  - ◆ 大量生産による安価な製品ではなく、手作業によって芸術性を高めた製品を広め、生活の質を高めようとした
    - 富裕な階層にしか行き渡らず失敗
- モダンデザイン
  - ◆ バウハウス（ドイツの美術・建築学校）
    - 合理的・機能的な工業製品を多くの人に届けようとした
  - ◆ アイソタイプ（オットー・ノイラートら）
    - 視覚言語で文字が読めない社会的弱者への情報伝達を可能にした

# 情報デザインとは何か

- 情報デザインとは

- ◆ 効率的なコミュニケーションや問題解決のために、情報を整理し、意図を明確に伝える手段を選択すること



送り手が記号化した「情報」が、正しく受け手の「知識」となる工夫

- ◆ 提示手段の例

- ピクトグラム (JIS Z8210) <https://www.mlit.go.jp/common/001315215.pdf>
- インフォグラフィックス (図表4)
- グラフ (ナイチンゲールによるクリミア戦争における死因分析グラフは有名)

- ◆ 情報デザインの活用例: 使いやすいユーザインタフェース

# 情報の構造化

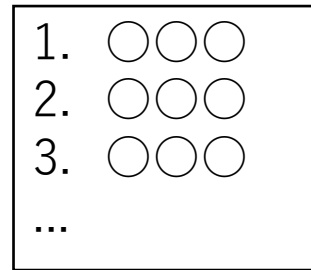
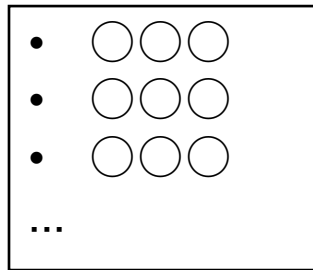
- 構造化すべき情報の具体例
  - ◆ 元素表, メシエ天体のリスト, ことわざ辞典, 年表, 植生図, 生物の進化の系統図, 書店や図書館での配架, ディレクトリ型検索
    - 広島大学 大学生からの情報リテラシー 7章より  
[https://www.riise.hiroshima-u.ac.jp/jkk/jkk2018/Text2018\\_2nd.pdf](https://www.riise.hiroshima-u.ac.jp/jkk/jkk2018/Text2018_2nd.pdf)
- 情報を構造化する手順
  1. 整理する
  2. 結びつきを明らかにする
  3. 提示する

# 整理する

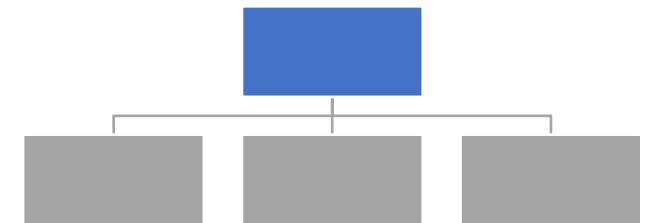
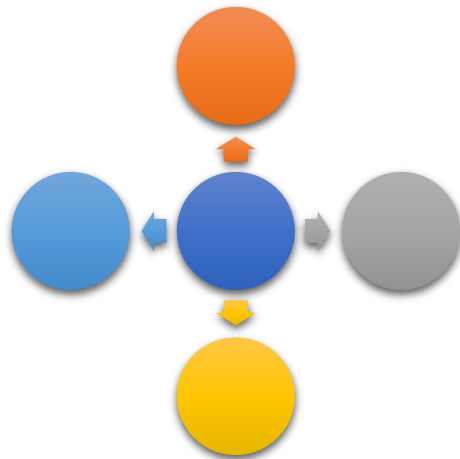
- 情報を整理する基準:「究極の5個の帽子掛け」(LATCH法)
  - ◆ Location 位置による並べ方
  - ◆ Alphabet アルファベット順、五十音順による並べ方
  - ◆ Time 時間軸による並べ方
  - ◆ Category カテゴリー、分類による並べ方
  - ◆ Hierarchy ヒエラルキー、物理量による並べ方

# 結びつける

- 情報の結び付き: 並列、順序、分岐、因果、階層
  - 並列・順序は、それぞれ箇条書き・番号付きリストにより表現できる



- 分岐・因果・階層は構造を明示する図解がわかりやすい



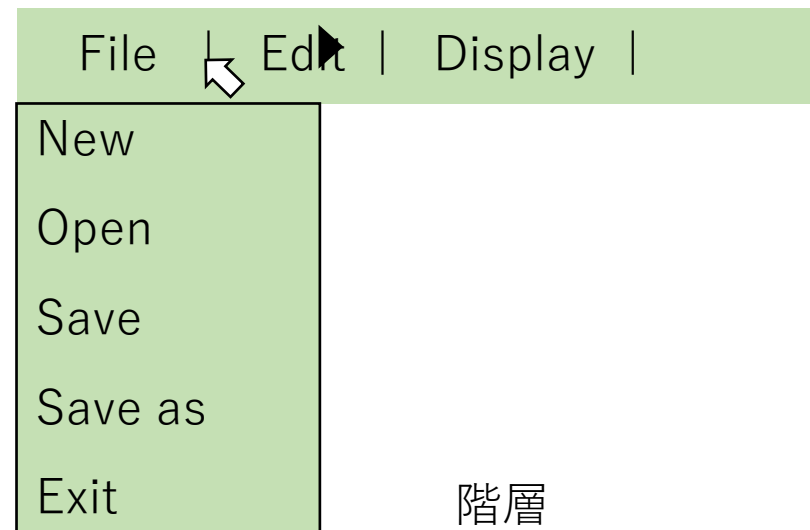


# 提示する

- 送り手が主体となって情報を届ける
  - ◆ 例) スライドやポスターによるプレゼンテーション (補足)
- 受け手が主体となって情報を得る
  - ◆ 例) webサイトの構造デザイン
    - 情報の結び付き (並列、順序、分岐、階層) をサイト構造に反映させる (図表7)



順序



階層

# 人間中心設計とデザインプロセス

- 人間中心設計
  - ◆ 情報デザインとして優れ、利用者の立場に立ったデザインがなされているものの設計プロセス
    - 利用者のニーズ把握、デザイン、評価を繰り返す
  - ◆ 設計プロセスがISOやJISで規定されている(図表8)
- デザインプロセス
  - ◆ 人間中心設計のプロセスに基づき、発散指向でアイデアを積み上げ、収束指向で解決案に絞り込む = デザイン思考(学習10)
    - cf) 問題を分割して部品まで落とし込む=分析思考

# 補足：スライドの作成

- アウトラインの作成
  - ◆ プレゼンテーションの持ち時間からスライドの枚数を決める
    - 1分1枚が目安
  - ◆ 全体の枚数から項目毎の割り当て枚数を決める
    - 背景、関連事例、提案内容、実証報告、まとめ など、項目を決めて、適切なバランスでそれぞれの枚数を決める
  - ◆ それぞれのスライドに入れる内容を箇条書きで書く

# 補足：スライドの作成

- powerpointの使い方

スライドの追加やレイアウト

文字の調整

並びの調整

図形描画

The screenshot displays the Microsoft PowerPoint interface. The title bar shows 'chap02.pptx'. The ribbon includes tabs for 'ホーム' (Home), '挿入' (Insert), 'デザイン' (Design), '画面切り替え' (Slide Show), 'アニメーション' (Animations), 'スライドショー' (Slide Show), '校閲' (Review), '表示' (View), 'ヘルプ' (Help), ' Acrobat', 'IguanaTeX', and '図形の書式' (Format). The 'ホーム' tab is active, showing options for '貼り付け' (Paste), '新しいスライド' (New Slide), 'スライドの再利用' (Reuse Slides), 'クリップボード' (Clipboard), 'スライド' (Slide), 'フォント' (Font), '段落' (Paragraph), '図形描画' (Drawing), '校閲' (Review), '表示' (View), 'ヘルプ' (Help), ' Acrobat', 'IguanaTeX', and '図形の書式' (Format). The main slide area shows slide 46 with the title '結びつける' (Connecting). The content includes a bulleted list: '情報の結び付き: 並列、順序、分岐、因果、階層' (Information connection: parallel, order, branching, cause and effect, hierarchy). Below this, two examples are shown: a parallel list of circles and a numbered list of circles. Further down, a diagram illustrates branching, cause and effect, and hierarchy using circles and arrows. The status bar at the bottom indicates 'スライド 46/67' (Slide 46/67), '日本語' (Japanese), and 'アクセシビリティ: 検討が必要です' (Accessibility: Needs review).

結びつける

- 情報の結び付き: 並列、順序、分岐、因果、階層
  - 並列・順序は、それぞれ箇条書き・番号付きリストにより表現できる
- 分岐・因果・階層は構造を明示する図解がわかりやすい

# 補足：スライドの作成

- 表示 → アウトライン表示
  - ◆ 別途作成したテキストを貼り付けてスライドのページを一括作成できる

The screenshot shows the Microsoft PowerPoint interface. The top ribbon includes tabs for 'ファイル' (File), 'ホーム' (Home), '挿入' (Insert), 'デザイン' (Design), '画面切り替え' (Slide Show), 'アニメーション' (Animations), 'スライドショー' (Slide Show), and '録画' (Record). The 'スライドショー' tab is active, and the 'アウトライン表示' (Outline View) icon is selected. The main area displays a slide titled '49 補足：スライドの作成' (49 Supplement: Creating Slides). The slide content is as follows:

- アウトラインの作成
  - ◆ プレゼンテーションの持ち時間からスライドの枚数を決める
    - 1分1枚が目安
  - ◆ 全体の枚数から項目毎の割り当て枚数を決める
    - 背景、関連事例、提案内容、実証報告、まとめ など、項目を決めて、適切なバランスでそれぞれの枚数を決める
  - ◆ それぞれのスライドに入れる内容を箇条書きで書く

Below this slide, a preview of the next slide is visible, titled '50 補足：スライドの作成' (50 Supplement: Creating Slides), with the content:

- powerpointの使い方

# 補足：伝わるデザインの具体的な方法

- 「伝わるデザイン」 <https://tsutawarudesign.com/>

- ◆ 読みやすく

- 書体の使い分け／フォントの選び方／おすすめフォント／文章／箇条書きと段落／約物と単位

- ◆ 見やすく

- グラフと表／画像と写真／図解と描画／レイアウト／配色

- ◆ バリアフリー

- 配色／書体とフォント／文章

# 補足：伝わるデザインの具体的な方法

- 読みやすく

- ◆ 書体の使い分け

- 日本語

- ✓ 「明朝体」は可読性が高く長文でも疲れにくいので、読ませる文章に使う（游明朝）
      - ✓ 「ゴシック体」は視認性が高く目立つのでプレゼン資料などに使う（メイリオ）

- 欧文

- ✓ 読ませる文章には「セリフ体」（for example, Times New Roman）
      - ✓ スライドには「サンセリフ体」（for example, Segoe UI）

# 補足：伝わるデザインの具体的な方法

- 読みやすく

- ◆ UD(Universal Design)フォント


- 弱視や視覚過敏などの障害を持つ人たちにも読みやすいフォント
- アキを大きくして誤読を少なくしている
- 参考：教育現場で話題！子どもの学習意欲を上げる「UDデジタル教科書体」とは？ <https://diamond.jp/articles/-/236214>



# 補足：伝わるデザインの具体的な方法

- 読みやすく

- ◆ 文章

- 行間：フォントサイズの70%程度
    - 文字間：メイリオなどの字面の大きいフォントや太いフォントでは調整する
      - ✓ パワーポイントでは  ボタンで調整可能
    - 一行文字数
      - ✓ ワードでは二段組みにするなどして行長が長くなりすぎないようにする
    - 左揃え：内容と揃え位置を合わせる
    - 強弱：見出しなどの段階に応じて文字の大きさを変える

# 補足：伝わるデザインの具体的な方法

- 読みやすく
  - ◆ 箇条書きと段落
    - ぶら下げインデントで文の開始位置を揃える
    - 項目毎に空白を開けてまとまりがわかりやすくする
  - ◆ 約物と単位
    - 数字は大きく、単位は小さく（図などの場合）

# 補足：伝わるデザインの具体的な方法

- 見やすく
  - ◆ グラフと表
    - グラフ
      - ✓ Excelのグラフはそのまま使わない
      - ✓ 要素の区別を色だけに頼らない
    - 表
      - ✓ 線を減らし、行間を広げる
  - ◆ 図解と描画
    - 文字だけの説明に頼らず、図解を有効に使う
    - 色・形が同じ図形は同じ概念を表すように

# 補足：伝わるデザインの具体的な方法

- 見やすく

- ◆ レイアウト

- 余白、揃え、まとまり、コントラスト、繰り返しを意識する

- ◆ 配色

- 彩度や明度を低く
    - 使う色は4色までで、テーマに合わせた取り合わせを選ぶ
    - 赤と緑が区別しにくいなどの色覚多様性に注意

# 補足：伝わるデザインの具体的な方法

- バリアフリー

- ◆ 配色

- 赤と緑は見えづらいので、青やオレンジを使う
    - 色に頼らない、色数を増やさない
    - コントラストを強くしすぎない

- ◆ 書体とフォント：UDフォントの使用

- ◆ 文章

- 文章を減らし、重要なところは強調する
    - 行間を空ける
    - 背景に色をつけるときは、文字を細くしすぎない

# 学習10 デザインするための一連の進め方

- (1) デザインの対象を見つける
- (2) デザインのための情報を収集する
- (3) 要件を定義する
- (4) プロトタイプを作成する
- (5) 評価、検証する
- (6) 改善、運用する
- (7) アーカイブする

# デザイン思考

- デザイン思考 (Design Thinking) とは
  - ◆ シリコンバレーにあるデザインコンサルティング会社 IDEO社創業者のデビッド・ケリーが2004年に設立した Stanford大学 d.school によって知られるようになった問題解決の方法論
  - ◆ デザイナーの視点や方法論(ツール)を、解決への手順が明確ではない問題解決に適用するもの
  - ◆ デザインの本質は人を助ける(快適にさせる、喜ばせる)ことで、それは「課題の発見とその解決」にある

# デザイン思考

- デザイン思考の手順

- ◆ いくつかの異なる区切りがある

- 着想、発案、実現
    - デザインリサーチ、シンセシス、ブレスト&コンセプト作り、プロトタイピング&ストーリーテリング
    - 対象の発見、情報収集、要件定義、プロトタイピング、評価、改善、アーカイブ

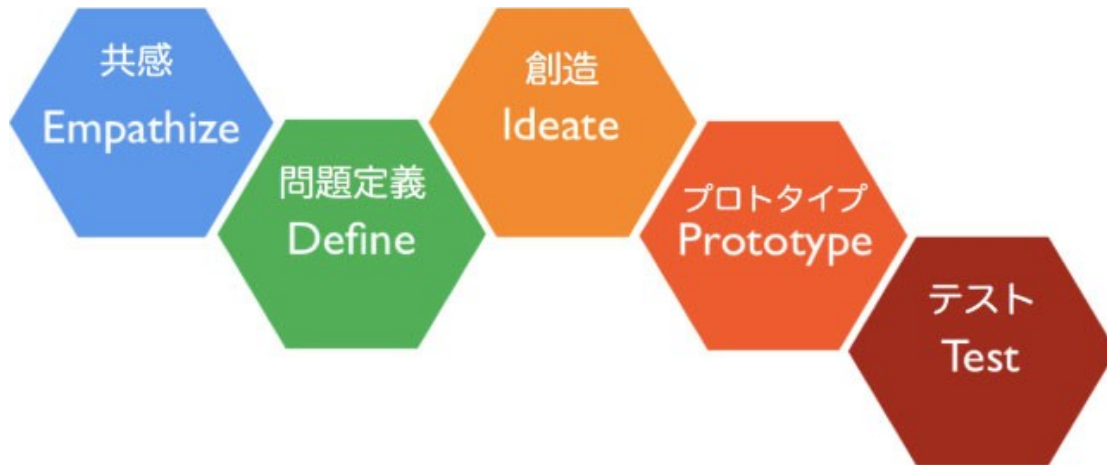
- ◆ 共通する方法論

- ブレーンストーミング、KJ法、観察、インタビュー、プロトタイピング、ユーザビリティテスト



# デザイン思考のプロセス

- d.schoolのプロセス













- 共感
  - ◆ 観察、インタビュー
- 問題定義
  - ◆ KJ法
- 創造
  - ◆ ブレインストーミング、ペルソナ、シナリオ
- プロトタイピング
  - ◆ アジャイル開発
- テスト
  - ◆ ユーザビリティテスト

# 共感

- まず、大きな視点で仮の「問題」を設定し、問題に直面している人に共感し、潜在的ニーズを発見する
  - ◆ 「クライアントからの依頼」からスタートする状況もある
  - ◆ 仮の問題の例：リモートワークを快適にする
- 観察
  - ◆ 虚心坦懐にリモートワーク中の人の動作や言葉を観察する
  - ◆ 他者の目・経験・感情を通じて世界を感じ取る
- インタビュー
  - ◆ 観察された行動に対して、「なぜ」そうしたかを深掘りする
  - ◆ 解決案を聞き出すことではないことに注意

# 共感のツール

- 観察に用いる共感マップ
  - ◆ 左側には何が起きたかを書く（無意味に思える行動も見逃さない）
  - ◆ 右側には観察対象がそのときどう考えたか・感じたかを推測して書く
  - ◆ 項目は付箋で色分け（赤：ポジティブ、黄：中立、青：ネガティブ）
  - ◆ 共感マップ中に意外なもの、矛盾を感じるものがあればニーズの候補

say   	think  
do   	feel  

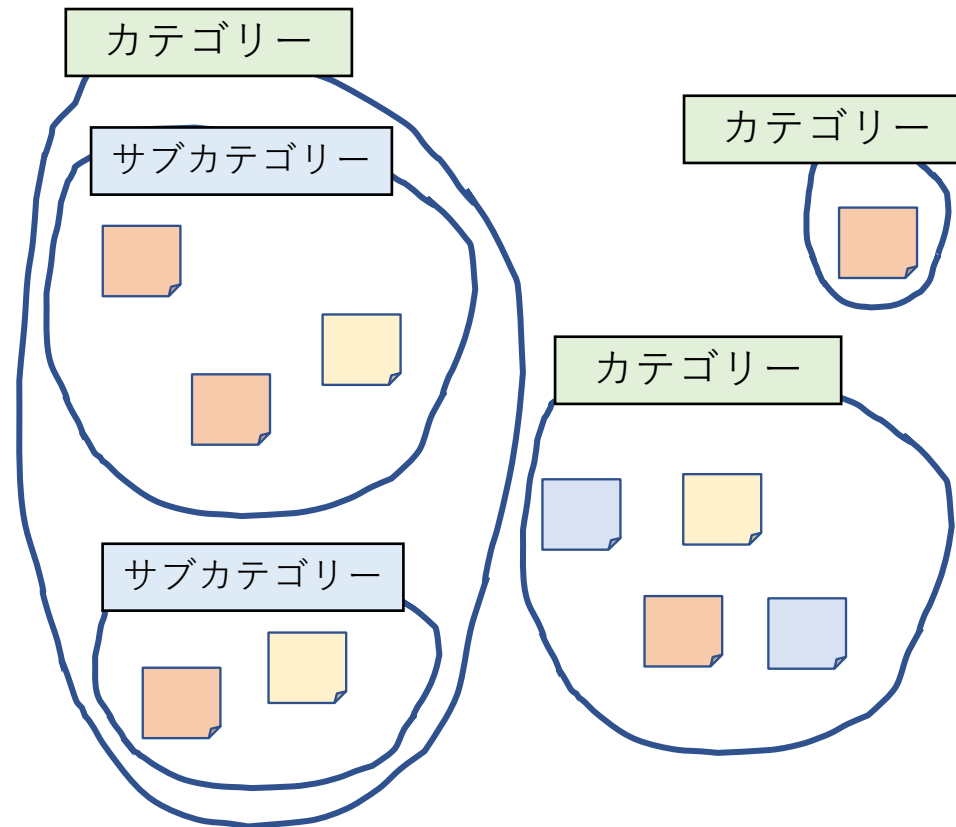
# 問題定義

- 問題定義ステップで行うこと
  - ◆ 共感ステップで集まった情報を整理・分類・解釈して意味づけし、質の高い「問い」を設定する
  - ◆ 「問い」のサイズを意識する
    - 漠然とした大きすぎる問いではないか
    - 具体的すぎて解決手順が明確な問いではないか
  - ◆ 「問い」の例：リモートワークにおいて就業時間とプライベートの時間をすっきり切り替えるにはどうすればよいか

# 問題定義のツール

- KJ法

- ◆ 似ているコンセプトを近くに配置し、カテゴリーを設定する



# 問題定義のツール

- マインドマップ
  - ◆ キーワードを中央に置き、関連するキーワードを放射状に配置する
  - ◆ <https://mindmeister.jp/>
- マンダラート
  - ◆ 3x3のマスの中心にキーワードを置き、関連するキーワードを周囲に8つ配置する
  - ◆ 周囲のキーワードも同様の手順で広げる
  - ◆ マンダラートの例
    - 大谷翔平選手の目標達成シート <https://dime.jp/genre/567605/>

# 創造

- 設定した「問い」に対して解決法を考案するステップ
- 基本的な手順は発散と収束
- 発散
  - ◆ ブレインストーミング: 多様な視点が重要
- 収束
  - ◆ 具体的な問題解決場面を想定してアイデアを検証

# 創造のツール

- ブレインストーミング

- ◆ アイディア出し

- とにかく多くのアイディアを出す
    - この段階では他人のアイディアを批判しない
    - アイディアの「いいところ」だけに着目して、キーワードをリストアップする
    - 他人のアイディアと似ていると思っても遠慮なく出す
    - 他人のアイディアに便乗して、さらにアイディアを付け加えることが推奨される

- ◆ 整理

- KJ法のように、考え方が近いアイディアをまとめて複数の解決案候補を作る

- ◆ 決定

- 解決案候補を批判的に検討し、最終的にひとつの「コンセプト」に絞る





# 創造のツール

- ペルソナ

- ◆ 具体的なユーザを想定

- 広告のターゲティングではないので、年齢・性別・居住地のような属性ではなく、行動パターンや嗜好などを元にユーザ像を考える

- シナリオ

- ◆ 設定したペルソナについて、問題解決の場面を時系列に沿ったシナリオとして書き出す
  - ◆ 問題点が存在するときと、解消されたときとで、どのような違いが生じるかを明確にする
  - ◆ 「コンセプト」の妥当性を検証する

# プロトタイピング

- 最初はコンセプトを検証できる部分に限定して、安く・速く作る
- 「小さく失敗すること」を何度も繰り返して進む
  - ◆ 速ければ失敗がダメージにならない
- 検証すべき3要素（3要素のバランスを取る解を見つける）
  - ◆ 技術的実現性：現在または近い将来に実現可能な技術か
  - ◆ 経済的実現性：コスト的に実現・存続可能か
  - ◆ 有用性：人の欲求を捉えてポジティブな感情をもたらすか

# プロトタイプの種類

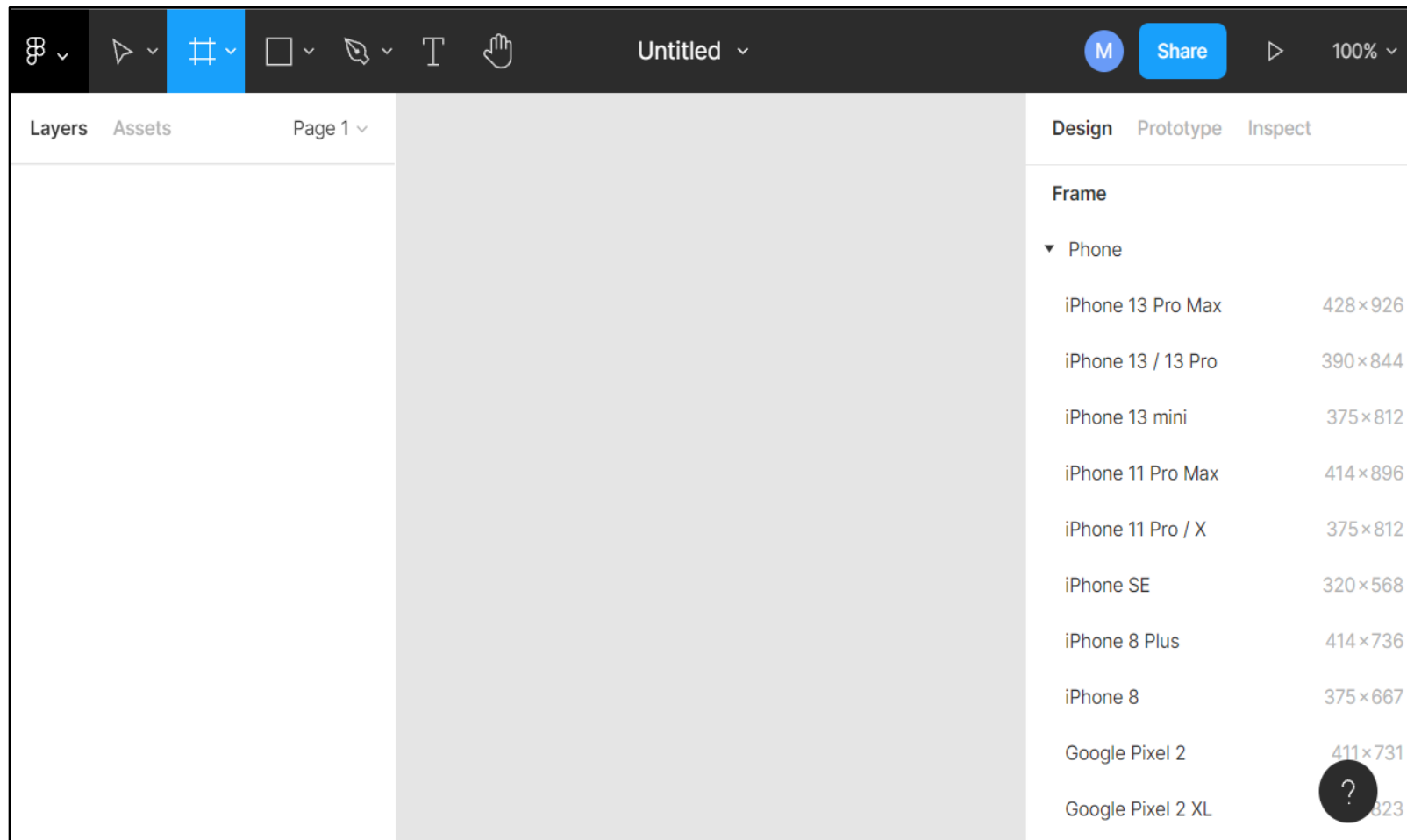
- デザインプロトタイプ
  - ◆ どのように見えるかを確認
  - ◆ 形があるものについてはモックアップを作る
- ファンクショナルプロトタイプ
  - ◆ このような操作をすればこのように動く、ということが実現可能かどうかを検証
- コンテクスチュアルプロトタイプ
  - ◆ 実際に使用しているところを想定したムービーなど

# デザインプロトタイプのツール

- Figma <https://www.figma.com/>
  - ◆ UIデザインやwebサイトのワイヤーフレーム作成のためのツール
    - ブラウザ上で操作できる
    - ファイルを共有し、複数人で作業ができる

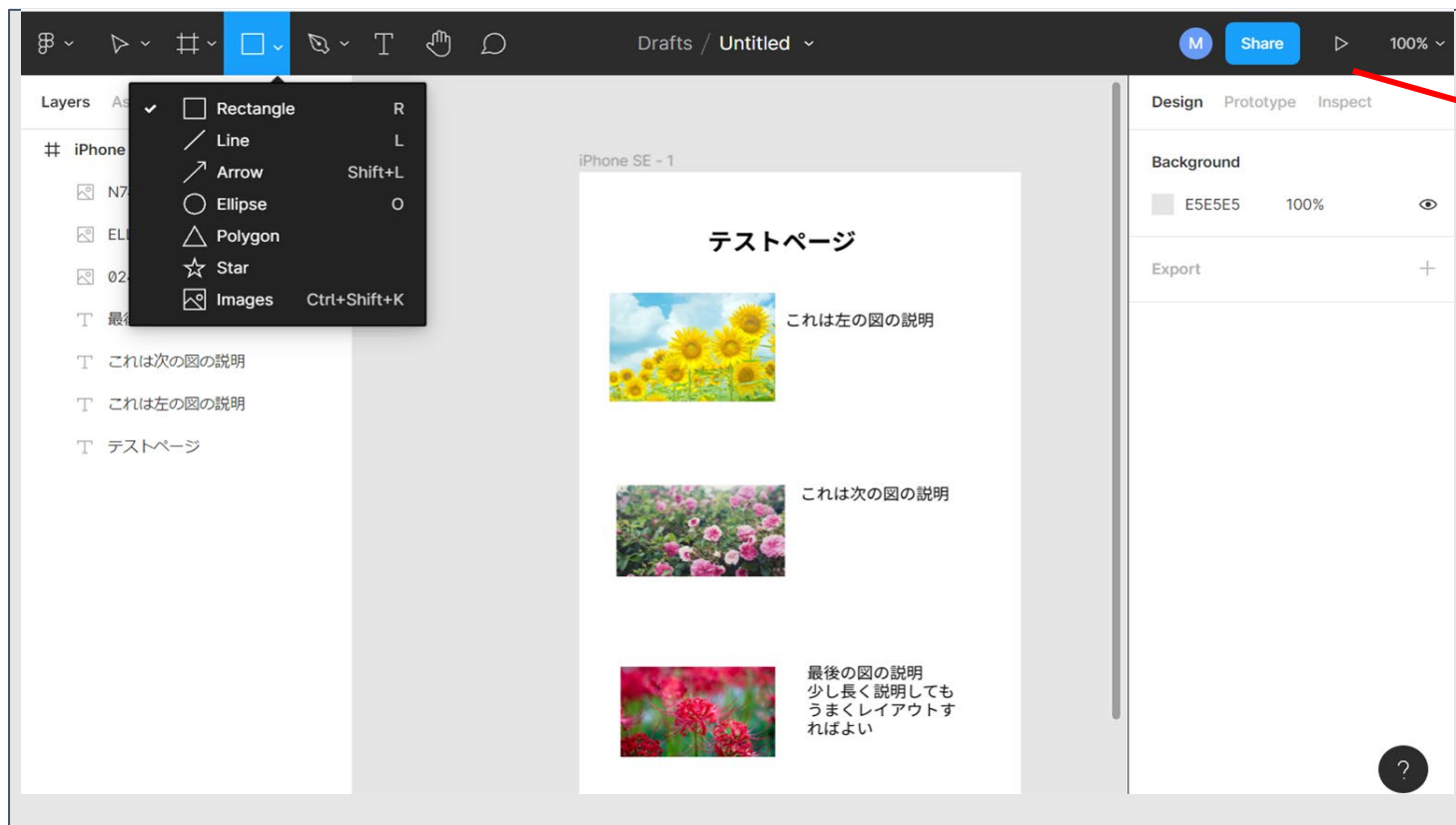
# Figma

- ターゲットのフレームをポピュラーなものから選択できる



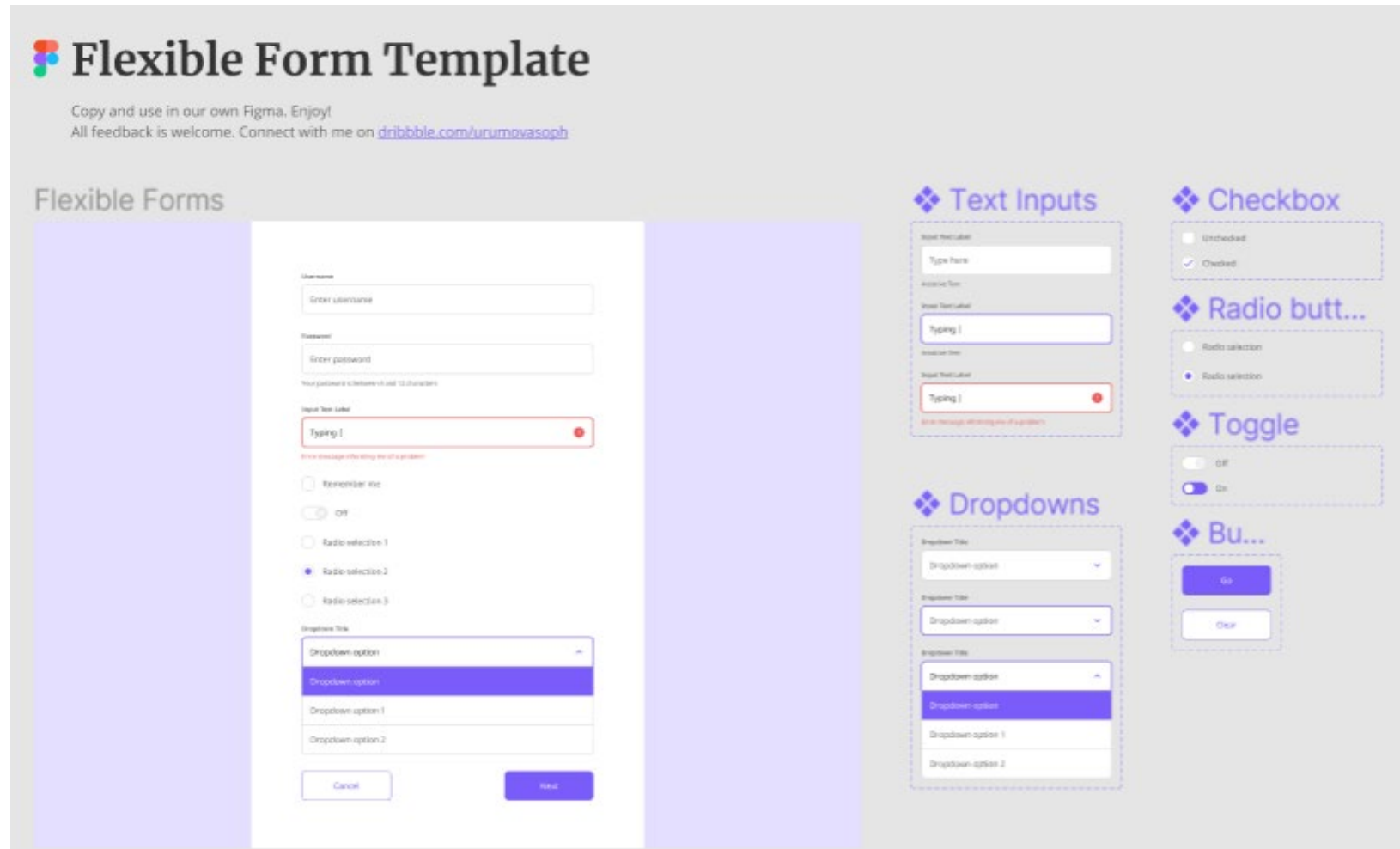
# Figma

- 文字や図形を自由に配置して画面イメージを作成できる



# Figma

- 入力フォームのテンプレートやプラグインもある



# ファンクショナルプロトタイプのツール

- Flask <https://flask.palletsprojects.com/>
  - ◆ Pythonベースのwebアプリ構築ツール
  - ◆ 1つのPythonファイルで複数ページにわたる処理を記述可能
  - ◆ 手元のPCで簡単にサーバを立ち上げて動作確認が可能



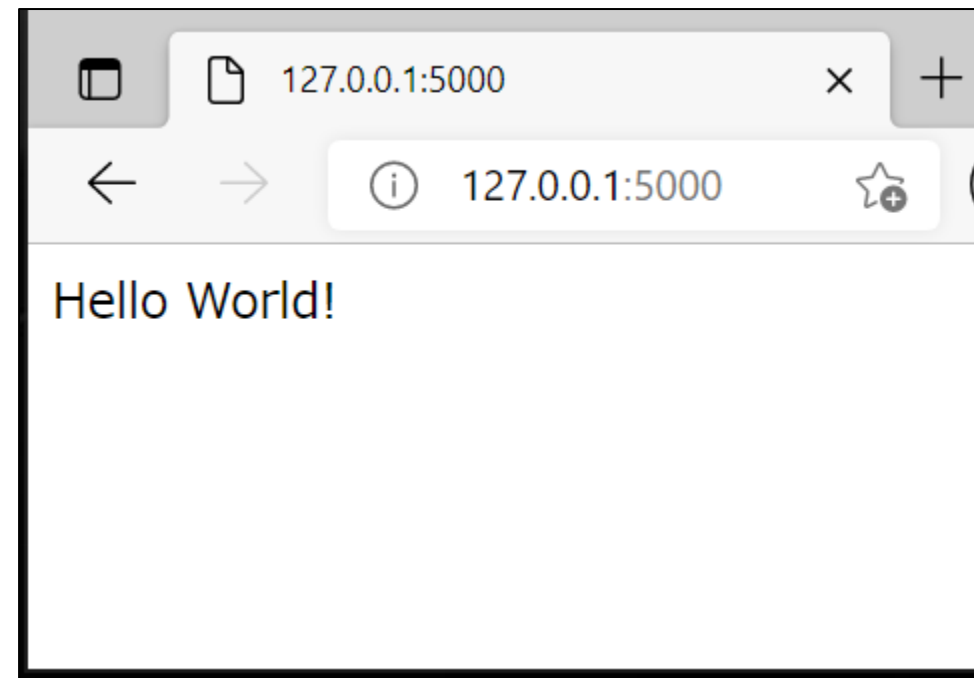
# Flask

- 動作確認

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return "Hello World!"

if __name__ == '__main__':
    app.run()
```



# Flask

- formで値を取得して遷移先のページで処理

```
from flask import *  
app = Flask(__name__)
```

```
@app.route('/')  
def root():
```

```
    return ""
```

```
    <html><body>
```

```
    <form action="/calc" method="post">
```

```
        <input type="text" name="a"> ×
```

```
        <input type="text" name="b">
```

```
        <input type="submit" value="calc">
```

```
    </form>
```

```
    ""
```

```
@app.route("/calc", methods=["post"])
```

```
def calc():
```

```
    a = int(request.form.get("a"))
```

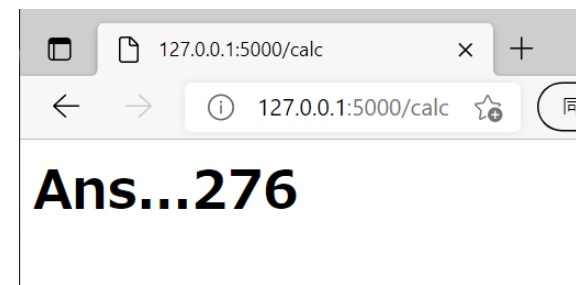
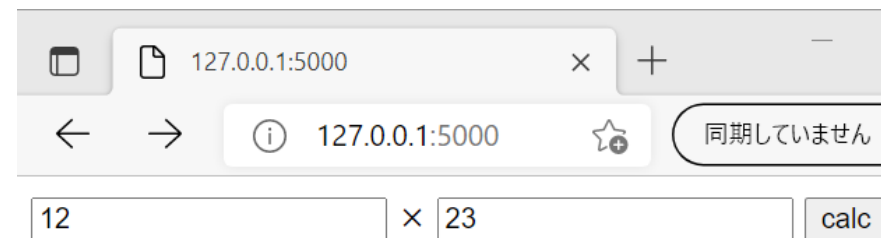
```
    b = int(request.form.get("b"))
```

```
    r = a * b
```

```
    return "<h1>Ans..." + str(r) + "</h1>"
```

```
if __name__ == '__main__':
```

```
    app.run()
```



# テスト

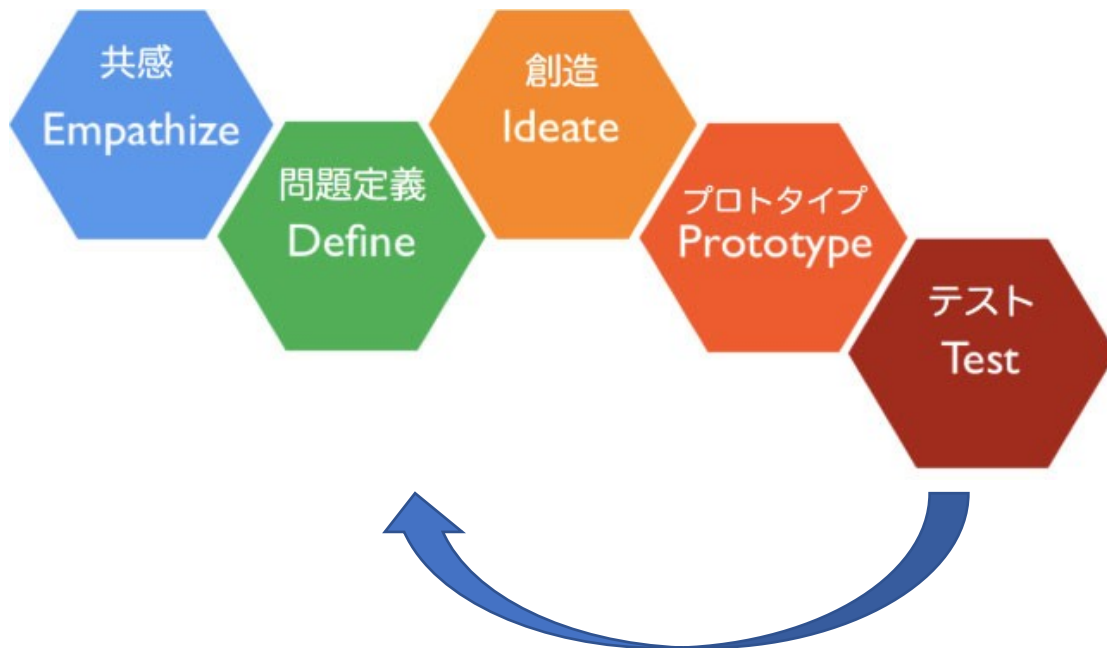
		評価の目的	
		現状把握	改善点の発見
評価主体	開発者	<b>ヒューリスティック評価</b> 事前に専門的知識を反映させたチェックリストを作成する必要がある	<b>認知的ウォークスルー</b> 初めて使うシステムのつもりで、開発者が与えられた目的達成への手順や学習のしやすさを推定
	ユーザ	<b>アンケート評価</b> 定量的な結果が得られるが、信頼性が低く、実施コストが高い	<b>ユーザビリティテスト</b> 新たな視点で問題が発見される可能性があるが、完成品に近いプロトタイプが必要

# ユーザビリティテスト

- なぜユーザビリティテストを行うか
  - ◆ 実装上の問題点を抽出し、プロトタイプを洗練させる
  - ◆ 「5人でユーザビリティテストをすればユーザビリティ上の問題のうち85%が見つかる」といわれている
- ユーザビリティ評価指標
  - ◆ 有効性:どのくらいタスクの目的を達成出来たか
  - ◆ 効率性:どのくらいの資源(主に時間)を費やすか
  - ◆ 満足度:どのくらい不快感がないか

# デザイン思考のプロセス

- d.schoolのプロセス



必要に応じてフィードバック

- 共感
  - ◆ 観察、インタビュー
- 問題定義
  - ◆ KJ法
- 創造
  - ◆ ブレインストーミング、ペルソナ、シナリオ
- プロトタイピング
  - ◆ アジャイル開発
- テスト
  - ◆ ユーザビリティテスト

# 参考文献・資料

- スライド、ポスターなどのデザイン
  - ◆ 高橋 佑磨, 片山 なつ: 伝わるデザインの基本 増補改訂3版, 技術評論社, 2021.
- デザイン思考
  - ◆ 小田、坂元、田村: まんがでわかるデザイン思考, 小学館, 2017.
  - ◆ 石川 俊祐: HELLO, DESIGN 日本人とデザイン, 幻冬舎, 2019.

# 大学入試センター サンプル問題『情報』第1問 問2

- 解答

- ◆ (オ) 1 (集合の関係) 複数手段を利用する人や徒歩の人がいることに注意
- ◆ (カ) 5 (循環)
- ◆ (キ) 2 (2軸による分類)