

# 成果物

## プレ実証講座教材

- AI・プログラミング体験 シラバス
- AI・プログラミング体験 コマシラバス
- AI・プログラミング体験 ワークシート（グループワーク用）
- AI・プログラミング体験 テキスト

本教材は、文部科学省の教育政策推進事業委託費による委託事業として、学校法人穴吹学園 穴吹ビジネス専門学校が実施した令和6年度「地方やデジタル分野における専修学校理系転換等推進事業」の成果物です。

## OAI・プログラミング体験 シラバス

科目名	総授業時間	回数	担当
A I ・ プログラミング体験	180分	全2回	小林 カ
授 業 の 概 要			
<ul style="list-style-type: none"> <li>・ 専門学生・文系学科1年生（医療事務・クラーク学科）を対象としたプレ実証講座。</li> <li>・ 文系学生が理系学科においてAIを活用することについての興味を持ってもらう。</li> <li>・ 理系学科への興味の間口を広げる。</li> <li>・ AIの機能・活用例、AIの4つのレベル、AIの学習の構成要素を通してAIの概要を伝え、実習（Teachable Machine画像認識、Quick, Draw!）において機械学習を体験する。</li> <li>・ 今学習している内容とA Iをどの様に活用するかをリンクさせながら自分たちの生活にどのように関わってくるかを感じさせる。</li> <li>・ A I の出す結果をそのまま、うのみにしない様に指導する。</li> <li>・ プログラミングを体験し、基本的な考え方を知る。</li> </ul>			
授業終了時の到達目標			
<ol style="list-style-type: none"> <li>1. A I の概要を知り身近なものに感じさせる。</li> <li>2. A I の課題を知りファクトチェックの必要性を知る。</li> <li>3. プログラミングを体験し、基本的な考え方を知る。</li> </ol>			
回	テ ー マ	内 容	
1	A I の概要と自分との関わり	①A I の概要・機械学習 ②実習（機械学習体験） ③グループワーク（AIと自分とのかかわりを知る）	
2	プログラミング体験（Python学習）	④表示・入力命令 ⑤繰り返し処理	
教科書・教材		その他	
○オリジナル教材 理解を深めるためのスライド資料 ○GoogleTeachableMachine ○Quick, Draw! ○ChatGPT ○Gamma ○GoogleColaboratory			

## OAI・プログラミング体験 コマシラバス(1/2)

コマシラバス			
科目名	授業時間	回数	担当
AI・プログラミング体験	90分	1回	小林 カ
<p>[今回のテーマ]</p> <p>AIの機能や学習の構成要素などを学習する中で、AIの概要を理解する。 機械学習の実習やグループワークを通して、AIを身近に感じさせる。 AIの課題を理解させる</p> <p>[今回の到達目標]</p> <ul style="list-style-type: none"> <li>・AIがすでに身近な存在であることを理解し、活用を身近に感じられるようにする。</li> </ul>			
<p>[今回の学習内容]</p> <ul style="list-style-type: none"> <li>・AIの機能・活用例</li> <li>・AIの4つのレベル</li> <li>・AIの学習の構成要素</li> <li>・AIが変える私たちの暮らし</li> <li>・医療事務に与える影響</li> <li>・生成AIの使い方</li> <li>・グループワーク</li> </ul> <p>医療事務の仕事に関連して、病院、施設内で必要な掲示物やマニュアルの作成を行う。 グループにて、作成物のテーマ、目的、なぜ必要か、などを話し合う。 生成AIを使用して、アイデア出し、必要な内容を入力して、資料を作成する。</p> <ul style="list-style-type: none"> <li>・AIの課題</li> </ul>		<p>[キーワード]</p> <ul style="list-style-type: none"> <li>・機械学習ディープラーニング</li> <li>・生成AI</li> <li>・プロンプト</li> <li>・誤情報、ハルシネーション</li> <li>・倫理的問題</li> <li>・ブラックボックス</li> </ul>	
<p>[使用教材等]</p> <ul style="list-style-type: none"> <li>・スライド資料</li> <li>・GoogleTeachableMachine</li> <li>・Quick, Draw!</li> <li>・ChatGPT</li> <li>・Gamma</li> </ul>			
<p>[受講者へのアドバイス等]</p> <ul style="list-style-type: none"> <li>・生成AIはすでに身近な存在となっており、簡単に活用できる状態にあります。活用する際には、その限界を理解し、上手に付き合ってください。</li> </ul>			

## OAI・プログラミング体験 コマシラバス(2/2)

コマシラバス			
科目名	授業時間	回数	担当
AI・プログラミング体験	90分	2回	小林 カ
[今回のテーマ]			
Pythonプログラミング体験			
[今回の到達目標]			
<ul style="list-style-type: none"><li>・プログラミングの基本的な考え方を体験する。</li><li>・変数の考え方を理解し、文字と数値の違いを体感する。</li><li>・繰り返し処理を行うことにより、累計計算するための考え方を理解する。</li><li>・簡単なプログラムを作成、実行が出来るという体験に興味を持ってもらう。</li></ul>			
[今回の学習内容]		[キーワード]	
<ul style="list-style-type: none"><li>・表示命令</li><li>・入力命令</li><li>・変数</li><li>・繰り返し命令</li></ul>		<ul style="list-style-type: none"><li>・Print命令</li><li>・Input命令</li><li>・変数</li><li>・文字と数値の違い</li><li>・For命令</li></ul>	
[使用教材等]			
<ul style="list-style-type: none"><li>・スライド資料</li><li>・GoogleColaboratory</li></ul>			
[受講者へのアドバイス等]			
<ul style="list-style-type: none"><li>・エラーが出ると難しく感じるかもしれませんが、慌てず落ち着いて全角・半角などを見直せば解消できることが多いです。シンプルなプログラミングを楽しんでくださいね！</li></ul>			

## 〇AI・プログラミング体験 ワークシート(グループワーク用)

### グループワーク用 ワークシート

#### ■生成AIでプレゼン資料を作ってみよう

- 1 グループ名 :  
グループ員 :  
リーダー :  
書記 :  
発表者 :
- 2 何を作る？
- 3 なぜそれを作る？
- 4 誰を対象にする？
- 5 どこに(で)使おうとしている？
- 6 なぜそれは必要だと思ったか？
- 7 生成AIに何を作ってもらったか？  
何を手伝ってもらったか？
- 8 どんな工夫をしたのか？



## 今日のメニュー

- ▶ AIの機能・活用例
- ▶ AIの4つのレベル
- ▶ AIの学習の構成要素
- ▶ 実習（Teachable Machine画像認識、Quick,Draw !）
- ▶ AIが変える私たちの暮らし
- ▶ 医療事務に与える影響
- ▶ 生成AIの使い方
- ▶ グループワーク
- ▶ AIの課題
- ▶ プログラミング

## 目標





- 1 AIの特徴を知り、機械学習の体験
- 2 生成AIの利用に慣れる
- 3 注意すべき点を理解する
- 4 課題発見する
- 5 協調して課題解決する  
AIを活用してみる
- 6 AIと自分のかかわりを知る
- 7 プログラミング体験  
考え方、手順を学ぶ

## 人工知能の機能・活用例

画像認識	音声認識
画像に写っている 人・物の識別 	人間の音声の テキスト化 処理 
自然言語認識	予測・異常検知
人間が入力する テキストの 変換・処理 	過去のデータに基づいた 将来的な予測、外れ値・ 変化点などの検出による 異常値検知 

3

## 「人工知能」と言われるものの4つのレベル

レベル1	<b>単純制御：指示されたことをそのまま行う</b> 
レベル2	<b>ルールベース：指示されたことを自ら考えて実行する</b> 
レベル3	<b>機械学習：着眼点は人間が教え、対応パターンを自動的に学習する</b> 
レベル4	<b>深層学習：着眼点を人間が教えずに、対応パターンを自動的に学習する</b> 

4

## AI利用の例 スマートスピーカー

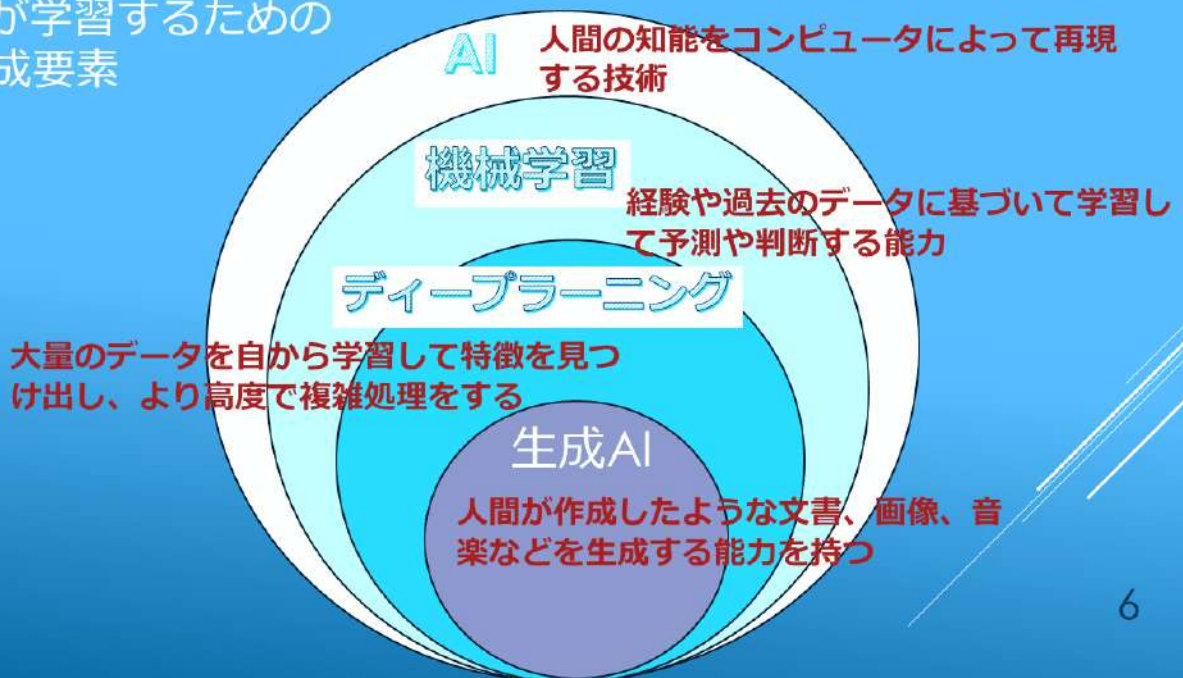


### 音声対話によるサービスや機器の操作

- ✓ 「苦手」や「難しい」というユーザーの取り込み
- ✓ ユーザー増大によるサービス収益源の拡大
- ✓ 「標準」の地位を確保しユーザーの囲い込み

5

## AIが学習するための構成要素



6



## Google Teachable Machineとは

### 実習のねらい

- AIの仕組みを理解する
- 機械学習を身近に感じる
- AI技術への興味関心を高める

### Teachable Machineへのアクセス

プロジェクトの作成

クラスの作成

データの収集

学習

プレビューとエクスポート

### WEAR (ウェア)

<https://wear.jp/>

WEARは、ファッションコーディネート投稿アプリです。このアプリでは、ユーザーが投稿したコーディネート画像をAIが解析し、類似商品を検索して表示する機能があります。

この機能に、Google Teachable Machineのような画像認識技術が活用されていると考えられます。

## ニューラルネットワークは絵を認識できるようになるのでしょうか？ QUICK, DRAW！ を体験

- ▶ Classroomのリンクをクリックしてください
- ▶ google検索で「[quick, draw!](https://quickdraw.withgoogle.com/)」



Quick, Draw!

<https://quickdraw.withgoogle.com/> · このページを訳す

### Quick, Draw!

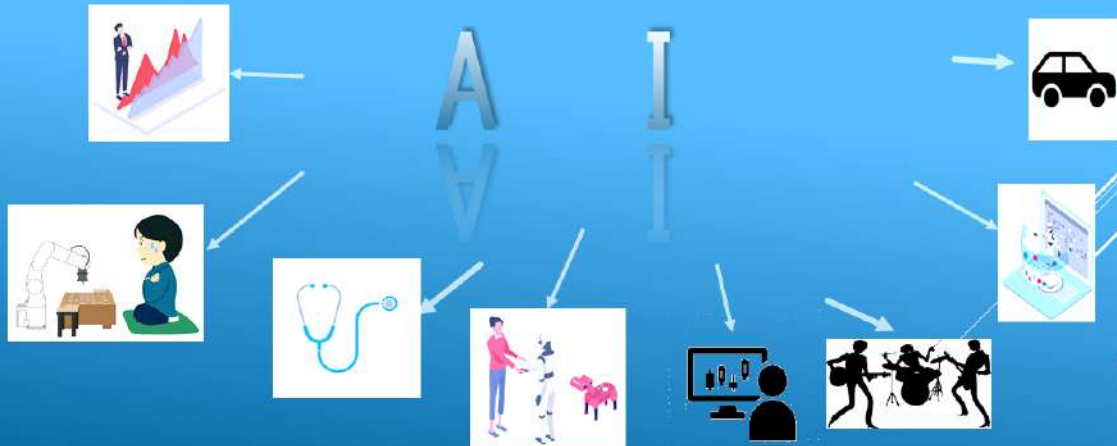
Help teach it by adding your drawings to the world's largest doodle machine learning research. Let's ...

### ニューラルネットワークは絵を ...

ニューラルネットワークは絵を認識できるようになるでしょう

# AIが変える私たちの暮らし

「AIは私たちの日常を支えています。」



## AIは医療事務にどのような影響を与えるか

業務効率化

事務処理の自動化

業務品質向上

患者満足度向上

▶生成AI搭載「Gamma」によるプレゼンテーション資料作成例

▶高齢者にもわかりやすい受付・院内案内業務

▶ <https://gamma.app/docs/-t841lwihbayn5f?mode=doc>

GW

高齢者にもわかりやすい受付・院内案内業務



11

## 生成AIの使い方（プロンプトの書き方）

> 生成AIと脳のコラボ

- ▶ 1 役割（立場）を与える
- ▶ 2 目的を伝える
- ▶ 3 どんな内容にしたいかを具体的に
- ▶ 4 誰が読むかを教える
- ▶ 5 文章の形式や長さを指定する
- ▶ 6 雰囲気やトーンを伝える

12

# 生成AIの使い方の練習

1 生成AI Chat-GPT  
<https://chatgpt.com/>

## ChatGPTで主にできることは

- 文章作成 - エッセイ、小説、記事、キャッチコピーなどを作成
- 要約 - 長文の要点を短くまとめる
- 翻訳 - 日本語⇄英語を含む多言語翻訳
- プログラミング - コードの作成・修正・デバッグのサポート
- アイデア出し - ビジネス・創作・企画のブレインストーミング
- 学習支援 - 難しい概念の説明や問題の解説
- スケジュール管理 - タスクの整理や計画のアドバイス
- 雑学・豆知識提供 - 幅広い知識の提供
- 対話・相談 - 雑談や悩み相談の相手
- 最新情報の検索 - Web検索で最新ニュースや情報を取得  
などなど

13

OpenAI 研究 製品 安全性 企業情報

ChatGPT 無料版とPlus Team

ChatGPT

# 答えを手にも。ひらめきを感じて。生産性を新たな次元へ。

無料でかんたんに使えるChatGPTに質問するだけで、文章作成や学び、アイデア出しをお手伝いします。

クリック

今すぐ始める

アプリをダウンロードする

14



## 2 生成AI gemini

<https://gemini.google.com/app?hl=ja>

Geminiは、以下の10個の機能を備えています。

1. **多言語対応:** 様々な言語でコミュニケーションできます。
2. **コード生成:** Python、Java、C++など様々な言語のコードを生成できます。
3. **翻訳:** 英語、日本語、フランス語など様々な言語の翻訳ができます。
4. **要約:** 長文を要約して短くわかりやすく説明できます。
5. **質疑応答:** 質問に対して適切な回答を生成できます。
6. **文章作成:** メール、記事、物語など様々な文章を作成できます。
7. **テキスト分類:** テキストをテーマやカテゴリ別に分類できます。
8. **感情分析:** 文章に込められた感情を分析できます。
9. **情報検索:** ウェブから必要な情報を検索できます。
10. **対話:** 人間と自然な対話をすることができます。

Gemini

# Gemini

創造力や生産性を高めましょう

Google AI とのチャットで、文章やリストを作成したり、計画を立てたり、新しいことを学んだりできます

ログイン

ログイン、クリック

17



## アカウントの選択

カ

ログアウトしました

カ

アカウントを使用

アカウントを削除

学校のメールアドレス入力

日本語

ヘルプ

プライバシー

規約

18



パスワードを入力

パスワードを表示する

学校のメールアドレスの  
パスワードを入れてください

19

## 生成AIに手伝ってもらおう（アイデアもらおう）

- ▶ 1. 患者さんにやさしい受付サービス：「待ち時間を快適にする方法を考えよう」
- ▶ 2. 医療事務の効率化
- ▶ 3. 医療情報を分かりやすく伝える工夫「患者さんに伝わりやすい説明ツールを考えよう」
- ▶ 4. 未来の医療事務の役割を考える
- ▶ 5. 医療事務のやりがいを考える
- ▶ 6. 困った患者さん対応を想定したトレーニング
- ▶ 7. 健康や医療に関心を持ってもらう工夫  
などなど

20

## グループワーク

- ▶ 1 4名～3名のグループを作ってください
- ▶ 2 リーダー。発表者。書記を決めてください
- ▶ 3 病院や施設内で必要なこと（院内掲示や受付マニュアルなど）  
自分たちは何を作るか決めてください
- ▶ 4 生成AIに手伝ってもらって、作成したものを発表してください

### 発表内容)

- 何を作ったか？誰を対象として、どこにどのように使うのか？
- なぜそれは必要だと思ったか？
- AIに何をどのようにお願いし、どのような工夫をしたか？

21

## AIの課題

- ▶ 誤情報やハルシネーション  
うのみにせず、確認する
- ▶ 個人情報や会社・組織の内部情報を入力しない
- ▶ 著作権、倫理問題を理解する
- ▶ AIの限界を知って、依存しすぎない
- ▶ ブラックボックス問題



22



1. プログラムとは？
2. プログラムを書かなくてもAIは使える
3. プログラムの考え方を体験してみよう
4. なぜPythonを学ぶのか？

#### 今日の実習の目的

- ◆◆ プログラムの基本的な考え方を体験する
- ◆◆ Pythonを使って、シンプルなプログラムを書いてみる
- ◆◆ プログラムを知らなくてもAIは使えるが、知っておくとAIをより活用できる！

23

## Pythonでプログラムを作ってみよう Colabの使い方

- ▶ Googleアカウントでログインしてください。
- ▶ Classroomのリンクから入ってください

Gmail 画像 ログイン

Gmail 画像 カ

<https://colab.research.google.com>



24

Colaboratory へようこそ  
ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

目次

- はじめに
- データサイエンス
- 機械学習
- その他のリソース
- 使用例

+ セクション

+ コード + テキスト | ドライブにコピー

## Colab へようこそ

(新規) Gemini API をお試しください

- [Generate a Gemini API key](#)
- [Talk to Gemini with the Speech-to-Text API](#)
- [Gemini API: Quickstart with Python](#)
- [Gemini API code sample](#)
- [Compare Gemini with ChatGPT](#)
- [More notebooks](#)

すでに Colab をよくご存じの場合は、この動画でインタラクティブなテーブル、実行されたコードの  
ご覧ください。



25

## ノートブックの作成

ノートブックを開く

Colab へようこそ

Untitled2.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

+ コード + テキスト

▶ | ローディングを開始するか、AI で生成します。

+ ノートブックを新規作成

26

Untitled6.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール

- ドライブで探す
- Playground モードで開く
- ドライブの新しいノートブック
- ノートブックを開く Ctrl+O
- ノートブックをアップロード
- 名前の変更
- 移動
- ゴミ箱に移動
- ドライブにコピーを保存
- コピーを GitHub Gist として保存
- GitHub にコピーを保存
- 保存 Ctrl+S
- 版を保存して固定 Ctrl+M S
- 変更履歴
- ダウンロード
- 印刷 Ctrl+P

ノートブックの名前

ノートブックの新規作成

履歴からノートブックを探す

ノートブック名 変更

ノートブックの保存。フォルダーは指定できない

27

# プログラムを書いてみよう 表示命令 : print

## print文 表示させてみよう

実行結果

🔄 こんにちは、小林 さん  
頑張って、小林 さん

Print( "こんにちは、〇〇 〇〇さん" )

Print( "頑張って、〇〇 さん" )

28

## 四則演算をPythonでやってみよう！ 演算子

	意味	例	優先順位
-、+	負符号、正負号	-10	2
+	加算	1+2	4
-	減算	20-10	4
*	乗算	5*3	3
/	除算	3/2 → 1.5	3
//	整数除算	3//2 → 1	3
%	余り	10%3 → 1	3
**	べき乗	2**3 → 8	1

10 + 20  
10 - 20  
10 × 20  
10 ÷ 20 を

それぞれprint文  
を使って表示しよう



```
print(10 + 20)  
print(10 - 20)  
print(10 * 20)  
print(10 / 20)
```



```
30  
-10  
200  
0.5
```

## 楽しよう！ 変数を使おう

```
print ( a + b )  
print ( a - b )  
print ( a * b )  
print ( a / b )
```

```
▶ a=10  
b=20  
print ( a + b )  
print ( a - b )  
print ( a * b )  
print ( a / b )
```

31

## # 簡単な自己紹介プログラム **入力命令 input**

```
name = input ("あなたの名前は何ですか? :")
```

```
Print ("こんにちは、", name, "さん!")
```

```
Print ("プログラミングの世界へようこそ!")
```

```
▶ name = input ("あなたの名前は何ですか? :")  
print ("こんにちは、", name, "さん!")  
print ("プログラミングの世界へようこそ!")
```

```
⇒ あなたの名前は何ですか?: 小林  
   こんにちは、 小林 さん!  
   プログラミングの世界へようこそ!
```

32

## # 簡単な自己紹介プログラム2

```
名前 = input("あなたの名前は何ですか？")
歳 = input("あなたは何歳ですか？")
print("こんにちは、" + 名前 + "さん！")
```

キーボードで  
入力

変数

```
print(歳 + "なんですね。プログラミングを楽しもう！")
```

```
あなたの名前は何ですか？： 小林
あなたの年齢は何歳ですか？： 18
こんにちは、 小林 さん！
18 歳なんですね。プログラミングを楽しもう！
```

33

例題) キーボードから ドル→円 換算できるプログラムを作ろう ただし1ドル=155円とする  
入力した値を計算する

- 1 何ドルかを入力して変数 **ドル** に**入力**
- 2 変数 **円** に  $\text{ドル} \times 155$  を計算して**代入**
- 3 `print`命令 で「ドル換算は？」と **円** **表示**

34



繰り返し命令 ( `for~in~` 処理)

繰り返し、回数が決まっている場合、3回繰り返します

```
!pip install emoji
```

```
import emoji
```

```
# 繰り返したい絵文字をリストに格納
```

```
動物絵文字 = [ ":dog:", ": dog :", ": dog" ]
```

```
# 繰り返し処理
```

```
for 絵文字 in 動物絵文字:  
    print ( emoji.emojize( 絵文字 ) )  
    print ()
```

実行結果



37

「1～5までの数字の合計（累計）を出すプログラムを考えてみよう

```
数字 = [1, 2, 3, 4, 5]
```

```
合計 = 0
```

```
for 数 in 数字:
```

```
    print("繰り返した回数", 数)
```

```
    print("今の合計 . . .", 合計)
```

```
    合計 = 合計 + 数
```

```
    print("計算後の合計 . . .", 合計)
```

```
    print()
```

```
print( 合計 )
```

38



5回を指定して繰り返しをしてみよう

```
合計 = 0
```

```
for 数 in range(1,6,1):
```

```
    print("繰り返した回数",数)
```

```
    print("今の合計 . . .",合計)
```

```
    合計 = 合計 + 数
```

```
    print("計算後の合計 . . .",合計)
```

```
    print()
```

```
print( 合計 )
```

「1から5」までの数字を  
順番に作り出す

range(1,6,1)となる

range(5)は「0から4」までの  
数字を順番に作り出します。

39

例題) 1～50までの数字の合計（累計）を出すプログラム  
を考えてみよう

練習) 1 ~ 50までのうち、奇数の合計を出すプログラムを考えてみよう

41

練習) 1 ~ 50までのうち、奇数の合計と平均を出すプログラムを考えてみよう

42

応用) 1 ~ 5 1 までのうち、偶数の合計を出すプログラムを考えてみよう

43

応用) 1 ~ 5 1 までのうち、偶数の合計と平均を出すプログラムを考えてみよう

44

## 今日の運勢を占ってみよう

```
[1] def 占い():  
    import random  
    choices = ["大吉", "中吉", "小吉", "末吉", "凶"]  
    print("今日の運勢は...", random.choice(choices))
```

```
[2] 占い()
```

45

```
def 占い():  
    import random  
    choice = ["大吉", "中吉", "小吉", "末吉", "凶"]  
    print("今日の運勢は...", random.choice(choice))  
    最大 = 100 # 任意の大きな数を設定  
    for 人数 in range(1, 最大 + 1):  
        print()  
        print(人数, "人目の占い結果:")  
        占い()  
        print()  
        next = input("次の方の占いをしますか? (y/n) :")  
        if next == "n":  
            print()  
            print("おしまい")  
            break  
    print("占い終了です!")
```

46