

# このファイルは授業用スライドです

授業で改編して使用する場合は [Googleスライド形式はこちら](#) に用意してあります。

共有権限をリクエストせず、ファイル → ダウンロード からファイル形式を選んで、ダウンロードしてください。



# 双方向性のあるコンテンツの プログラミング①

# 今日のめあて

- 双方向性のあるコンテンツのしくみを理解する
- 様々なメディアの特徴を理解する

## 双方向性のあるコミュニケーション

私たちは日常、友達や先生たちとコミュニケーションを取るとき、お互いに会話をします。



# 双方向性のあるコミュニケーション

話しかけた内容に対して、返事をしたり、答えを返しますよね？



あしたの  
天気は？



晴れだよ

# 双方向性のあるコミュニケーションとは

**双方向性**とは、利用者の働きかけ(入力)に応じて、応答(出力)する。そのやり取りが一方方向ではなく、相互にやり取りする様子です。



あしたの  
天気は？

入力



晴れたよ

出力

# 双方向性のあるコンテンツとは何か

例) スマートスピーカー (音声コンテンツ)



あしたの  
天気は？

電話かけて



スマートスピーカー

# 双方向性のあるコンテンツとは

例) 地図検索 (画像コンテンツ)

例) オンラインショッピング (画像、文字、動画コンテンツ)



これらのコンテンツは  
みなインターネットを利用している

# もしもインターネットがなかったら...

- インターネットを利用しないこれらのコンテンツとの差はなんだろうか？



# ネットワークを利用することの利点

- 情報が新しい
- 知りたい情報を詳しく知れる
- 知りたい情報を素早く知れる
- 文字・音声・画像など複数の情報を得られる



# 様々なメディアとその特徴

	文字	音声	静止画	動画
プラス面	<ul style="list-style-type: none"> <li>●情報が正確に伝わりやすい。</li> <li>●大きさや形を変えることで、伝えたい部分を強調できる。</li> <li>●容易に作成し、修正できる</li> <li>●データ量が<b>少ない</b></li> </ul>	<ul style="list-style-type: none"> <li>●音楽で作品の内容を伝えることができる</li> <li>●音声を流すことで、人がいなくても説明できる</li> <li>●視覚に障害がある人にも内容を伝えることができる</li> </ul>	<ul style="list-style-type: none"> <li>●文字で説明しにくい状況を、わかりやすく伝えられる</li> <li>●デジタルカメラなどで撮影すれば、簡単に作成できる</li> <li>●多くの情報を見るままに伝えることができる</li> </ul>	<ul style="list-style-type: none"> <li>●伝えたい情報に動きを持たせ、変化を強調して表現できる</li> <li>●状況の変化を、時間を追って伝えることができる</li> <li>●編集でさまざまな効果を付けられる</li> </ul>
マイナス面	<ul style="list-style-type: none"> <li>●多くの情報を伝えにくい</li> <li>●視覚に障害のある人や、まだ文章が読めない子どもには、内容が伝わらない。</li> </ul>	<ul style="list-style-type: none"> <li>●順番に時間をかけて聞かなくてはならない。また、データを修正しにくい。</li> <li>●データ量が<b>やや多い</b></li> <li>●聴覚に障害のある人には、内容が伝わらない。</li> </ul>	<ul style="list-style-type: none"> <li>●データを修正しにくい</li> <li>●データ量が<b>やや多い</b></li> <li>●前後の動きを示しにくい</li> <li>●視覚に障害がある人には内容が伝わらない</li> </ul>	<ul style="list-style-type: none"> <li>●データを修正しにくい</li> <li>●データ量が<b>多い</b></li> <li>●順番に時間をかけて見なくてはならない</li> <li>●視覚に障害がある人には内容が伝わらない</li> </ul>

# いろいろなテーマに適したメディア

- 次のテーマについてどのメディアで伝えるのが良いか、グループで話し合ってみよう
  1. 明日の天気
  2. 魚のさばき方
  3. 富士山
  4. 私たちの学校
  5. 選挙の結果

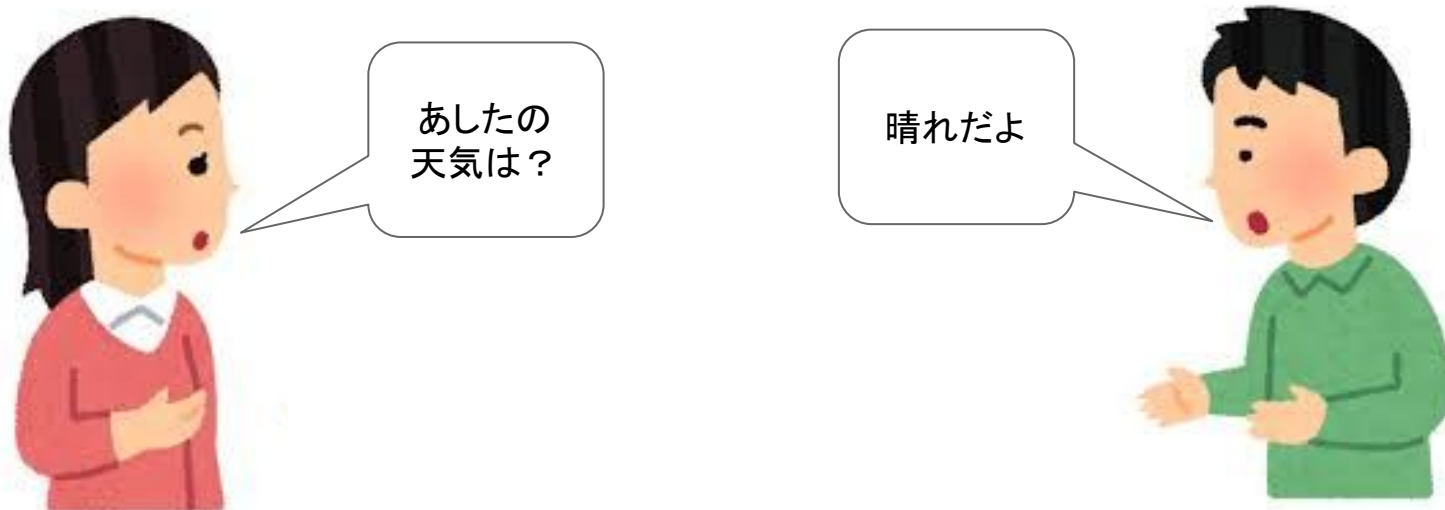
## 今日のまとめ

- 双方向性のあるコンテンツのしくみを理解する
- 様々なメディアの特徴を理解する

# 双方向性のあるコンテンツのプログラミング②

# 双方向性のあるコミュニケーション

**双方向性**とは、利用者の働きかけ(入力)に応じて、応答(出力)する。そのやり取りが一方向ではなく、相互にやり取りする様子です。



# 双方向性のあるコンテンツ

(例) 地図検索

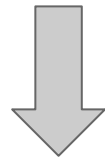


## 今日のめあて

- 問題点を見つけ出し、課題を設定してみよう
- ID/パスワードの意味を理解し適切に管理しよう

# 学校内で友達と情報のやり取りをするときの問題点

- コロナ禍のために人が一ヶ所に集まらない
- メールアドレスや電話番号を知らない人がいる



チャットツール(伝言板)で情報を共有

# チャットツール(伝言板)を作ってみよう

- クラスのみんなが自由に書き込めるチャット

Social Networking Service → SNS

(例) LINE、Facebook、Twitter、Instagram...

文字・写真・動画などそれぞれに得意なメディアがある

# チャットツールではどのような情報を やり取りしたいか

- 文字・文章
- 写真・イラスト・地図・スタンプ
- 動画 など

# ブラウザを起動

- 「プログル技術」と検索

middle.proguru.jp ▼

## プログル技術 | 中学校・技術で使える「双方向性のある ...

プログル技術は「双方向性のあるコンテンツのプログラミング」に特化した、中学校・技術の授業ですぐに使えるプログラミング教材です。チャットアプリの制作を通して、情報科社会に生きる上で必要となる考え方を自然と身につけることが ...

### ログイン

ログインページ. クラスID. 出席番号. パスワード. パスワードの設定が ...

### お試しレッスンを始める

プログラムを組んだら「実行する」ボタンを押してね. ステージ0. 画像の送 ...

[proguru.jp からの検索結果 »](#)

# プログル技術を使う準備をしよう



プログル技術をはじめる

先生	生徒	お試しレッスン
<p>新規登録 &gt;</p> <p>ログイン &gt;</p>	<p>新規登録 &gt;</p> <p>ログイン &gt;</p>	<p>お試しレッスン &gt;</p>

# 利用登録画面



クラスID

クラスID

出席番号

自分の  
出席番号

初期パスワード

先生から  
配られたもの

アカウントを作成する

# パスワードの設定

## パスワードの再設定

- 半角の英字 (a~z, A~Z) や数字 (0~9), 記号 (!, ?, & など) が使えるよ  
半角の英字・数字・記号の中から2種類以上を組み合わせ  
8文字以上の強いパスワードを設定しよう

パスワード

とても弱い

パスワード (確認用)

パスワードを更新する

①まず、強いパスワードを考えてみよう。

②考えたパスワードをメモしよう。

③間違えないように2ヶ所に入力しよう

④更新する

# ブラウザにパスワードの保存を聞かれたら...



パスワードを更新しますか？

ユーザー名

パスワード

パスワードは Google アカウントに保存されるため、  
どのデバイスからでも使用できます

スキップ  
してください  
(保存しない)

# ニックネームとアイコンの決定

## プロフィールの編集

ニックネーム

①ニックネームを  
入力

②好きなアイコン  
を選んでください

※あとで変更もできます

③更新する

更新する

# ログイン画面



クラスID

出席番号

パスワード

**① クラスID**

**② 出席番号**

**③ 自分で作ったパスワード**

ログインする

# この「ホーム画面」になったら準備完了



あなたの作成したチャット  
チャットはレッスン5で完成します

[チャットを確認する](#)



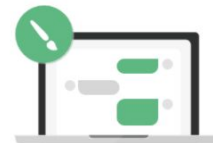
# 1

Lesson 1

## 伝言板をデザインしよう！

Design a message board!

クラス内で使う伝言板のデザインに関して学んでいくよ！使う人にとって見やすいデザインを考えながら伝言板の画面を作成しよう！



## 今日のまとめ

- 問題点を見つけ出し、課題を設定してみよう
- ID/パスワードの意味を理解し適切に管理しよう  
→次回からプログラム技術を起動できるように  
しっかりと管理しよう

# 双方向性のあるコンテンツのプログラミング③

チャットツールをデザインしよう！

## 今日のめあて

- ユニバーサルデザインに配慮した、誰にでも使いやすい伝言板を作ろう！

<ここで質問>

ユニバーサルデザインってなに？

# プログルを使う準備をしよう



## プログル技術をはじめる

先生	生徒	お試しレッスン
<a href="#">新規登録 &gt;</a>	<a href="#">新規登録 &gt;</a>	<a href="#">お試しレッスン &gt;</a>
<a href="#">ログイン &gt;</a>	<a href="#">ログイン &gt;</a>	

# ログイン画面



クラスID

出席番号

パスワード

① クラスID

② 出席番号

③ 自分で作ったパスワード

ログインする

# Lesson1 伝言板をデザインしよう

# 1

Lesson 1

## 伝言板をデザインしよう！

Design a message board!

クラス内で使う伝言板のデザインに関して学んでいくよ！使う人にとって見やすいデザインを考えながら伝言板の画面を作成しよう！

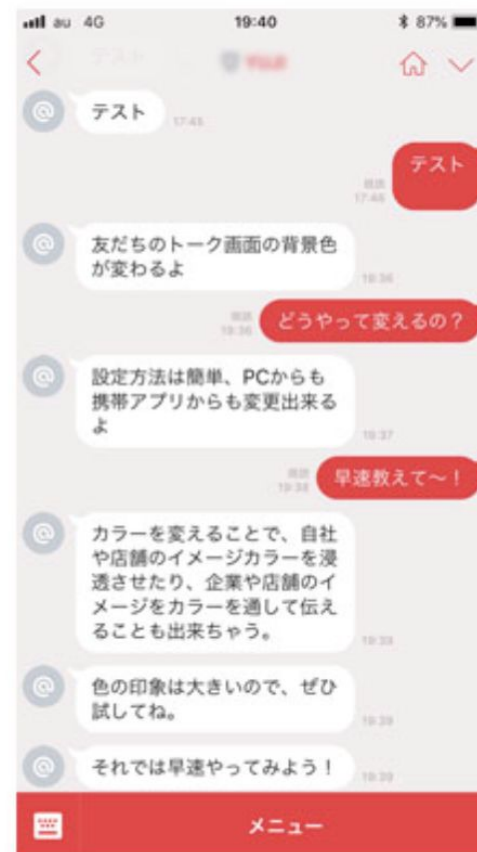
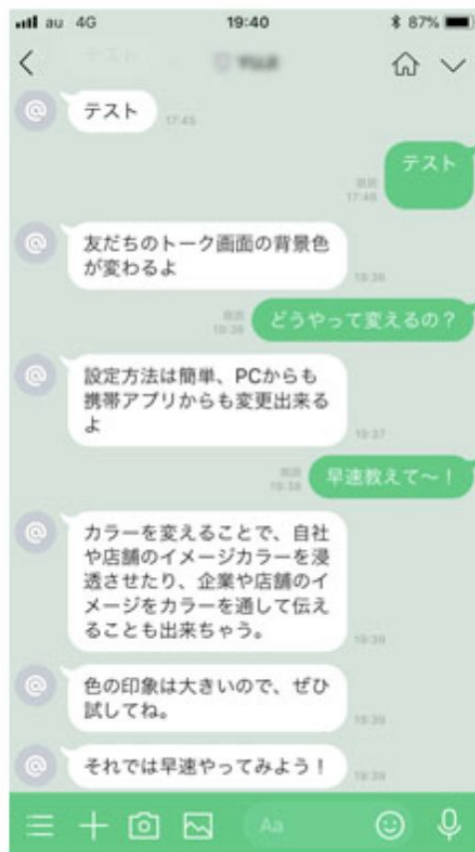
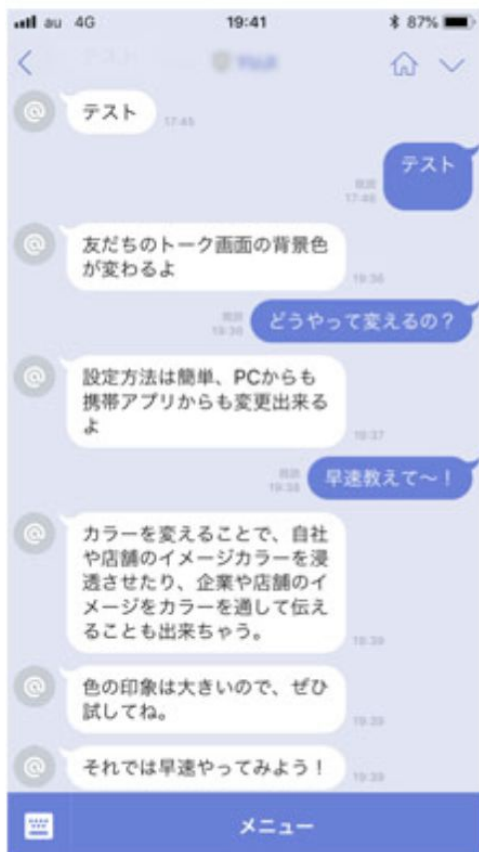


🏠 ステージ 0/8

はじめから進める

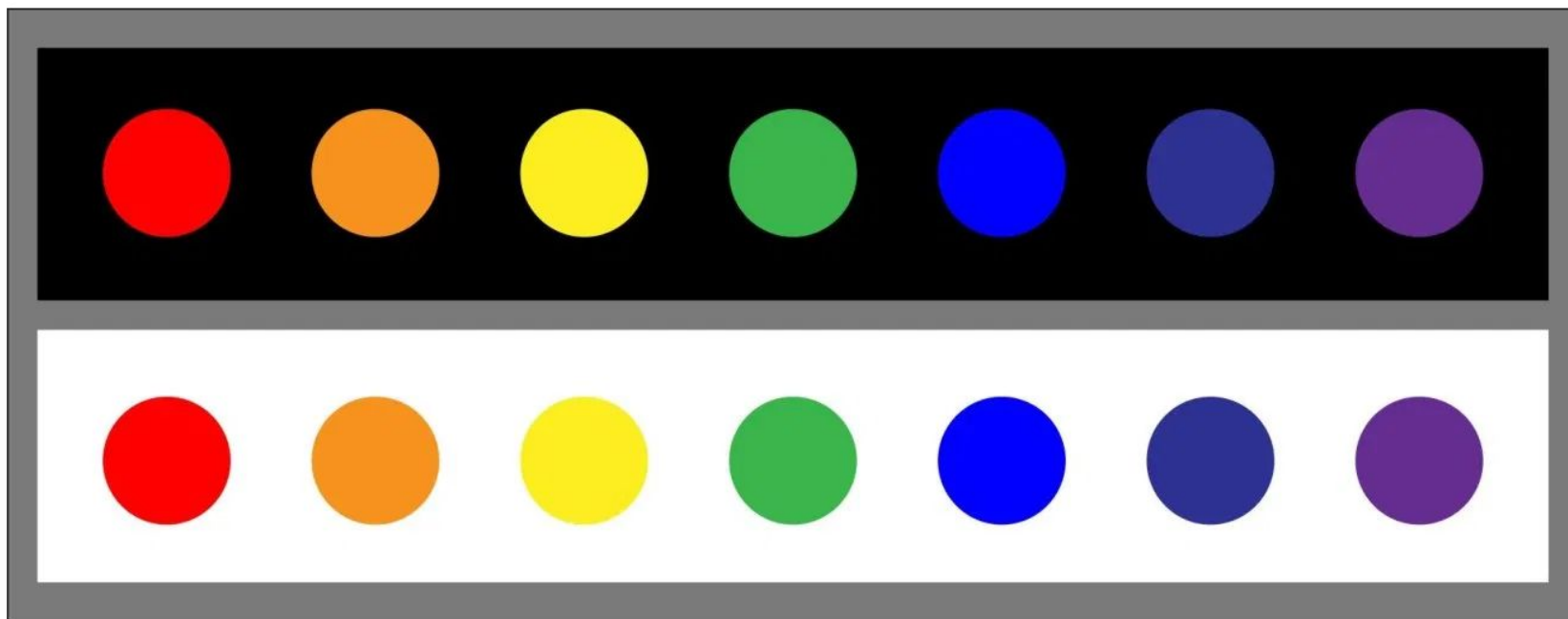
# 伝言板のデザインって？

- 色を変えるだけで、印象が大きく変わる



# 見やすい色って何だろう？

- どの色に目がいきますか？



# 町中でもよく見かけますよね？



# 見やすい色って何だろう？

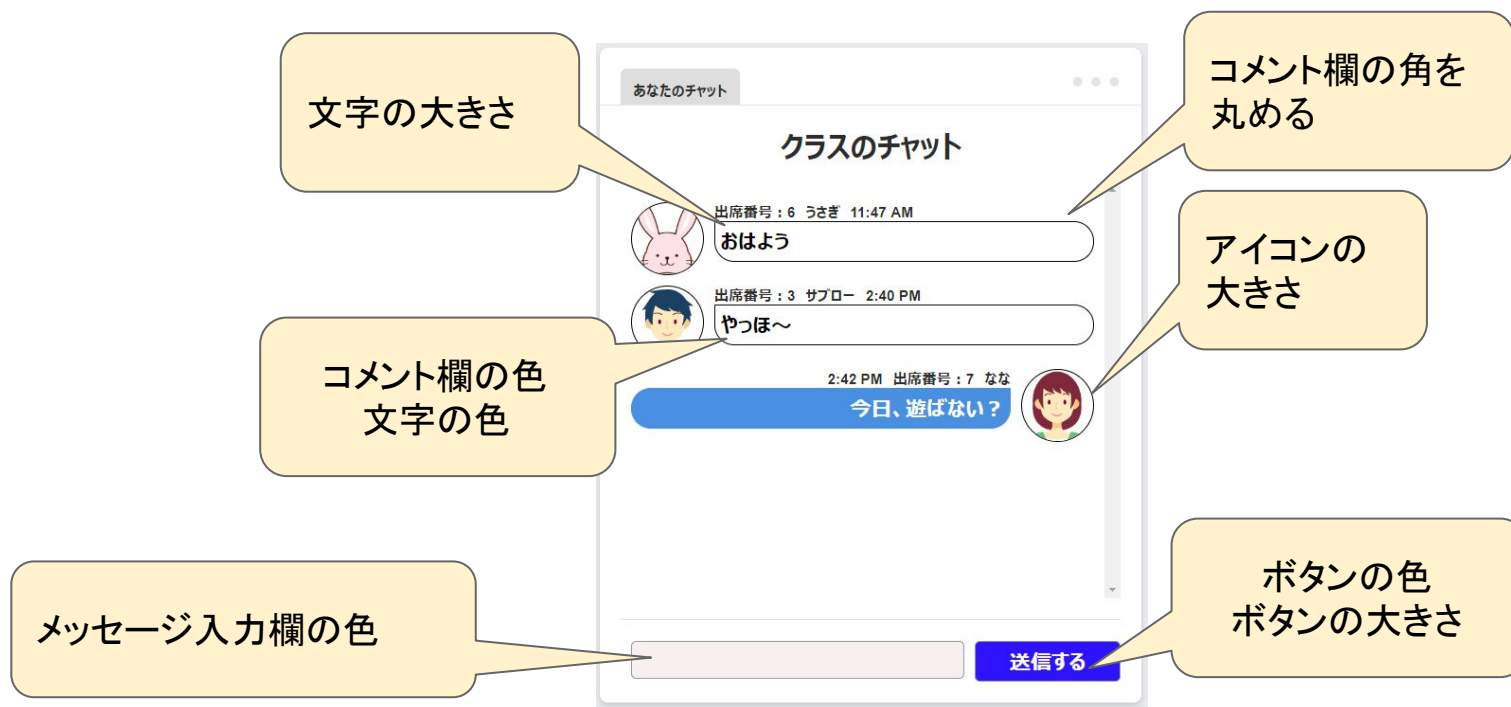
明度差があると文字が読みやすい

明度差があると文字が読みやすい

明度差があると文字が読みやすい

# 本日のめあて

- チャットツールのデザインを行ってみよう
- ユニバーサルデザインに配慮した画面をプログラミングしてみよう



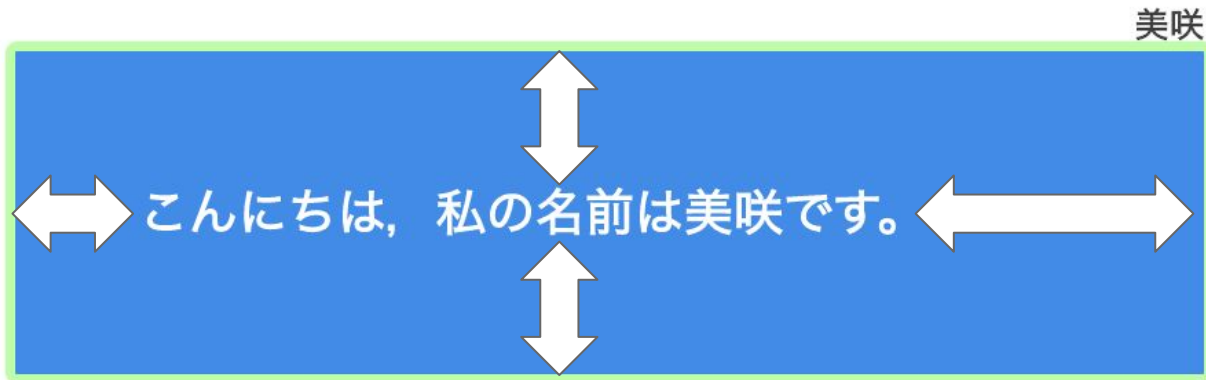
# ステージ1

- 文字の色と背景の色を見本と同じ色に変更してみよう。
  - 文字: 白
  - 吹き出し: 青



# ステージ2

- 文字の大きさ → 16px
- 左揃え
- 余白の設定 → 20px



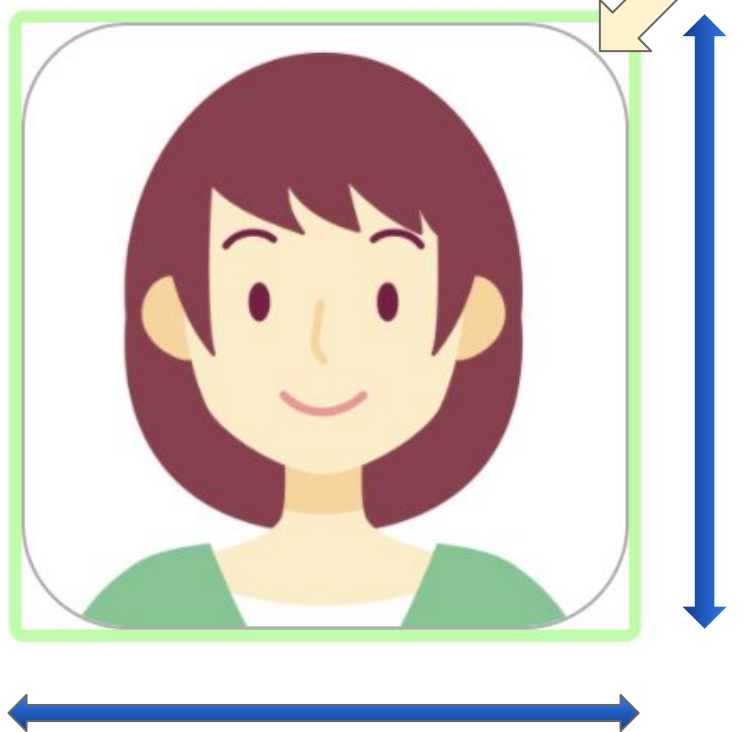
# ステージ3

- 3つの角を50pxに設定しましょう。



## ステージ4 画像サイズと角丸について

アイコンの  
角Rを指定



縦・横の  
大きさを指定

# ステージ5 テキスト入力エリア

- テキスト入力エリアの枠線 → 1px
- 文字の大きさ → 16px
- 余白を上下左右 → 14px



テキスト入力  
エリアはここ

# ステージ6 送信ボタンのデザイン

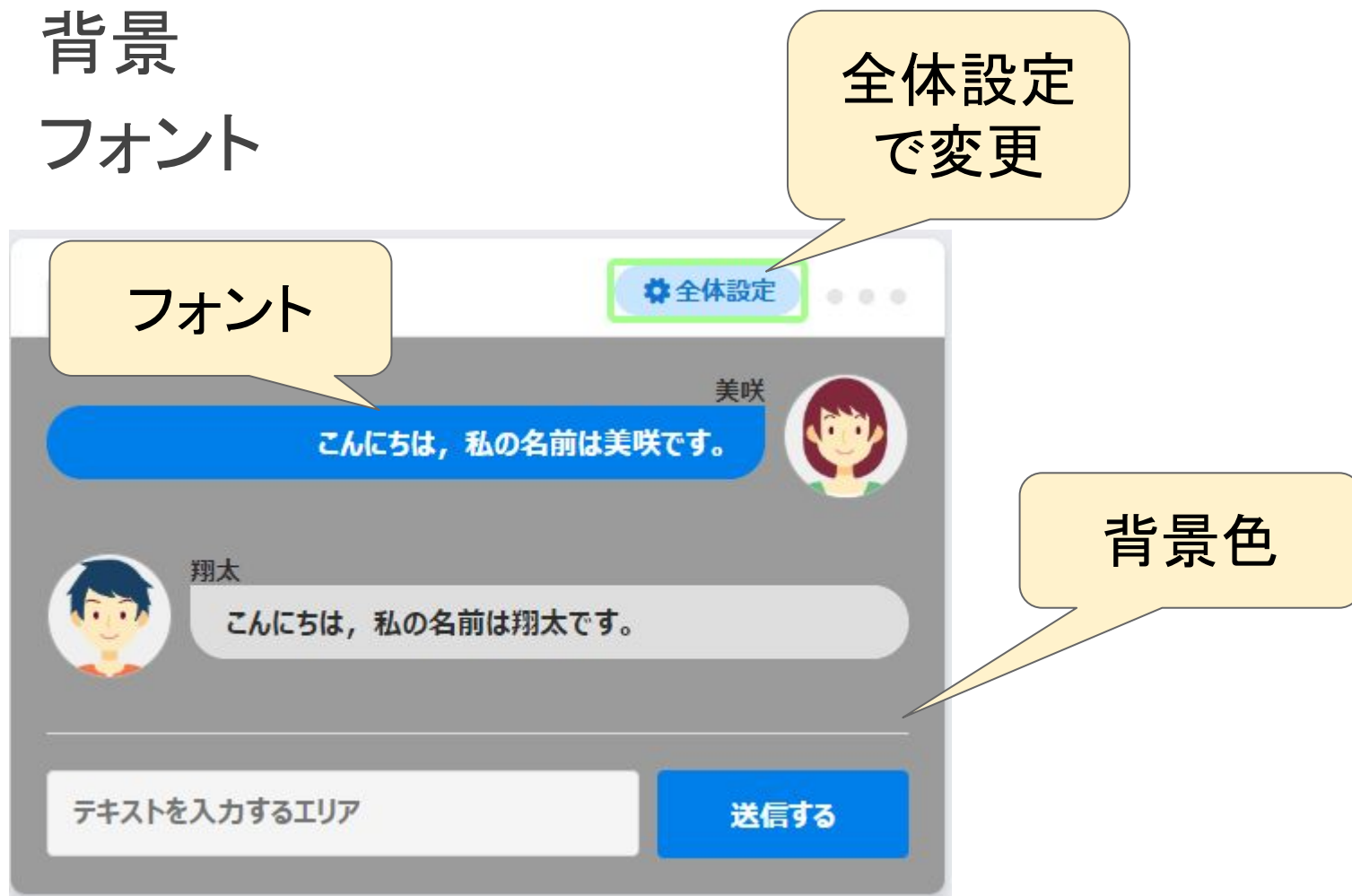
- 送信ボタンの文字の大きさ →16px
- 文字の配置を中央揃え
- 余白を上下左右 →14px
- 全ての角 →4px



送信ボタンは  
ここ

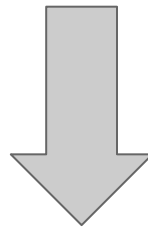
# ステージ7

- 背景
- フォント



# ステージ8 まとめ

- ステージ1～7までで学習したことを生かして、画面の制作をしよう。
- **ユニバーサルデザイン**を意識して制作しよう



この掲示板を使う人はどんな人？

この掲示板を使って欲しい人はどんな人？

# この時間の振り返り

- プリントに自己評価をしましょう
- 工夫した点をアピールしましょう

<書き方の例>

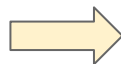
～のために、〇〇を大きくしました。

～のために、△△を選びました。など...

コメントが書けたら1～4で自己評価しよう

## 今後の注意

- 次回の授業でLesson2以降を学習していきます。そこで使用する画面デザインは、本日制作したLesson1-ステージ8のものが使用されます。



Lesson1-ステージ8に  
戻って修正をしてください

# 双方向性のあるコンテンツのプログラミング④

サーバと通信しよう！

# プログラブル技術

プログラブル技術は、中学校・技術で使える  
「双方向性のあるコンテンツのプログラミング」に特化した教材です。

## プログラブル技術をはじめる

先生

新規登録 >

ログイン >

生徒

新規登録 >

ログイン >

通常の利用は  
こちら

# ログイン画面



クラスID

①クラスID

出席番号

②出席番号

パスワード

③自分で作った  
パスワード

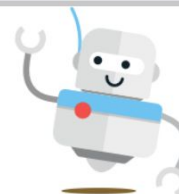
ログインする

# この画面になったら準備完了



あなたの作成したチャット  
チャットはレッスン5で完成します

[チャットを確認する](#)



# 1

Lesson 1

## 伝言板をデザインしよう！

Design a message board!

クラス内で使う伝言板のデザインに関して学んでいくよ！使う人にとって見やすいデザインを考えながら伝言板の画面を作成しよう！



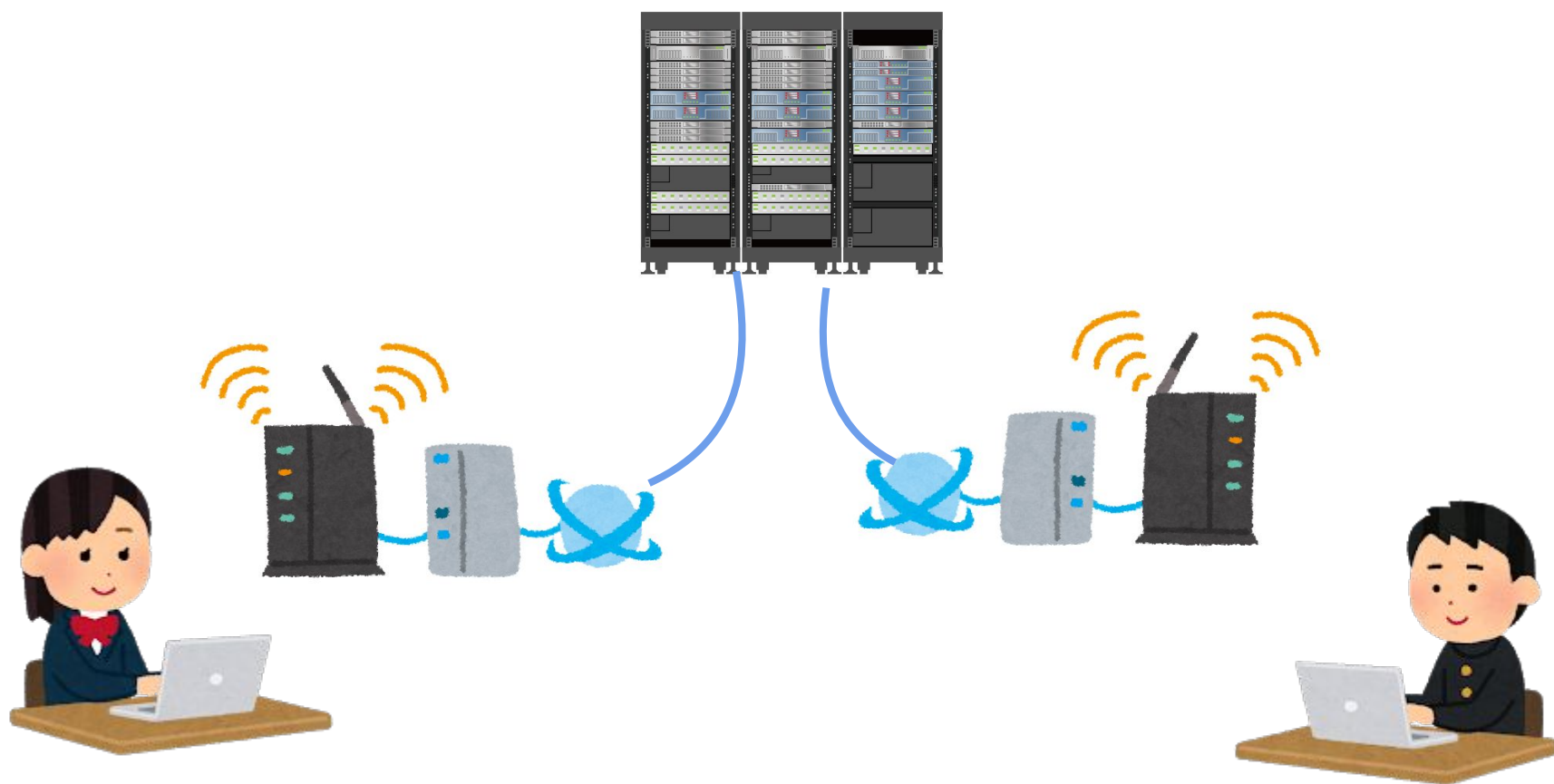
# この時間のめあて

- サーバと通信してみよう



# この時間のめあて

- サーバと通信してみよう



# この時間のめあて

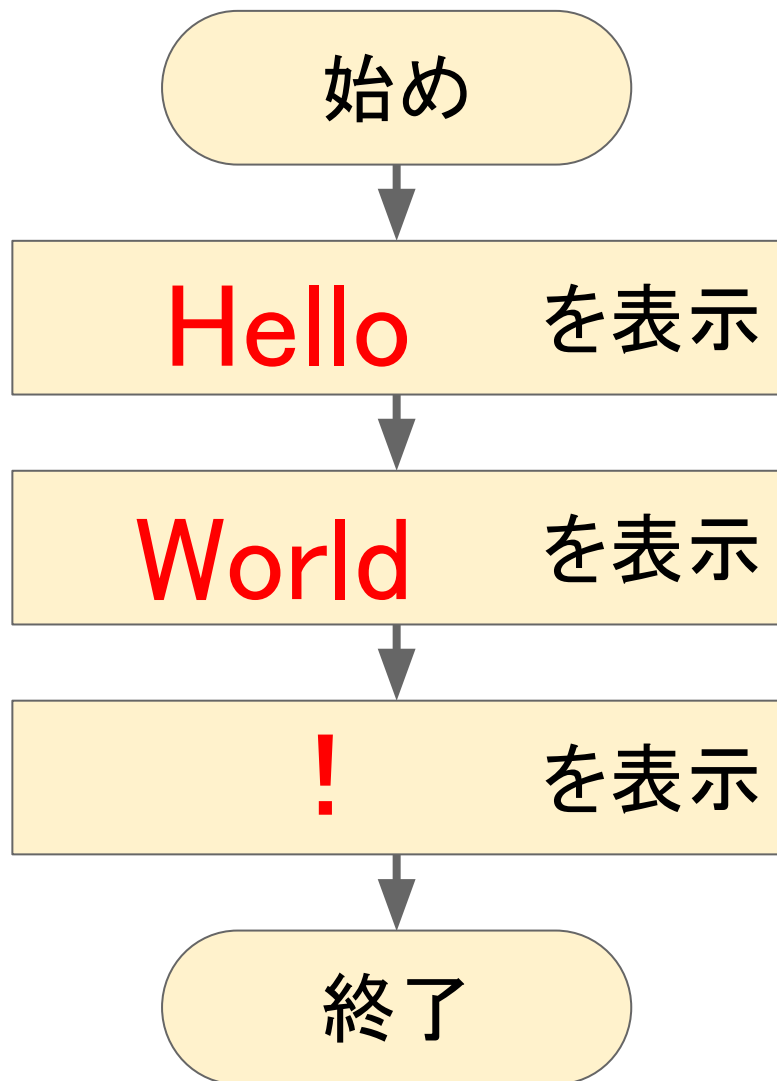
- 順次処理、反復処理、分岐処理を理解してプログラムを作ってみよう。

# ステージ1

- 画面にHello World! と表示させよう



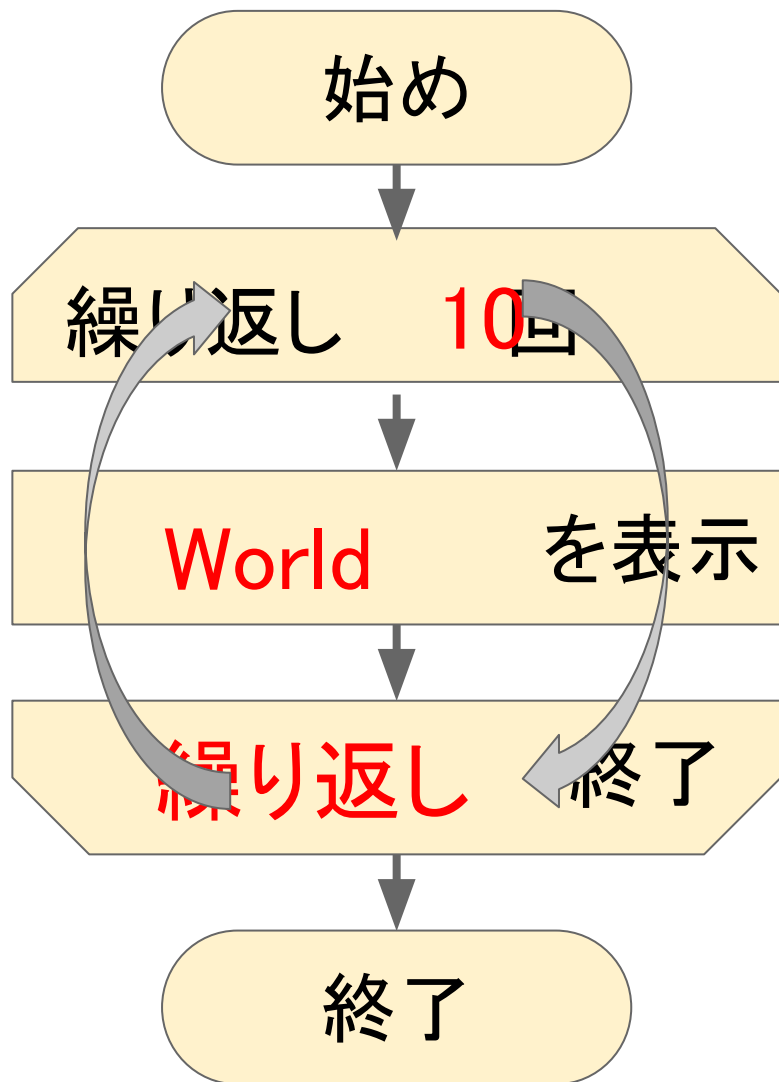
## ステージ2 順番に表示させよう



左図のことをフローチャート  
という

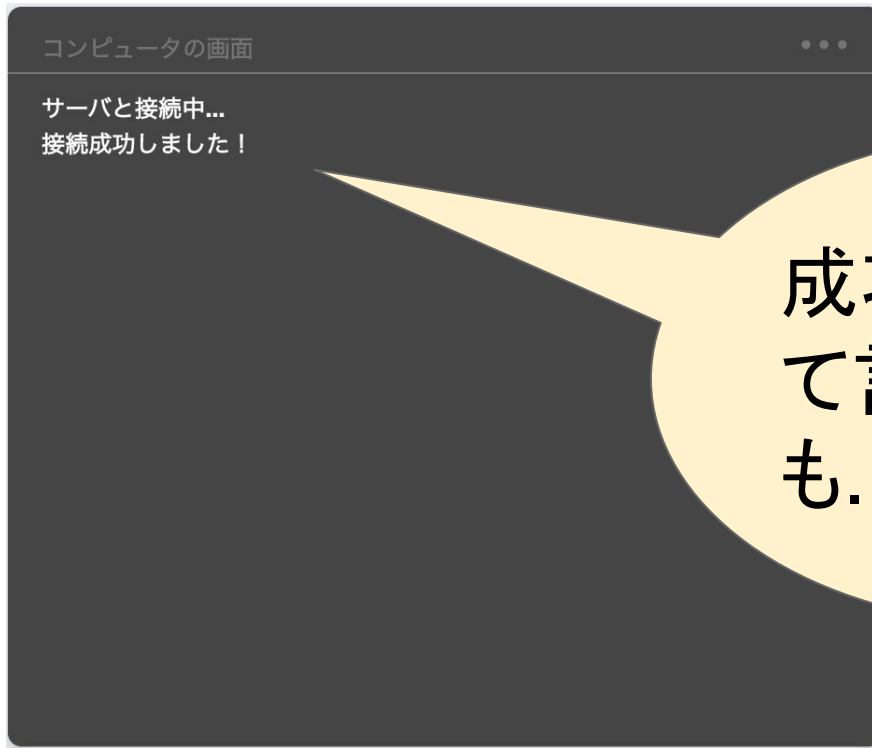
上から順番に実行されるプ  
ログラムのことを順次処理  
という

# ステージ3 繰り返し表示させよう



同じ処理を決められた条件で繰り返し処理することを**反復処理**という

# ステージ4 サーバへの接続



成功しましたっ  
て言われて  
も... 😊💧

# ステージ5 サーバへのメッセージ送信

コンピュータの画面

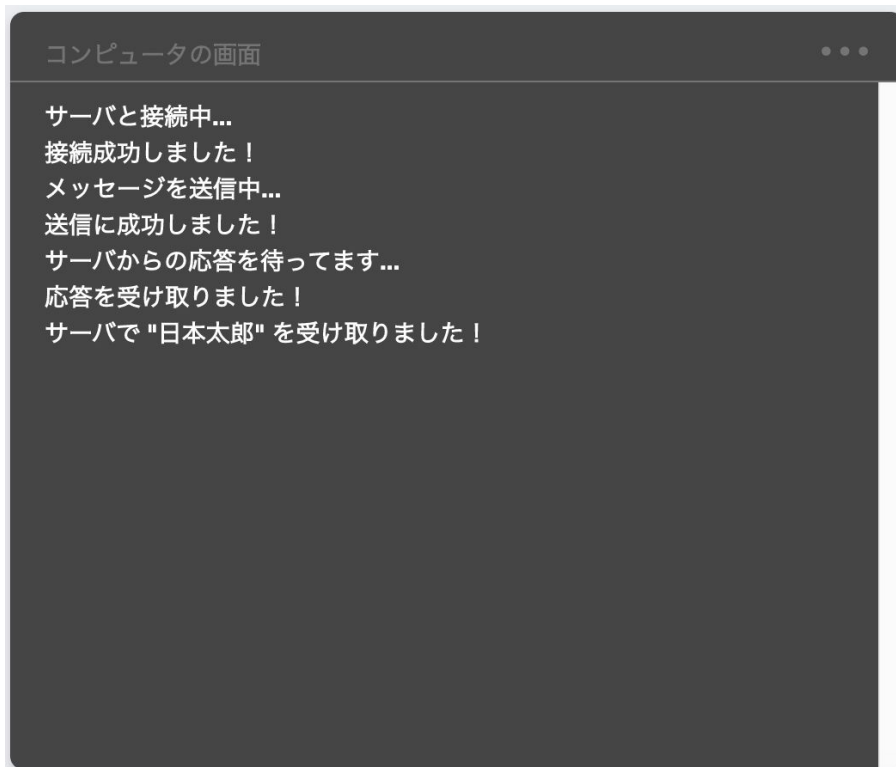
サーバと接続中...  
接続成功しました！  
メッセージを送信中...  
送信に成功しました！

成功しましたって言われても...「こんにちは」が出てないぞ

前のスクリーンに注目

# ステージ6

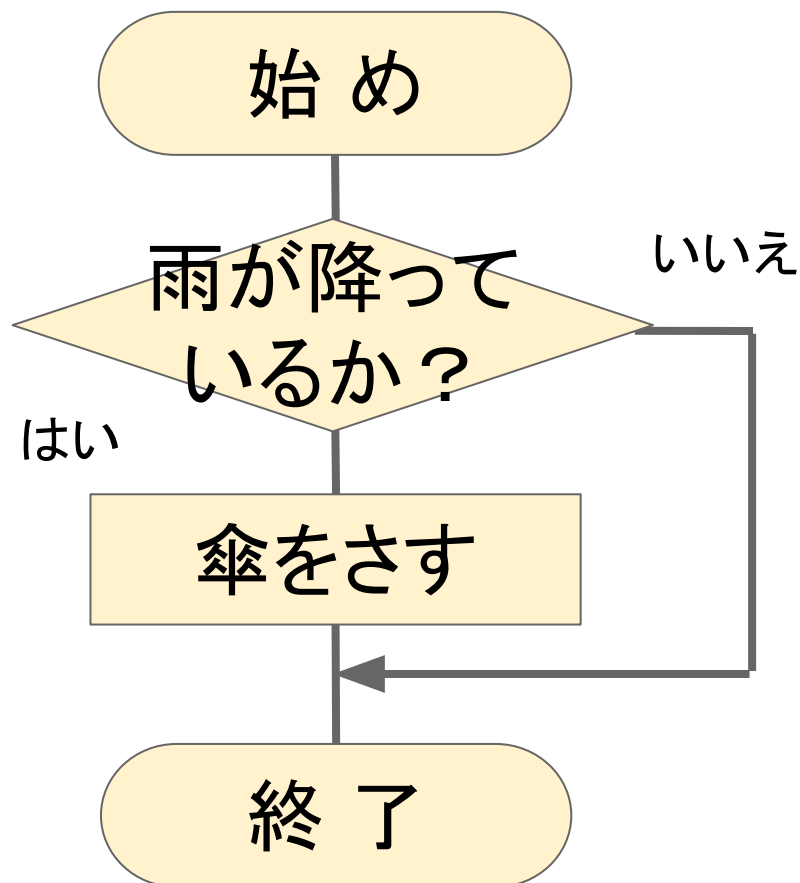
- サーバにニックネームを送る
- 応答の内容をコンピュータに表示する



```
コンピュータの画面
サーバと接続中...
接続成功しました！
メッセージを送信中...
送信に成功しました！
サーバからの応答を待ってます...
応答を受け取りました！
サーバで "日本太郎" を受け取りました！
```

# ステージ7 分岐処理

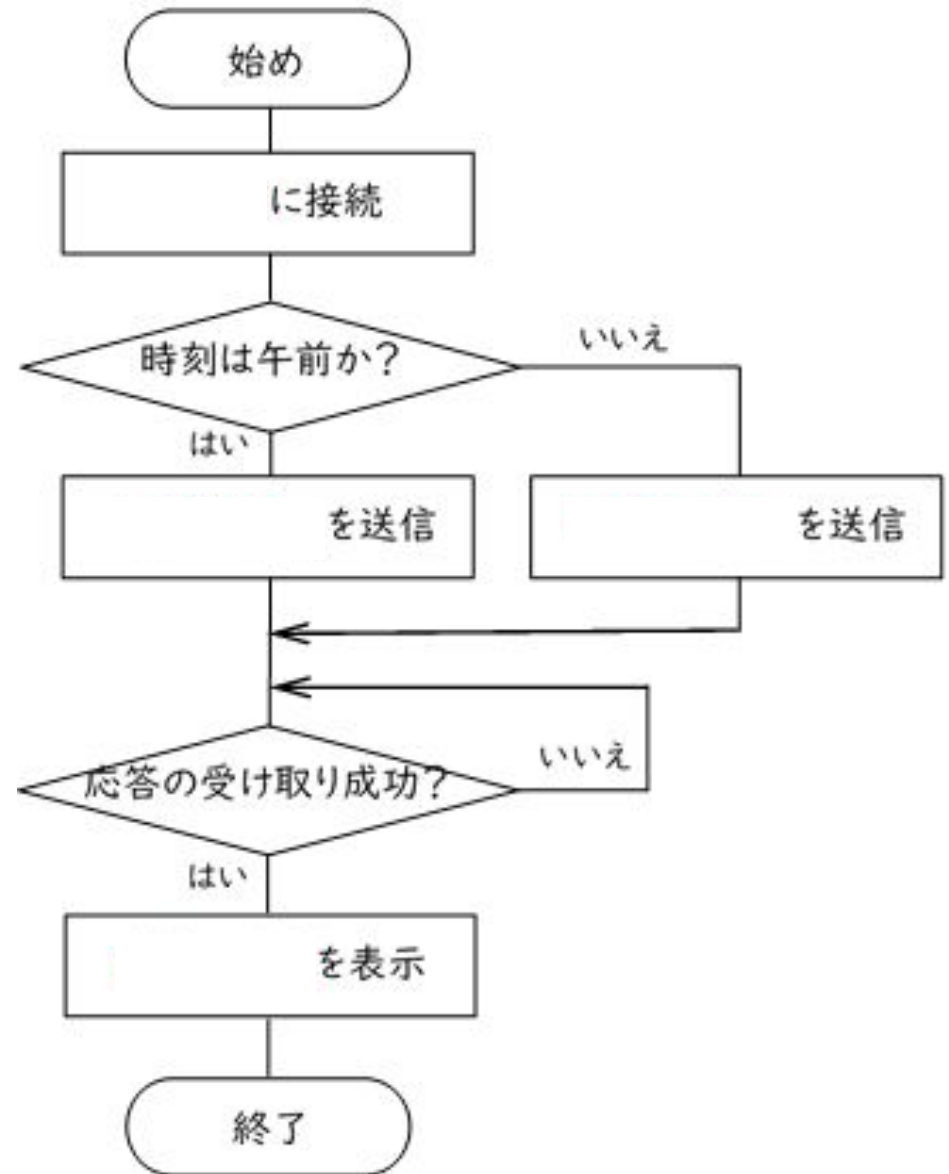
- 条件によって処理を選択する **分岐処理**



# ステージ7 分岐処理

- 作ったプログラムをフローチャートで表してみよう

<ステージ7のフローチャート>



# この時間の振り返り

- サーバにメッセージを送ることができた
- 順次処理・反復処理・分岐処理を理解することができた

# 双方向性のあるコンテンツのプログラミング⑤

文字を送受信しよう！

# この時間のめあて

1. サーバとメッセージの送受信を行う
2. 送受信するメッセージを画面に表示する

# 今日の授業で学ぶこと「文字の送受信」

④ 友達のメッセージも表示

チャット

クラスのチャット

③ コメントが表示される

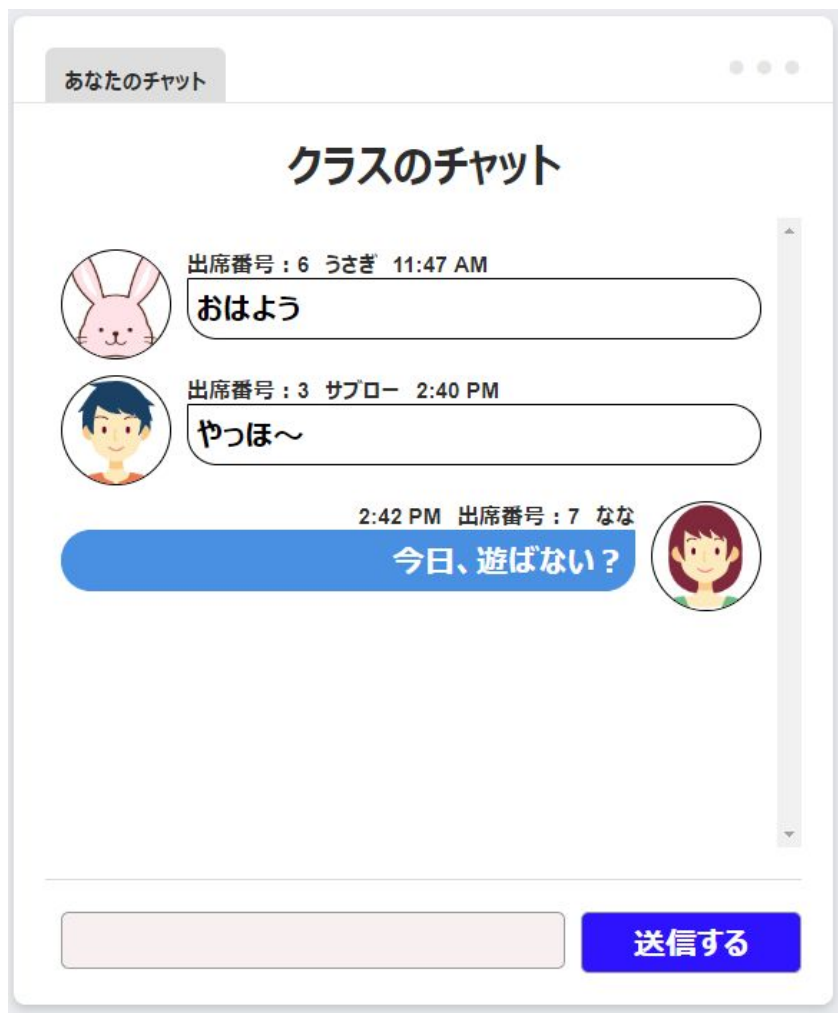
① 入力エリアに文字を入力

今日、遊ばない？

送信する

② 送信する

# チャット画面を変更したくなった場合



- 修正したくなった場合は、Lesson1、ステージ8に戻り、変更するとLesson3にも反映されます。

# ステージ1

- 送信ボタンを押したら「こんにちは」と表示しよう

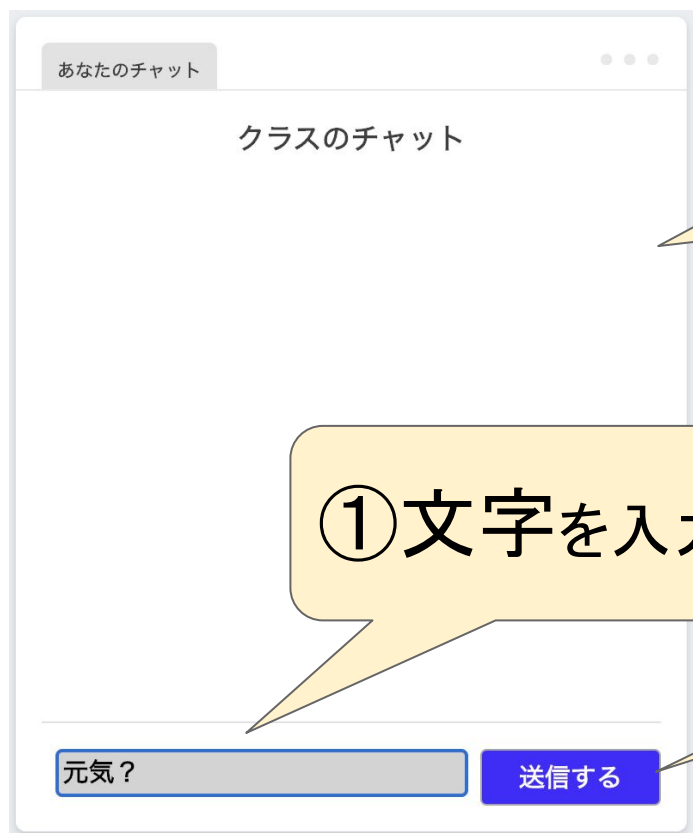


② こんにちは

① 送信するを押すと

## ステージ2

- 送信ボタンを押したとき、**変数**[送信テキスト]に入力エリアの内容を保存し、表示しよう。



# 変数とは？

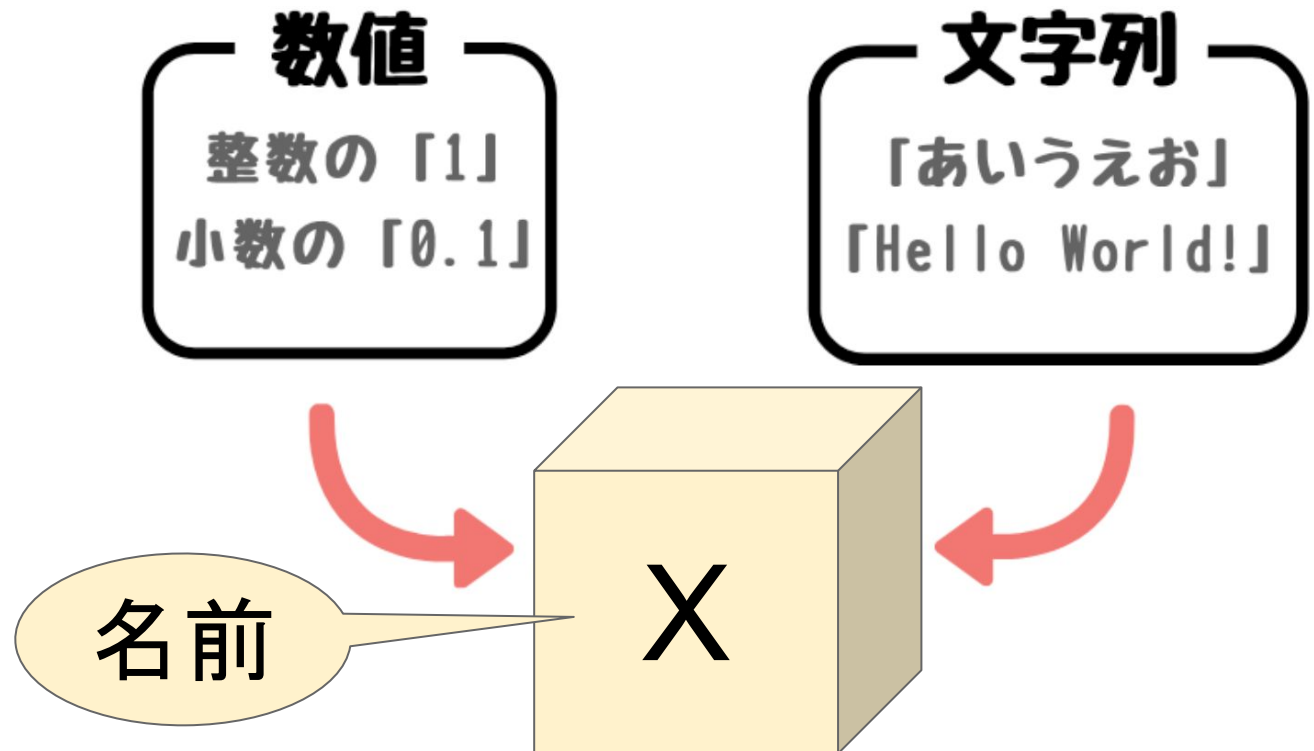
- 以下の式のXとYに、それぞれX=5、Y=3を入れたら答えはいくつ？

The diagram shows the equation  $2X + Y = 13$ . The variable  $X$  is represented by a yellow 3D box with a speech bubble above it containing the number 5. The variable  $Y$  is represented by a yellow 3D box with a speech bubble above it containing the number 3. The equation is written as  $2$  followed by the  $X$  box, a plus sign, the  $Y$  box, an equals sign, and the number 13.

$$2 \text{ X } + \text{ Y } = 13$$

# 変数とは？

- 数値や文字をしまっておける箱
- 名前を自由につけることができる



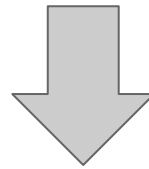
# ステージ3



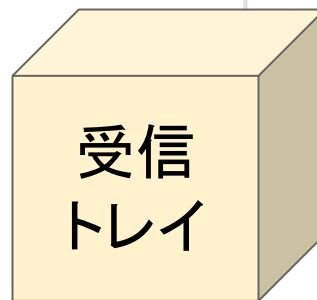
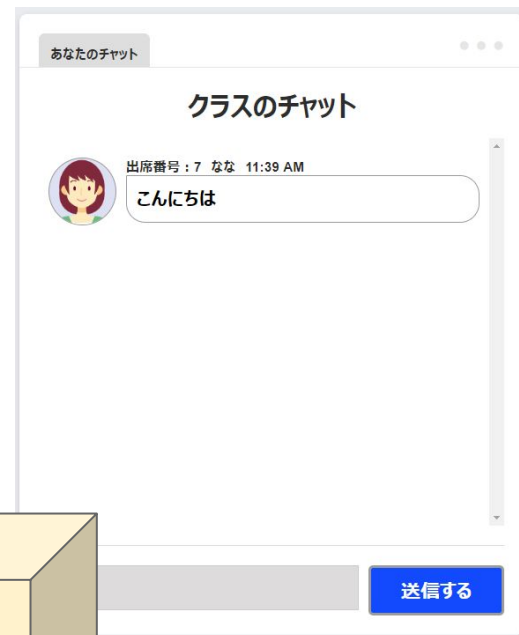
ステージ2と何が違うの？

# ステージ4

- 新着メッセージを受信したとき，変数 [受信テキスト] に新着テキストを保存し，表示しよう



サーバにあるメッセージを取ってきて表示する



# ステージ5

- 今まで出てきたブロックを使ってテキストの**送受信**ができるようにしよう！

考え方は

ステージ3 + ステージ4  
(送信) (受信)

# 今日のまとめ

## 1. サーバとメッセージの送受信を行う

→ 送受信するときに変数にメッセージを貯めておく

## 2. 送受信するメッセージを画面に表示する

# 双方向性のあるコンテンツのプログラミング⑦

画像を送受信しよう！

# 今日の授業で学ぶこと「画像の送受信」

1. 画像を表示する
2. 画像を圧縮する&サイズ修正(リサイズ)する
3. 大きい画像のリサイズを修正する

# 今日の授業で学ぶこと「画像の送受信」

4. 画像を圧縮 & リサイズして、サーバに送信  
→ 成功したら画面に表示
5. 選んだ画像を変数に保存 → 画像を表示
6. 選んだ画像を変数に保存 → 圧縮 & リサイズ →  
サーバへ送信 → 画像を表示

# ステージ1 「表示」

- サンプル画像を表示

## ステージ2 「圧縮とリサイズ」

- 圧縮率10%
- 幅と高さをそれぞれ300pxにリサイズ

※圧縮とは:色の数や目の細かさを粗くすることによってデータ量を減らすこと。

圧縮率が高いほどデータは小さくなるが、画質は悪くなる。

※リサイズ:画像の大きさを変更すること

## ステージ3 「大きい画像のリサイズ」

- サンプル画像の幅と高さが500px以上の場合、幅と高さをそれぞれ300pxにリサイズ

<参考>

写真などの縦横比のことをアスペクト比という

ちなみに フィルムカメラ 3:2


デジタルカメラ 4:3

液晶テレビ 16:9


## ステージ4 「リサイズした画像の送信」

- サンプル画像を圧縮 & リサイズ
- サーバに送信
- 成功したら画像を表示

# ステージ5「自分で選んだ画像を表示」

- を押して、画像を選択し**変数**に保存
- **変数**に保存した画像を画面に表示

# ステージ6

- を押して画像を選択する
- 選んだ画像を変数に保存する
- 圧縮やリサイズを行う
- サーバーに送信する
- 送信に成功したら、画面に表示

# 画像の圧縮 & リサイズについて

- 通信量を減らすためにできることはなんだろう？
  - プログラミングで、できること
  - デジカメなどの操作でできること
- リサイズしていて困ることは？

# ステージ7

- 新着メッセージ(画像)を受信したとき、変数に新着画像を保存する
- 変数に保存した新着画像を表示する

## ステージ8

- 新着メッセージ(画像)を受信したとき、変数に保存する
- 新着メッセージ(画像)に圧縮やリサイズを行う
- 画面に表示する

## ステージ9

- 画像の送受信ができるようにしよう

考え方は  
ステージ6 + ステージ8  
(送信) (受信)

## ステージ9

- 下のブロックを活用すると大きいサイズの画像もちょうど良い大きさに表示することができるので、活用しよう



- どうすれば通信量を削減できるか？通信網の混雑を減らせるか？を考えてプログラムを工夫してみよう

# 今日のまとめ

1. サーバと通信して画像を送受信することができた
2. 送受信する際に、データ量を考慮してファイルの圧縮やリサイズができた

# 双方向性のあるコンテンツのプログラミング⑧

伝言板を完成させよう！

# 今日の授業で学ぶこと「掲示板の制作」

## 1. 画面のデザインを制作しよう

→ ユニバーサルデザインに配慮する

## 2. テキストと画像の送受信ができるようにプログラミングしよう

→ データの通信量を節約するには....

→ 写真を見やすく表示するには...

※わからなくなってしまった人は以前のLessonやプリントに戻って考えてみましょう

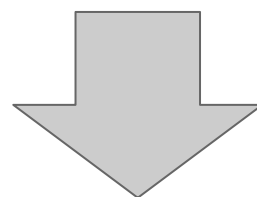
# 課題を解決するブロック

- Lesson5-2では以下の3つのブロックが使えます

電話番号  
検出ブロック

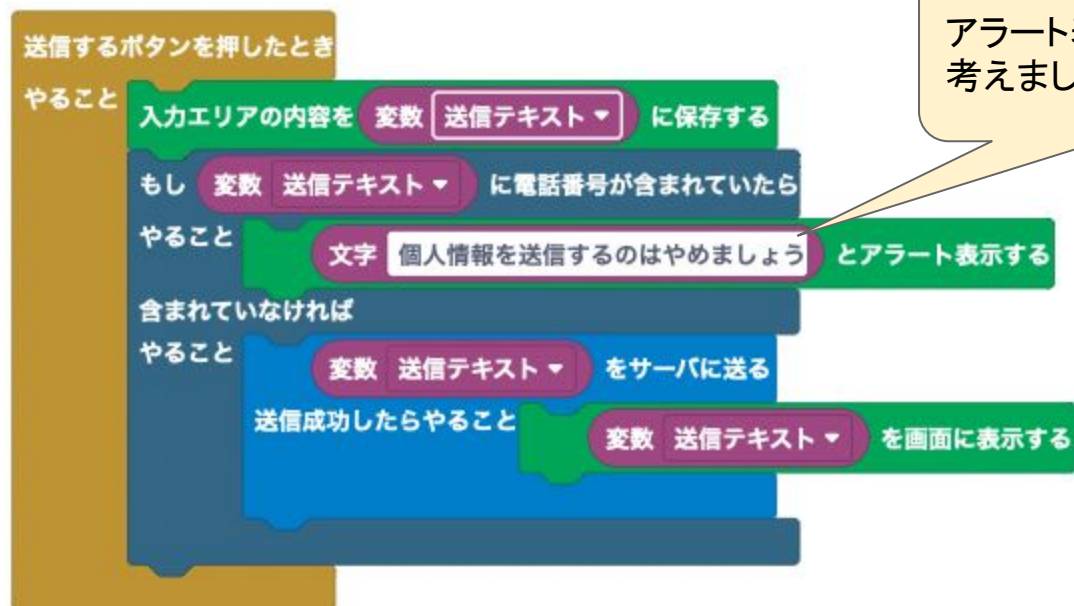
メールアドレス  
検出ブロック

キーワード  
検出ブロック



これらを活用して、課題を解決してみよう

# 電話番号が含まれていたら...

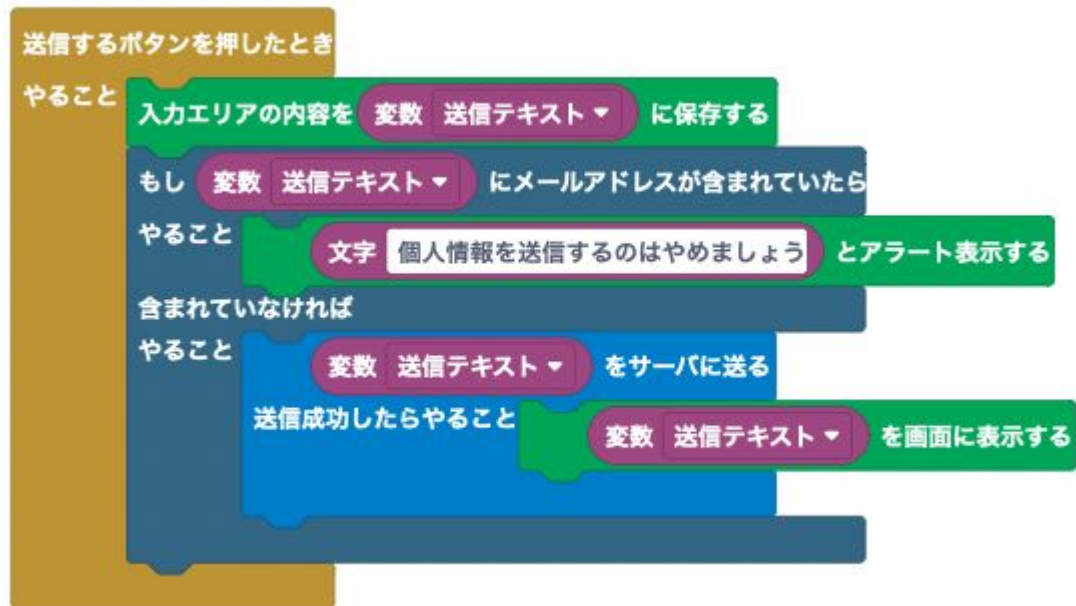


アラート表示の文章も  
考えましょう



アラート表示の例

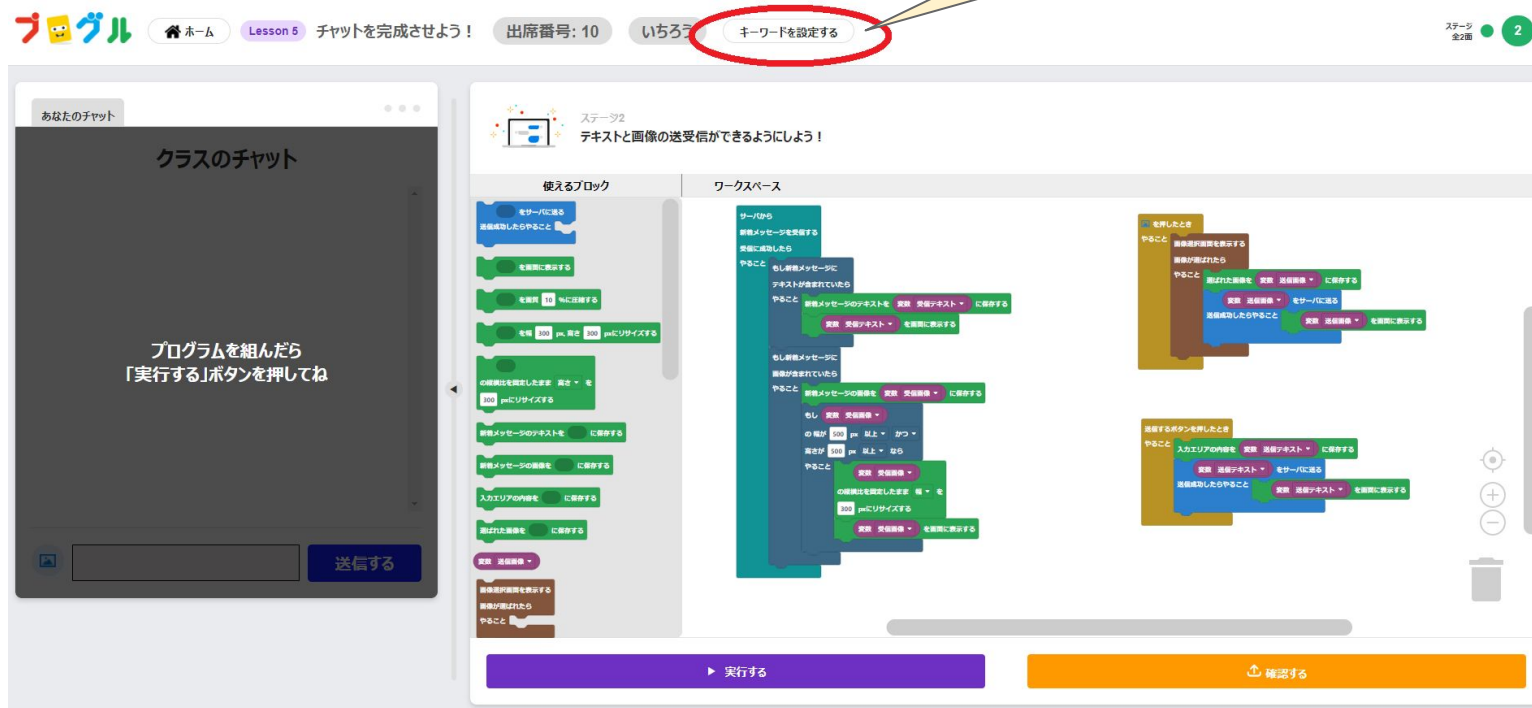
# メールアドレスが含まれていたら...



# キーワードを設定する(最大10個)

- Lesson5-2の画面

キーワードを設定する



# キーワードを生徒ごとに設定(10個)

最大10個まで  
登録できます

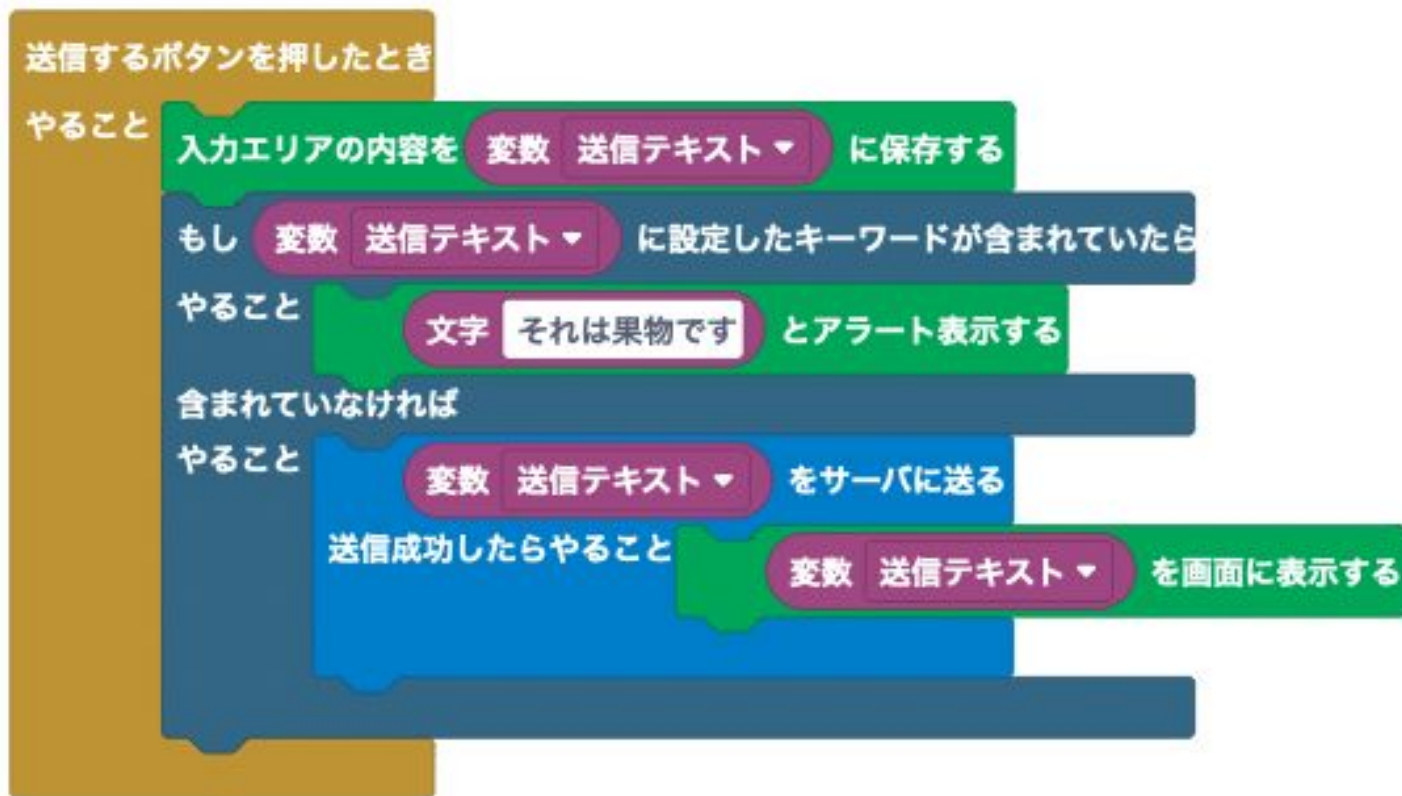
## キーワード一覧

キーワードを追加する

キーワードの上限は10個です

	キーワード	削除
1	りんご	🗑
2	バナナ	🗑

# 設定したキーワードによって分岐処理



# 設定したキーワードを出力してテキストマイニング

- クラス一覧 → チャットを見る

The screenshot shows the 'ブグル' (Boguru) teacher management interface. The top navigation bar includes the logo, '先生用管理画面' (Teacher Management Screen), and 'ログアウト' (Logout). The left sidebar contains '管理メニュー' (Management Menu), 'クラス一覧' (Class List), and 'アカウント設定' (Account Settings). The main content area is titled '2020年度 2-A (研修用) のチャット' (2020 Academic Year 2-A (Training) Chat). It features two tabs: 'レッスン' (Lesson) and 'チャット' (Chat), with the latter selected. Below the tabs, there is a button 'キーワード一覧のCSVファイルをダウンロードする' (Download CSV file of keyword list), which is circled in red. Underneath is a button 'チェックしたメッセージを削除' (Delete checked messages). A table displays chat messages with columns for selection, attendance number, name, send time, message content, and delete action.

<input type="checkbox"/>	出席番号	名前	送信時間	発言	削除
<input type="checkbox"/>	10	どんべえ	2021/01/14 15:41:51	あいうえお	🗑️
<input type="checkbox"/>	10	どんべえ	2020/11/02 17:17:03	aaa	🗑️

# AIテキストマイニングで分析する

- CSVファイルをアップロードする

<https://textmining.userlocal.jp/>

UserLocal AIテキストマイニング [分析結果をメディアで利用するには] 新規登録 ログイン

新規作成 文書一覧 カスタマイズ 前処理ツール ヘルプ

Googleサーチコンソールのキーワードをテキストマイニングする

無料で使える手書き文字認識(OCR)ツール (New!)

解析したいテキストを入力する

1つの文書を解析 2つの文章を比較 Excelクロス集計 音声入力

サンプル: 太宰治『走れメロス』

フォーム入力 ファイルをアップロード 履歴から選択

0 / 10,000文字  
文字数上限を増やす



## 今日のまとめ

- これまでの学習であなたが気づいたこと、考えた事を書きましょう。

→書けた人は振り返りで自己採点も

行いましょう

# 双方向性のあるコンテンツのプログラミング⑧

## 今日のめあて

- 完成したチャットツールを友達と評価し合い、改善につなげよう
- チャットツールの利用規約について考え、お互いに気持ちよく利用できるようにしよう

# グループを作ってお互いに評価しあいましょう

次の項目についてチェックしましょう

- 見やすさ
- 操作性
- 効率
- 上記以外に良い点/惜しい点

## 利用規約とは

- 利用規約とは、掲示板の利用者が守らなくては  
いけない( )をまとめたものです。利用  
者のモラルだけに頼って運営するのではなく、文  
章として( )を示すことも大切です。

## 利用規約を考えてみよう

- プリントのキーワードを参考に利用者が気持ちよく使える規約(ルール)を考えて、プリントに記入しましょう

# 掲示板の改善

- これまでの授業を振り返り、制作したプログラムの修正/改善点をあげましょう
- また、新たに追加してみたい機能のアイデア（あくまでアイデアなので、実現できなくてもよい）をあてましょう