

第4章 SNSの仕組みを理解し、プログラムを改善しよう

1 SNSを理解し、プログラムを体験する



【プログラムの概要説明】

2台のPCを準備し、1台目のPCをクライアントA（送受信者）、2台目のPCをクライアントB（送受信者）とする。

互いのコミュニケーションのやり取りを一画面で行い、メッセージが吹き出しとして表示される。自身のキャラクターは大きく、相手方のキャラクターは小さく表示されている。

自身のキャラクターをクリックし、メッセージボックスに文字を入力する。Enterキーを押すことでメッセージが送信される。送信されたメッセージは相手方で、吹き出しとして表示されるようになっている。相手方でも自身のキャラクターをクリックし、メッセージを入力してやりとりできるようになっている。

（1）プログラムの体験（D-2）

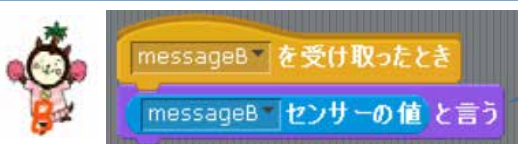
ここでは、二人一組でSNS上でのメッセージのやり取りを想定して、まずは自由にやり取りをさせる。



○ D-2-SNS-AB 側



【プログラム内容】
スプライト4がクリックされたら、「答え」を聞いてくる。その「答え」を変数 (messageA) として送る。



【プログラム内容】
messageBを受け取ったら、吹き出しとして表示される。

○ D-2-SNS-BA 側



【プログラム内容】
スプライト4がクリックされたら、「答え」を聞いてくる。その「答え」を変数 (messageB) として送る。



【プログラム内容】
messageAを受け取ったら、吹き出しとして表示される。

《送信者 (クライアント A) の操作手順》

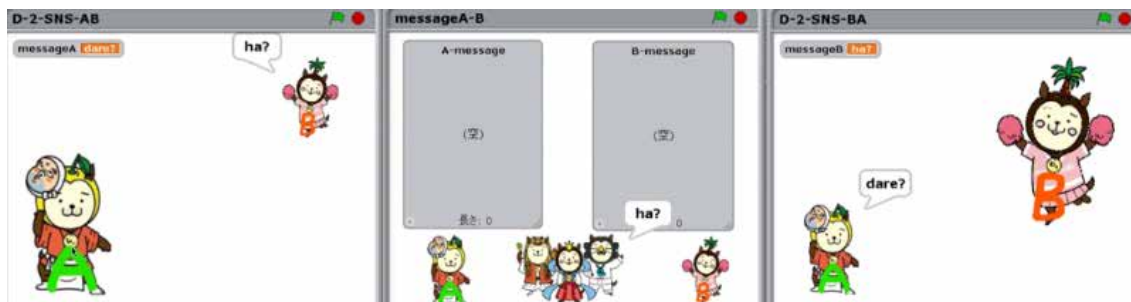
- ① キャラクター (ひよつこの面のみやぎき犬 = スプライト4) をクリックする。
- ② 文字の入力を半角英数で行う。(残念ながら、ひらがなの入力はいできない)
- ③ 入力後、Enter キーを押す。 転送

《受信者 (クライアント B) の操作手順》

- ④ 受信すると、キャラクター (ひよつこの面のみやぎき犬) に吹き出しとして表示される。

互いのコミュニケーションのやりとりになるため、相手方でも同じ動きとなる。

(2) SNSの問題点を見出す(D-2・3)【情報モラルや健康面に関して考える】



【プログラムの概要説明】

3台のPCを準備し、1台目のPCをクライアントA(送受信者)、2台目のPCをサーバ、3台目のPCをクライアントB(送受信者)とする。

入力された情報は、サーバを通して相手方に送信される。相手方の情報も同様である。サーバではそれぞれから送られてきたメッセージがリストに表示される。

【授業展開例】

プログラムを体験させる

SNSの特徴について意見を述べさせる

サーバの状態を見せる(記録が残っていることを確認する)

情報モラル・健康面に関して考えさせる

(1)で取り寄せたデータのやり取りに関して、どこかの班(ペア)のHOSTのIPを教師が聞いておき、教師がサーバ役としてメッセージを表示させておく。

授業展開の中で、メッセージがサーバに残っていることに気づかせ、不適切な発言やその情報が残っていること・完全に消すことが難しいことを指導内容に盛り込み、情報モラルに関する授業を展開していく。

その後、情報モラル・健康面の視点から個・ペア・グループ等で話し合い、見いだした改善点をアクティビティ図に描き、プログラム制作を通じて問題解決していく。

学習課題の設定

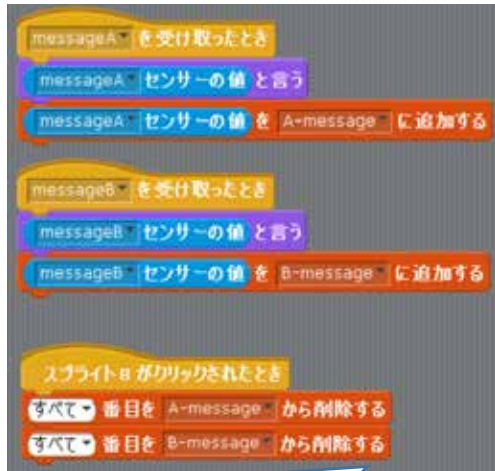
学習課題

情報モラル・健康面の視点から、SNSの問題を考えよう！

具体的な内容

- (1) SNSプログラムの自由なやりとりを行う。
- (2) 問題点を見出す。
- (3) 解説
 - ①情報モラルに関して
 - ・なりすまし、誹謗中傷、個人情報の流失、プライバシーの侵害、サイバーセキュリティなど
 - ②健康面
 - ・長時間の利用による体調の変化(睡眠不足・目の疲れ・ストレス・食欲不振など)
- (4) 補足 ~データがサーバ上に残っていることで、刑事事件として扱われることもある。
※ サーバ上の記録をみせる。
- (5) どのようなことに気をつけて取り扱っていかなければならないか話し合いをする。

サーバのプログラム



【プログラム内容】

messageA を受け取ったら、スプライト 8 (神楽スタイルの 3 匹のみやざき犬) の吹き出しとして表示され、その後、リスト (A-message) に表示される。
 messageB を受け取ったら、スプライト 8 (神楽スタイルの 3 匹のみやざき犬) の吹き出しとして表示され、その後、リスト (B-message) に表示される。
 スプライト 8 がクリックされたら、リストのデータが削除される。

第 4 章 SNS の仕組みを理解し、プログラムを改善しよう

2 利便性・安全性を高めるための機能について考える

知的財産を創造、保護及び活用しようとする態度、情報技術に関する倫理観や、他者と協働して粘り強く物事を前に進める態度を育成する。

これまで学んできた学習を基に、学習課題「みんなのためになる SNS を開発しよう」について考えさせ、設計 (アクティビティ図) させ、プログラムを制作させる。動作の確認及びデバッグをさせながら、問題解決していく。

そのためのプロセスとして、以下の手順で取り組んでいく。

技術の見方・考え方を働かせ、問題を見いだして、課題を設定

使用するメディアを複合する方法と効果的な利用方法の構想

情報処理の手順を具体化

安全・適切なプログラムの制作 / 動作の確認・デバッグ等

制作の過程や結果の評価、改善及び修正

(1) 問題解決のプロセスに従って学習を行う (学習の流れを一例紹介する)

技術の見方・考え方を働かせ、問題を見いだして、課題を設定

課題解決に向けた本コンテンツの問題点を洗い出す	課題解決に向けた本コンテンツの問題点を解決するために
問題点	解決策
<p>タイムを忘れて使いすぎてしまうんだよね・・・</p> <p>プライバシーに関わる他人の画像 (名前や画像など) を載せてしまうことがあるよね・・・</p> <p>相手の気持ちを考えず、思ったことを発信してしまうんだよね・・・</p>	<p>使い過ぎを防ぐために、時間制限をかけよう!</p> <p>始めるときに注意を促すコメントを出そう!</p> <p>自分の発言に責任もたけるために発信記録を表示させるとともに、誹謗中傷防止のための機能を搭載しよう!</p>

ここでは、以下の3つの解決策を設定した。

- ・始めるときに注意を促すコメントを出す
- ・使いすぎを防ぐために、時間制限をかける
- ・自分の発言に責任をもたせるために発信記録を表示させるとともに、誤送信防止のための機能を搭載する

今回はその中で、「使いすぎを防ぐために、時間制限をかける」に関して紹介する。

使用するメディアを複合する方法と効果的な利用方法の構想

新学習指導要領に「使用するメディアを複合する方法とその効果的な利用方法等を構想して情報処理の手順を具体化する」「コンテンツに用いる各種メディアの基本的な特徴や、個人情報の保護の必要性についても扱う」と記載されているように、メディアの特徴を整理した上で、必要なか不必要なのかを判断し、必要な情報を複合したプログラムを制作していくことが重要である。検討したことを吟味し、方法を具体化する。この部分を生徒に考えさせていく。

(ア)「時間制限をかけるために何が必要か」について考える

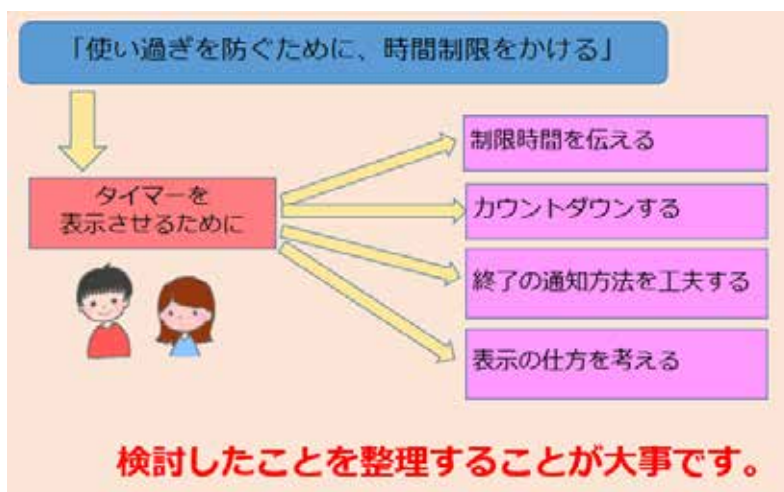


(イ) 具体的な方法に関して、掘り下げて考える

技術の見方・考え方を働かせ、問題解決していく必要がある。



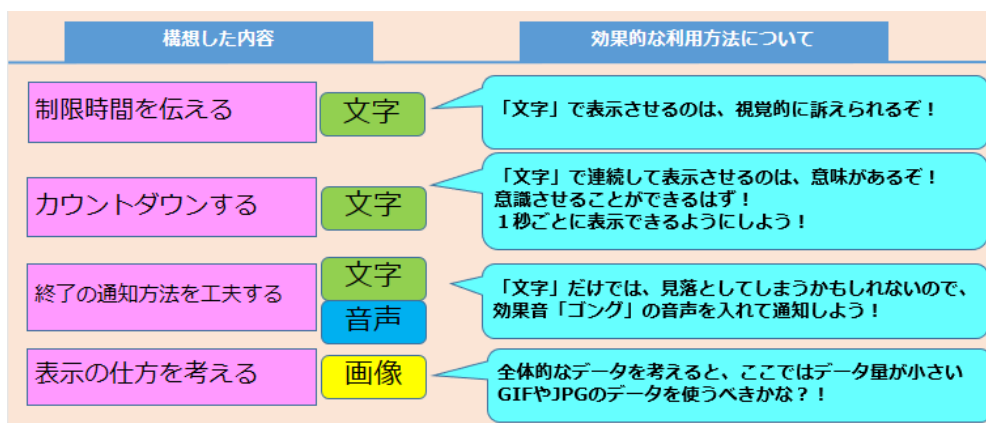
(ウ) 方針が決まったら、具体的にどのような手だてを講ずるか考え、整理する



(エ) 整理した内容に関して、どのようなメディアで表現すると効果的か考える

個人情報などを保護するために、コンテンツを利用する際の約束を明記させたり、ユーザーが見やすく疲れにくい文字のサイズや色を選択したりするなどユニバーサルデザインの視点から利用者の安全や健康に配慮しているかを検討することが大事である。その上で、整理した内容をどのようなメディアを使って表現するのか、また、どのように表現すると効果的になるのかを考えさせる。

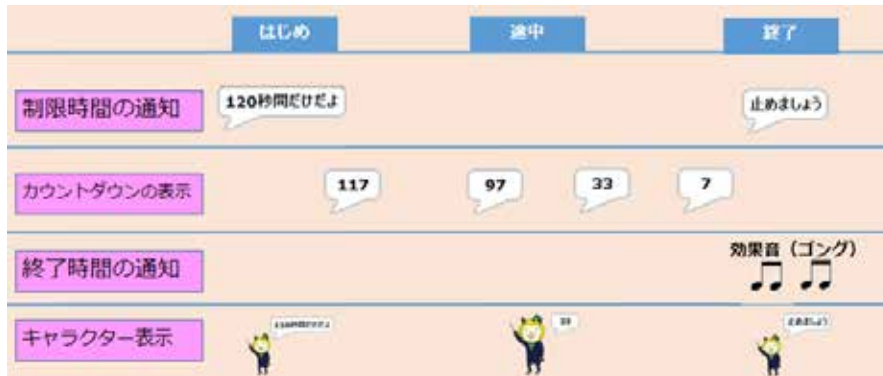
これまで学んだ技術の見方・考え方を働かせ、文字・画像・音声について具体的に検討する。判断に困る場合は、教科書などを使って調べることが考えられる。



「使い過ぎを防ぐために、時間制限をかける」ために、文字・画像・音声のどれをどのように組み合わせて使えばよいか考えさせる場面が必要である。

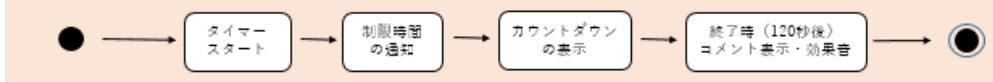
(オ) 情報処理の流れを構想する

【イメージ図】



自分の考えを整理し、よりよい発想を生み出せるよう、アクティビティ図のような統一モデリング言語や製作図等を用いていく。

【設計 (アクティビティ図の活用)】



設計を基にしてタイマーのプログラムを制作していく

安全・適切なプログラムの制作を動作の確認・デバッグ等を行いながら取り組んでいく。他者と協力して作業に取り組み、その成果をお互いに認め合うようにさせたりすることが大切である。

タイマーのプログラム



【プログラム内容】
 スプライト6がクリックされたら、「ルールやマナーを守って楽しみましょう!」を2秒間、「120秒間だけだよ」と2秒間言う。その後、カウントダウンが始まる。タイマーが0秒になったら、ゴングを鳴らして、「止めましょう」と2秒間言って、すべてのプログラムが停止する。
 なお、タイマーは相手方でも同時進行で動くようになっている。
 もし、相手方でタイマーをクリックしたら、自身のタイマーもカウントダウンが始まる。

製作の過程や結果の評価、改善及び修正をする

プログラムの制作の過程や完成したものを自己評価・相互評価しながら、よりよいものとなるよう更なる改善や修正をしていくことが考えられる。

(2) 解決策を打ち出した、その他の内容を含めた構想とプログラムの制作
課題の設定

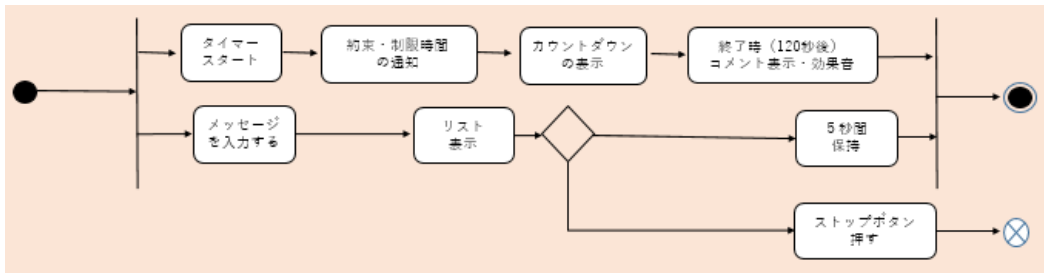


効果的な利用方法の検討

解決策	具体策	使用するメディア	効果的な利用方法の構想
使い過ぎを防ぐために、時間制限をかけよう!	・タイマー表示	文字・画像・音声	文字情報や音声で表示させることは視覚的に訴えることができるので有効。画像はデータ量の少ないもので行う。
始めるときに注意を促すコメントを出そう!	・コメント表示	文字	文字情報として最初に出すことに意味がある。しっかり伝えられるように時間設定も行う。
自分の発言に責任をもたせるために発信記録を表示させるとともに、誤送信防止のための機能を搭載しよう!	・リスト表示 ・プログラム制御	文字・画像	文字情報としてリスト表示させることで、視覚的に訴えられる。記録量を考える必要がある。ストップボタンとしてデータ量の少ない画像を使う。

解決策	機能 及び 表示方法
使い過ぎを防ぐために、時間制限をかけよう!	タイマーをつける。最初に、「120秒だけだよ」と表示させ、120秒経過後、「止めましょう」とメッセージを伝え、効果音を鳴らす。
始めるときに注意を促すコメントを出そう!	最初に「ルールやマナーを守って楽しみましょう!」と表示する。
自分の発言に責任をもたせるために発信記録を表示させるとともに、誤送信防止のための機能を搭載しよう!	リスト表示させる。誤送信防止のために、入力したものがすぐに転送されないように、5秒後転送。誤送信を防ぐためにストップボタンを導入する。発信しても5秒以内であれば、ストップボタンを押せば、送信されない。

情報処理の手順を具現化



(例)

- ・ タイマーをつける。120秒経過後、「止めましょう」とメッセージを伝える。
- ・ タイマーがスタートしたら、「ルールやマナーを守って楽しみましょう！120秒だけだよ」と通知する。
- ・ 自分の発言に責任をもたせるために、発言した内容をリストに表示する。
- ・ 入力したものがすぐに転送されないように、5秒後転送、誤送信を防ぐためにストップボタンを導入する。

プログラムの制作

タイマーをつける



```

スプライト6 がクリックされたとき
  ルールやマナーを守って楽しみましょう！と 2秒言う
  120秒間だけだよと 2秒言う
  timer1 を 120 にする
  ずっと
    もし timer1 = 0 なら
      発言1 の音を鳴らす
      止めましょうと 2秒言う
      すべてを止める
    やなければ
      1秒待つ
      timer1 を 1ずつ変える
    timer1 を送る
    timer1 と言う

timer2 が受け取ったとき
  もし timer2 センサーの値 > 118 なら
    ルールやマナーを守って楽しみましょう！120秒間だけだよと 2秒言う
  timer2 センサーの値 と言う
  もし timer2 センサーの値 = 0 なら
    発言1 の音を鳴らす
    止めましょうと 2秒言う
    すべてを止める
  
```

【プログラム内容】

スプライト6がクリックされたら、「ルールやマナーを守って楽しみましょう！」を2秒間、「120秒間だけだよ」を2秒間言う。その後、カウントダウンが始まる。タイマーが0秒になったら、ゴングを鳴らして、「止めましょう」と2秒間言って、すべてのプログラムが停止する。

なお、タイマーは相手方でも同時進行で動くようになっている。

もし、相手方でタイマーをクリックしたら、自身のタイマーもカウントダウンが始まる。

リストを表示し、5 秒後送信する



【プログラム内容】

5 秒待ってから、変数 (messageA) を A-message に追加する。
その後、送信する。

誤送信を防ぐために、ストップボタンを導入する



【プログラム内容】

Sprite 7 がクリックされたら、すべてのプログラムが終了する。

(3) プログラムの実際 (D-4)



【プログラムの概要説明】

2 台の PC を準備し、1 台目の PC をクライアント A (送受信者)、2 台目の PC をクライアント B (送受信者) とする。(3 台目をサーバ役とすることもできる)

クライアント A 側では Sprite (ひよっとこの面のみやざき犬) をクリックし、メッセージを入力する。入力されたデータが 5 秒後、リスト (A-message) に記録され、送信される。クライアント B 側では、送られてきたデータが吹き出しとして表示される。

Sprite (指し棒をもっているみやざき犬) をクリックすることで 120 秒のカウントダウンが始まる (Sprite をクリックしないと、スタートしないようになっている)。ス

スタート時に「ルールやマナーを守って楽しみましょう！120秒だけだよ」と言う。0秒になると、ゴングが鳴って「止めましょう」のメッセージが吹き出しとして表示され、プログラムが終了する。

また、誤送信を防ぐために次のような機能を備えている。入力されたメッセージは、Enterキーを押した後、リスト(A-message: リスト記録)にのるまでの5秒間でスプライト(神楽STOPスタイルのみやぎき犬)をクリックすると、すべてのプログラムが終了し、データの送信をしないようになっている。





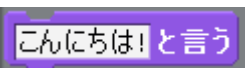







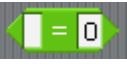



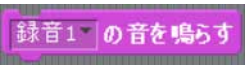

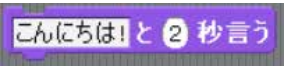



送受信者(クライアントA)のプログラム



【プログラム内容】
 スプライト6がクリックされたら、「ルールやマナーを守って楽しみましょう！」と2秒間、「120秒間だけだよ」と2秒間言う。その後、カウントダウンが始まる。タイマーが0秒になったら、ゴングを鳴らして、「止めましょう」と2秒間言って、すべてのプログラムが停止する。
 なお、タイマーは相手方でも同時進行で動くようになっている。
 もし、相手方でタイマーをクリックしたら、自身のタイマーもカウントダウンが始まる。

ブロック	ブロックのカテゴリ	備考
スプライト6がクリックされたとき	制御	スプライトをファイルから読み込んだ順で番号がかわる。
「ルールやマナーを守って楽しみましょう！」と2秒言う	見た目	「ルールやマナーを守って楽しみましょう！」と入力する。
「120秒間だけだよ」と2秒言う	見た目	「120秒間だけだよ」と入力する。

	変数	新しい変数 timer1 をつくる。その上でブロックを使う。「timer1」を選択し、「120」と入力する。
	制御	
	制御	
	変数	新しく変数をつくる。
	演算	timer1 を埋め込み、「0」を入力する。「timer1 = 0」を意味する。
	音	1 を参照
	見た目	「止めましょう」と入力する。
	制御	すべてのプログラムが停止する。
	制御	
	変数	変数部分を「timer1」にかえ、カウントダウンにするために「-1」と入力する。
	制御	新規で「timer11」とつくる。相手方に送る手段
	見た目	timer1 を埋め込むことで、変数を相手方でも吹き出しとして言うことができる。
	制御	相手方から送られてきたものを受け取る。
	制御	
	演算	timer2 センサーの値 を埋め込むことで、相手方での変数を読み込むことができる。ここでは 119 秒つまり相手がクリックした最初の 1 秒後にコメントが表示されるようになっている。


		<p>相手方の変数を読み込む。</p>
		<p>「ルールやマナーを守って楽しみましょう！120秒だけだよ」と入力する。</p>
		<p>を埋め込むことで、相手方での変数を読み込み、吹き出しとして言うことができる。</p>
 		<p>読み込みたい相手方の変数を指定する。</p>
		
		<p>「タイマーが0秒になったら」</p>
		
		<p>1を参照</p>
		<p>「止めましょう」と入力する。</p>
		

1 音を録音する方法



- ・自分で音声を入力する場合

新しい音 録音 を押す。

サウンドレコーダーの  を押して、音声を入力する。入力後は、「OK」を押す。

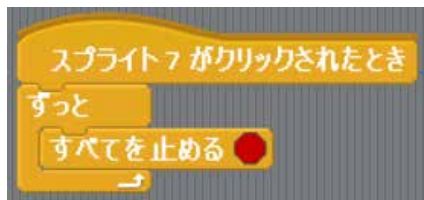
音声入力されると、メーターが動くので確認ができる。



- ・音声データを挿入する場合

読み込み を押す。

音声データが入っている場所を指定し、「OK」を押す。



【プログラム内容】
スプライト7がクリックされたら、
すべてのプログラムが終了する。

ブロック	ブロックのカテゴリ	備考
		
		
		



【プログラム内容】
messageB を受け取ったら、変数 messageB を言う。

ブロック	ブロックのカテゴリ	備考
		を埋め込むことで、相手方での変数を読み込み、吹き出しとして言うことができる。
		読み込みたい相手方の変数を指定する。



【プログラム内容】
スプライト4 がクリックされたら、「答え」を聞いてくる。その「答え」【変数 (messageA) とする】を 5 秒後にリスト (A-message) に追加する。
その後、messageA として送る。
スペースキーが押されたら、リストの中身がすべて削除される。

ブロック	ブロックのカテゴリ	備考
		をはめ込む。「答え」をはめ込むことで、吹き出しは表示されず、入力枠が出てくる。
		変数 を使う。
		5 秒にする。

	変数	A-message というリストをつくると、ブロックが出てくる。入力された変数 messageA を A-message に追加する。
	変数	
	制御	messageA として相手方に送る。
	制御	
	変数	

送受信者（クライアント B）のプログラム



```

スプライト7がクリックされたとき
  ルールとマナーを守って楽しめよう! と 2 秒言う
  120秒間だけだよ と 2 秒言う
  timer2 を 120 にする
  ずっと
    もし timer2 = 0 なら
      鈴音1の音を鳴らす
      止めましょう と 2 秒言う
      すべてを止める
    でなければ
      1 秒待つ
      timer2 を 1 ずつ変える
  timer2 を送る
  timer2 と言う

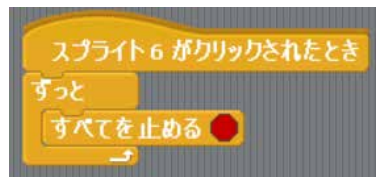
timer11を受け取ったとき
  もし timer1 センサーの値 > 110 なら
    ルールとマナーを守って楽しめよう! 120秒間だけだよ と 2 秒言う
    timer1 センサーの値 と言う
  もし timer1 センサーの値 = 0 なら
    鈴音1の音を鳴らす
    止めましょう と 2 秒言う
    すべてを止める
  
```



【プログラム内容】
 スプライト7がクリックされたら、「ルールとマナーを守って楽しめよう!」と2秒間、「120秒間だけだよ」と2秒間言う。その後、カウントダウンが始まる。タイマーが0秒になったら、ゴングを鳴らして、「止めましょう」と2秒間言って、すべてのプログラムが停止する。
 なお、タイマーは相手方でも同時進行で動くようになっている。
 もし、相手方でタイマーをクリックしたら、自身のタイマーもカウントダウンが始まる。

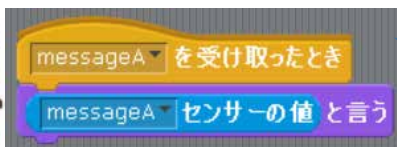
ブロック	ブロックのカテゴリ	備考
	制御	スプライトをファイルから読み込んだ順で番号が変わる。
	見た目	「ルールやマナーを守って楽しみましょう!」と入力する。
	見た目	「120 秒間だけだよ」と入力する。
	変数	新しい変数 timer2 をつくる。その上でブロックを使う。「timer2」を選択し、「120」と入力する。
	制御	
	制御	
	変数	新しく変数をつくる。
	演算	timer2 を埋め込み、「0」を入力する。「timer2 = 0」を意味する。
	音	1 を参照
	見た目	「止めましょう」と入力する。
	制御	すべてのプログラムが停止する。
	制御	
	変数	変数部分を「timer2」にかえ、カウントダウンにするために「- 1」と入力する。
	制御	新規で「timer22」とつくる。相手方に送る手段
	見た目	timer2 を埋め込むことで、変数を相手方でも吹き出しとして言うことができる。

		相手方から送られてきたものを受け取る。
		timer1 センサーの値 を埋め込むことで、相手方での変数を読み込むことができる。ここでは 119 秒つまり相手がクリックした最初の 1 秒後にコメントが表示されるようになっている。
		相手方の変数を読み込む。
		「ルールやマナーを守って楽しめよう! 120秒間だけだよ」と入力する。
		timer1 センサーの値 を埋め込むことで、相手方での変数を読み込み、吹き出しとして言うことができる。
		読み込みたい相手方の変数を指定する。
		「タイマーが 0 秒になったら」
		1 を参照
		「止めましょう」と入力する。



【プログラム内容】
 スプライト6がクリックされたら、すべてのプログラムが終了する。

ブロック	ブロックのカテゴリ	備考
スプライト6がクリックされたとき	制御	
ずっと	制御	
すべてを止める	制御	



【プログラム内容】
 messageAを受け取ったら、変数messageAを言う。

ブロック	ブロックのカテゴリ	備考
messageAを受け取ったとき	制御	
こんにちは!と言う	見た目	messageA センサーの値 を埋め込むことで、相手方での変数を読み込み、吹き出しとして言うことができる。
messageA センサーの値	調べる	読み込みたい相手方の変数を指定する。



【プログラム内容】
 スプライト4がクリックされたら、「答え」を聞いてくる。その「答え」【変数(messageB)とする】を5秒後にリスト(B-message)に追加する。その後、messageBとして送る。スペースキーが押されたら、リストの中身がすべて削除される。

ブロック	ブロックのカテゴリ	備考
		
		 をはめ込む。「答え」をはめ込むことで、吹き出しは表示されず、入力枠が出てくる。
		
		変数  を使う。
		5 秒にする。
		 というリストをつくと、ブロックが出てくる。入力された変数 messageB を B-message に追加する。
		
		messageB として相手方に送る。
		
		

サーバ役のプログラムを作った時のプログラムの紹介



【プログラム内容】

messageA を受け取ったら、スプライト 8（神楽スタイルの 3 匹のみやざき犬）の吹き出しとして表示され、その後、リスト（A-message）に表示される。
messageB を受け取ったら、スプライト 8（神楽スタイルの 3 匹のみやざき犬）の吹き出しとして表示され、その後、リスト（B-message）に表示される。
スプライト 8 がクリックされたら、リストのデータが削除される。

（４）プログラムの実行

《送受信者（クライアント A・クライアント B）の操作手順》

- ① キャラクター（指示棒をもっているみやざき犬）をクリックする。
120 秒のカウントダウンが始まる。
- ② キャラクターをクリックする。
文字の入力を半角英数で行う。（残念ながら、ひらがなの入力はいできない）
- ③ 入力後、Enter キーを押す。メッセージが 5 秒間後、A-message（B-message）に表示される。 転送

《誤送信を防ぐためのプログラム停止方法》

③の動作（Enter キーを押す）後、5 秒以内にキャラクター（神楽 STOP スタイルのみやざき犬）をクリックすると、すべてのプログラムが停止する。

第5章 評価・将来を展望した双方向性のコンテンツを検討しよう

1 学習課題の設定（例）

これまでの学習を基に身に付けた力、技術の見方・考え方を活かして、双方向性のコンテンツに関する将来展望に関して、例を一つ取り上げ利用者と開発者の両方の立場から考えていく活動を行う。

学習課題

双方向性のコンテンツに関する将来展望に関して、例を1つ取り上げ利用者と開発者の両方の立場から考えよう。

2 テーマの設定と実際

テーマ	一人住まいの高齢者が安心して住める住宅環境	
コンセプト	少子高齢化・核家族構成が進む中で、一人住まいの高齢者の孤独死が問題視されている。そこで、安否確認、高齢者の様子がプライバシーの侵害をしない範囲内でできる住宅環境を考える。	
実 際	電源ON・OFF通信管理システムの開発	
	開発者	利用者
仕様内容	<ul style="list-style-type: none">・家の照明・エアコン・冷蔵庫の開閉等の入出力情報を管理し、特定の人物（家族・管理業者）に情報伝達するシステムを開発する。・プライバシーの侵害にならぬよう最低限の情報にとどめる。そのため、利用者が仕様内容を選択できるようにする。	<ul style="list-style-type: none">・家庭内の電源が通信システムとして所定の相手に情報提供ができる。・緊急時のボタン、無反応の継続時間が見られれば、所定の相手にメール配信される通信システムとなる。