

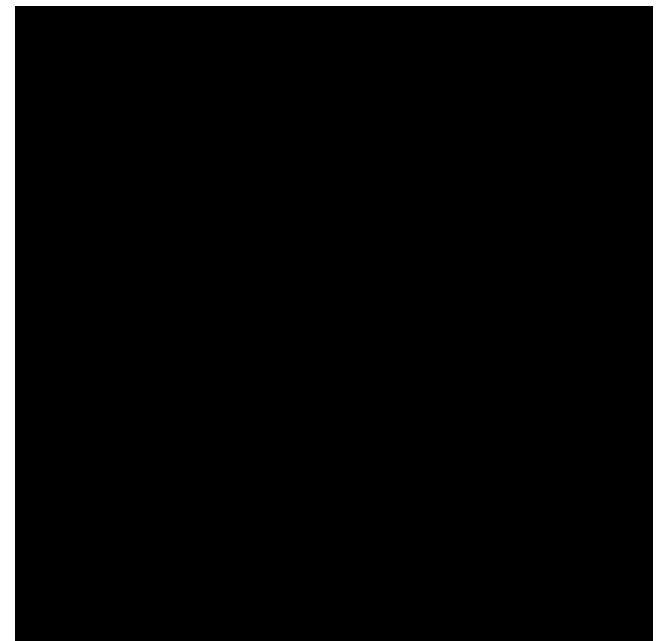
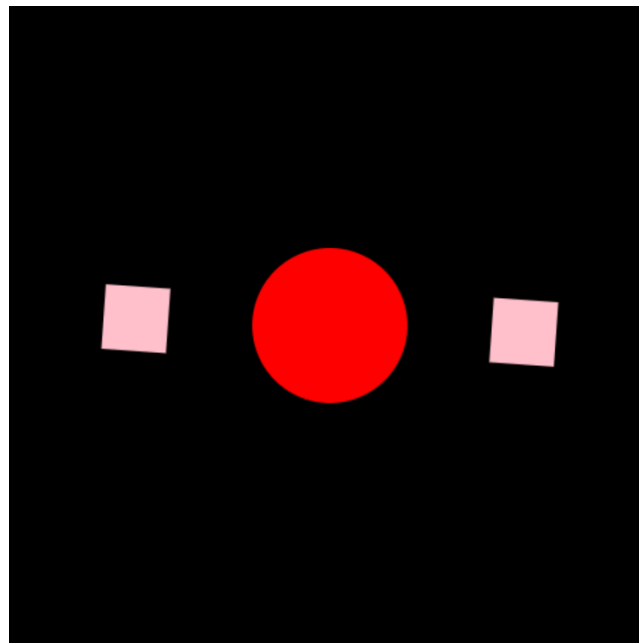
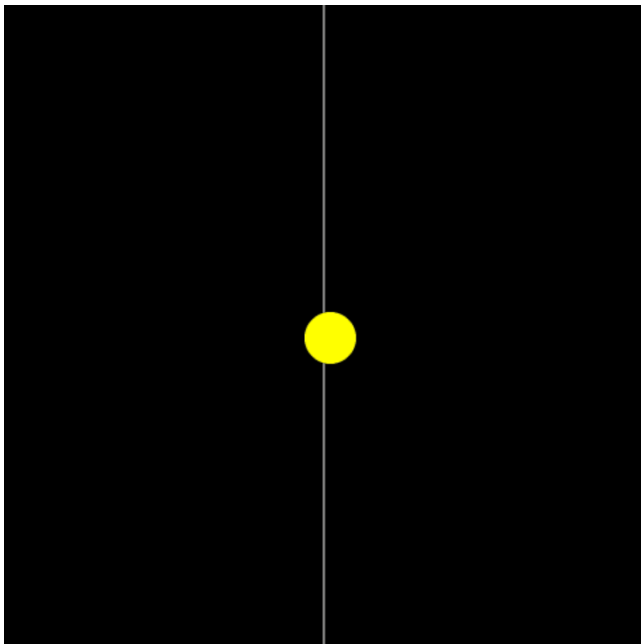


# 社会と情報 プログラミング超入門①

# はじめに

- この授業中の様子(PC上での作業記録や授業後の感想など)を大学での研究にデータを利用する予定です
- 個人が特定されることはありません
- データの利用を拒否しても成績には影響しません
- もしデータを使われたくない場合は教えてください

# アニメーション



# 授業目標

- プログラミングの基礎的なスキルと **プログラミング的思考** を使った問題解決が出来るようになる
  - プログラミング的思考…問題を解決するためには何をどういう順番で組み合わせれば実現できるか？ を論理的に考えられる力

# アルゴリズムの3構造

## 順次(順接)

- ・ Aをしたら次にBをする

## 分岐

- ・ もし〇〇ならAをする(そうでなければBをする)

## 反復(繰り返し、ループ)

- ・ 〇〇である限りずっとAを繰り返す

すべてこの3つで  
表現できる

# 授業で使うシミュレータの紹介

<https://mathinfolec.github.io/canvas/>

- ・「変数定義エリア」と「描画エリア」にコードを書き、キャンバス上にアニメーションを表示させる

変数定義エリア

・ 必要な変数(データを入れる箱)を定義する

描画エリア

・ キャンバス上に図形を描く

# セーブスロットの使い方

- スロット1は授業に合わせて使う
- スロット2-5で自由にコードを書き替える

セーブスロット(使用中:1) 1 2 3 4 5

初期化

変数定義エリア



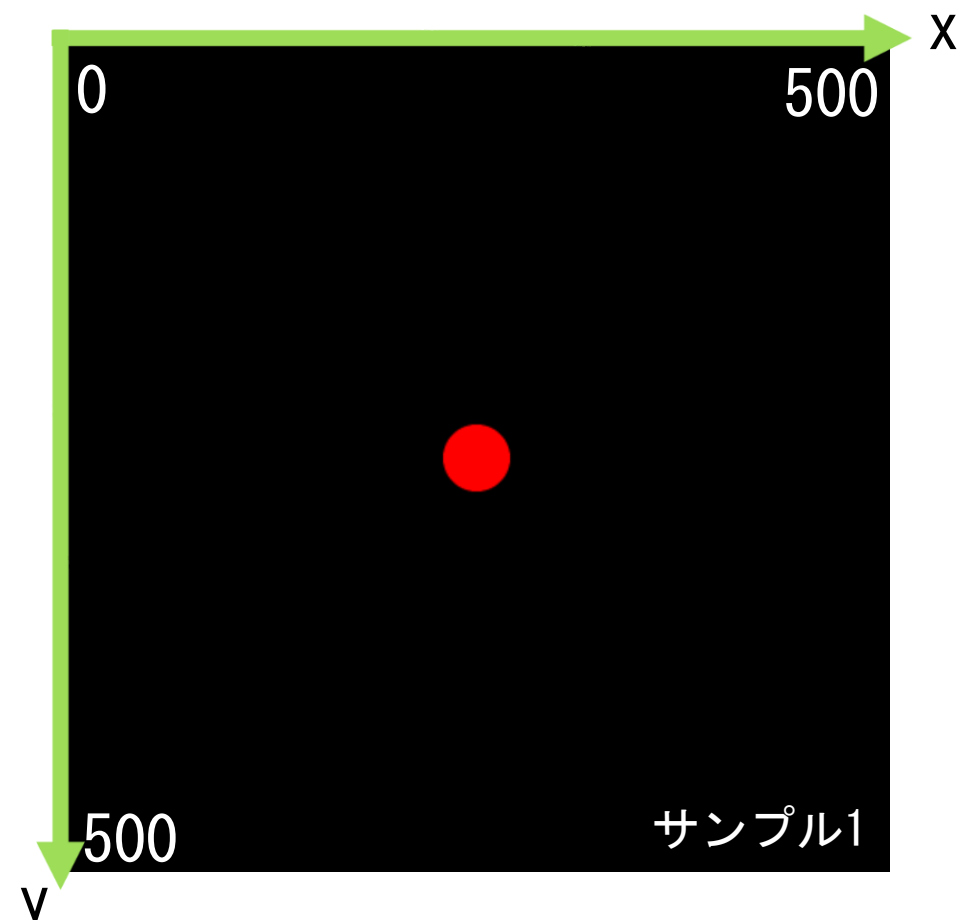
# 円の表示

これ以降に描く図形の色を赤色に設定したのち、(250, 250)を中心に半径20の円を描く  
(この処理を毎フレーム行う)

## 描画エリア

```
setColor("red");  
drawCircle(250, 250, 20);
```

すべて半角で記入する  
大文字・小文字は区別する  
セミコロンを末尾に必ず付ける





# 変数を使った表示

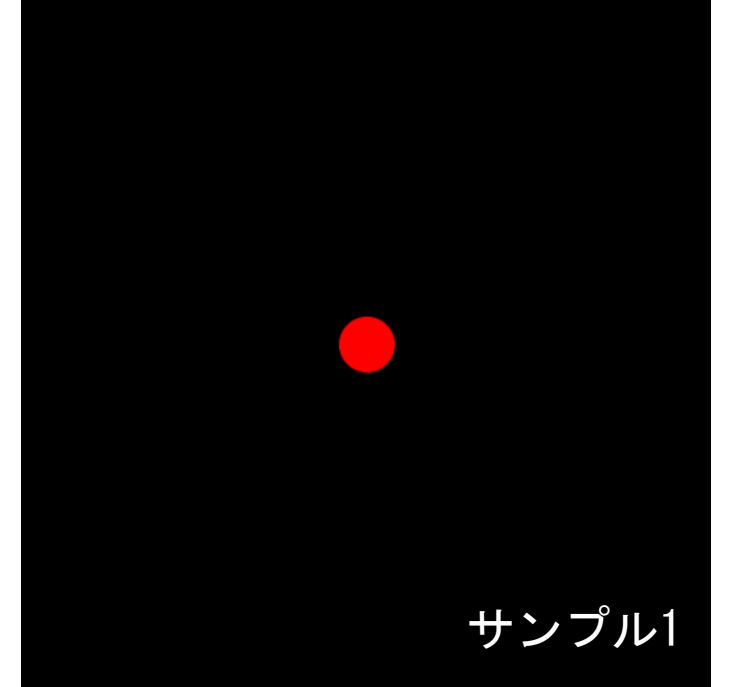
## 変数定義エリア

```
let x=250;  
let y=250;  
let r=20;
```

## 描画エリア

```
setColor("red");  
drawCircle(x, y, r);
```

- 変数は「データを入れる箱」のようなもの



サンプル1

# 演習

※問題番号に対応する  
セーブスロットを使う

## 変数定義エリア

```
let x=250;  
let y=250;  
let r=20;
```

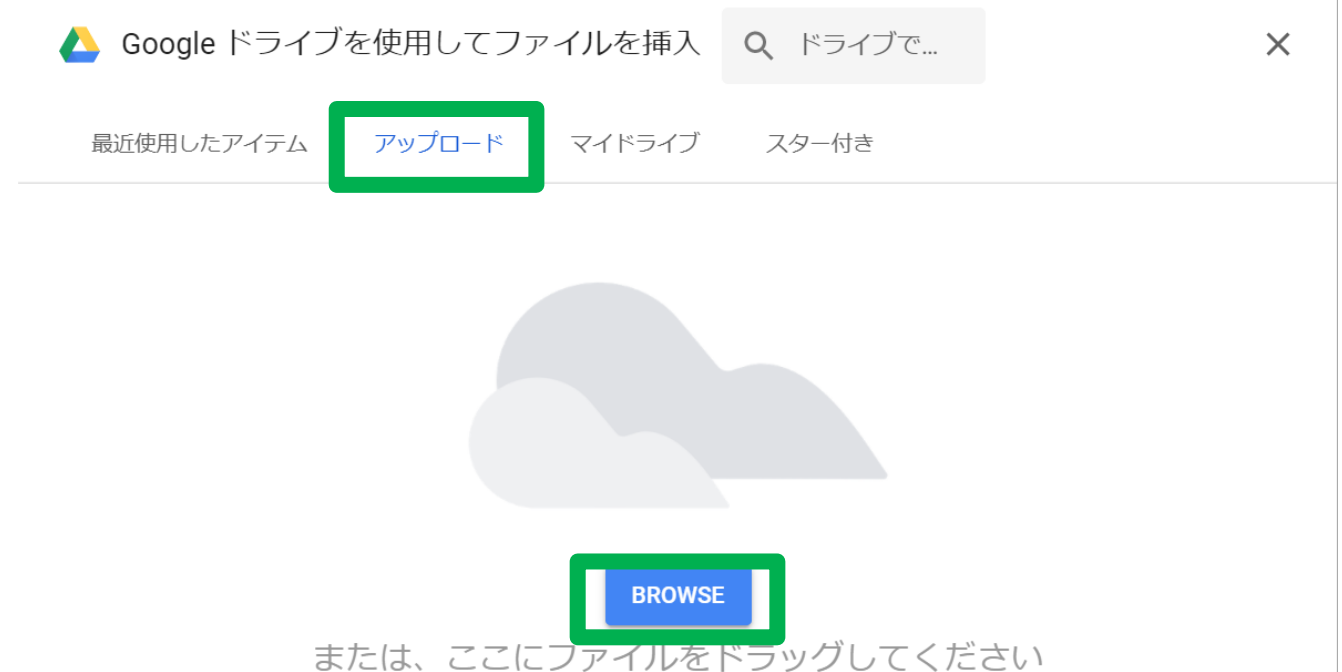
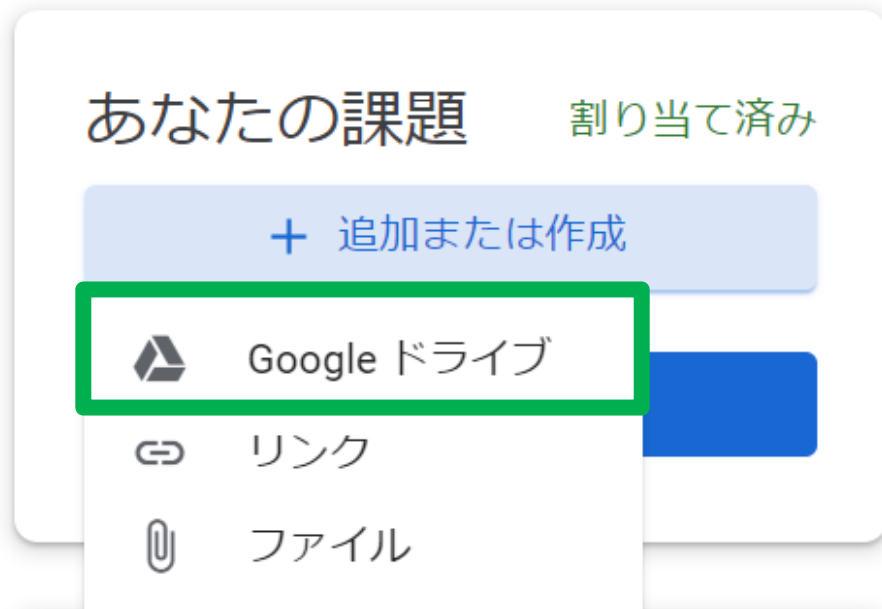
## 描画エリア

```
setColor("red");  
drawCircle(x, y, r);
```

1. 左のコードを写し、変数の使い方と描画方法を理解する
2. 変数の値などを自由に変えてみる
3. 中心の空いた円を描く
4. 【応用】パズルピースを描く（正方形の描き方は機能集を確認すること）

# 課題提出方法

- ログファイルを出力してClassroomに提出する



ファイルは「ダウンロード」フォルダにある