明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

プログラミング演習2
クラスと継承の補足

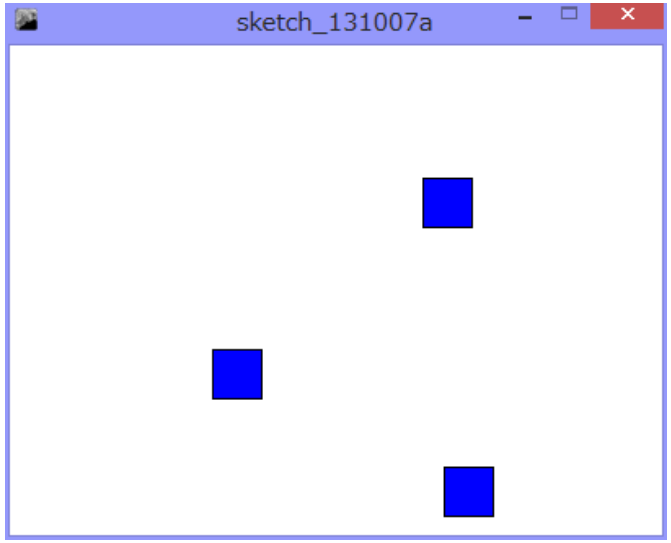
中村，小松，菊池

端で跳ね返る正方形を描く

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q1) 400x300のウインドウ内で，任意の場所から毎フレームx方向，y方向に任意の速度で移動する3つの青色の四角形を描画し，右端・左端・上端・下端に来ると跳ね返るようにするには？

- 考え方
  - 右端・左端・上端・下端で衝突する時の条件を整理
  - 衝突した時の速度を反転させる
    - speedX = -speedX;
    - speedY = -speedY;



Ballを改良しSquareを作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Square{
   int x;
   int y;
   int speedX;
   int speedY;

   Square(){
      init();
   }
   void init(){
      x = (int)random(width);
      y = (int)random(height);
      speedX = (int)random(5);
      speedY = (int)random(5);
   }
   void display(){
      fill( 0, 0, 255 );
      rect( x-15, y-15, 30, 30 );
   }
```

```
   void move(){
      x = x + speedX;
      y = y + speedY;
      if ( x+15 > width ) {
         x = width - 15;
         speedX = -speedX;
      }
      if( x - 15 < 0 ){
         x = 15;
         speedX = -speedX;
      }
      if( y + 15 > height ){
         y = height - 15;
         speedY = -speedY;
      }
      if( y - 15 < 0 ){
          y = 15;
          speedY = -speedY;
      }
   }
}
```

Ballを改良しSquareを作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Square{
   int x;
   int y;
   int speedX;
   int speedY;

   Square(){
      this.init();
   }
   void init(){
      this.x = (int)random(width);
      this.y = (int)random(height);
      this.speedX = (int)random(5);
      this.speedY = (int)random(5);
   }
   void display(){
      fill( 0, 0, 255 );
      rect( this.x-15, this.y-15, 30, 30 );
   }
```

```
void move(){
   this.x = this.x + this.speedX;
   this.y = this.y + this.speedY;
   if ( this.x+15 > width ) {
      this.x = width - 15;
      this.speedX = -this.speedX;
   }
   if( this.x - 15 < 0 ){
      this.x = 15;
      this.speedX = -this.speedX;
   }
   if( this.y + 15 > height ){
      this.y = height - 15;
      this.speedY = -this.speedY;
   }
   if( this.y - 15 < 0 ){
      this.y = 15;
      this.speedY = -this.speedY;
```

クラス内では省略しているが，基本的にthis（自分の）を使って
this.インスタンス変数名，this.インスタンスメソッド名という意味

Squareクラスを使うと

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
Square fukuchi;
Square nakamura;
Square hashimoto;
void setup() {
   size( 400, 300 );
   fukuchi = new Square();
   nakamura = new Square();
   hashimoto = new Square();
}

void draw() {
   background(255);
   fukuchi.move();
   nakamura.move();
   hashimoto.move();
   fukuchi.display();
   nakamura.display();
   hashimoto.display();
}
```

3つの四角形と3つの丸

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

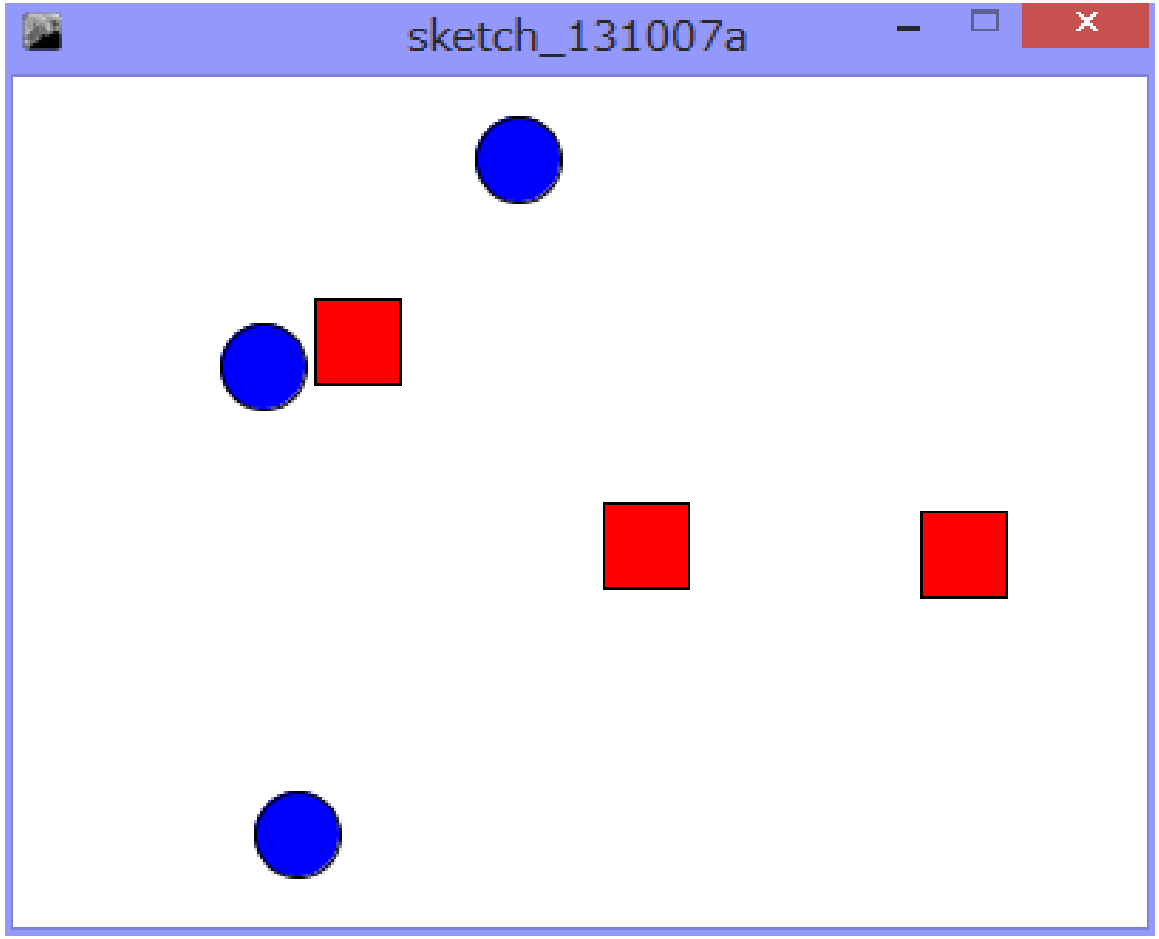
(Q2) 400x300のウインドウ内で，任意の場所から毎フレームx方向，y方向に任意の速度で移動する赤色の3つの丸と，青色の3つの四角形を描画し，右端・左端・上端・下端に来ると跳ね返るようにするには？

- 考え方
  - 右端・左端・上端・下端で衝突する時の条件を整理
  - 衝突した時の速度を反転させる
    - speedX = -speedX;
    - speedY = -speedY;

3つの四角形と3つの丸

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 前に作ったBallクラスと，今回作ったSquareクラスを組み合わせる



Ballクラス

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Ball{
   int x;
   int y;
   int speedX;
   int speedY;

   Ball(){
      init();
   }
   void init(){
      x = (int)random(width);
      y = (int)random(height);
      speedX = (int)random(5);
      speedY = (int)random(5);
   }
   void display(){
      fill( 255, 0, 0 );
      ellipse( x, y, 30, 30 );
   }
```

```
   void move(){
      x = x + speedX;
      y = y + speedY;
      if ( x+15 > width ) {
         x = width - 15;
         speedX = -speedX;
      }
      if( x - 15 < 0 ){
         x = 15;
         speedX = -speedX;
      }
      if( y + 15 > height ){
         y = height - 15;
         speedY = -speedY;
      }
      if( y - 15 < 0 ){
          y = 15;
          speedY = -speedY;
      }
   }
}
```

Ballを改良しSquareを作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Square{
   int x;
   int y;
   int speedX;
   int speedY;

   Square(){
      init();
   }
   void init(){
      x = (int)random(width);
      y = (int)random(height);
      speedX = (int)random(5);
      speedY = (int)random(5);
   }
   void display(){
      fill( 0, 0, 255 );
      rect( x-15, y-15, 30, 30 );
   }
```

```
   void move(){
      x = x + speedX;
      y = y + speedY;
      if ( x+15 > width ) {
         x = width - 15;
         speedX = -speedX;
      }
      if( x - 15 < 0 ){
         x = 15;
         speedX = -speedX;
      }
      if( y + 15 > height ){
         y = height - 15;
         speedY = -speedY;
      }
      if( y - 15 < 0 ){
          y = 15;
          speedY = -speedY;
      }
   }
}
```

3つの円と3つの四角形を移動

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
Ball miyashita;
Ball komatsu;
Ball kikuchi;
Square fukuchi;
Square nakamura;
Squre hashimoto;
void setup() {
  size( 400, 300 );
  fill( 255, 0, 0 );
  miyashita = new Ball();
  komatsu = new Ball();
  kikuchi = new Ball();
  fukuchi = new Square();
  nakamura = new Square();
  hashimoto = new Square();
}
```

```
void draw() {
  background(255);
  // 移動
  miyashita.move();
  komatsu.move();
  kikuchi.move();
  fukuchi.move();
  nakamura.move();
  hashimoto.move();
  // 描画
  miyashita.display();
  komatsu.display();
  kikuchi.display();
  fukuchi.display();
  nakamura.display();
  hashimoto.display();
}
```

ここで・・・

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Ballクラスと，Squareクラスはほとんど一緒
- 違いはクラス名とコンストラクタ，そしてdisplayのインスタンスメソッドのみ

無駄じゃね？

ほとんど一緒

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

	Ball	Square	
インスタンス変数	x, y speedX, speedY	x, y speedX, speedY	一致
クラス名	Ball	Square	違う
コンストラクタ	Ball()	Square()	内部は一致
void init()			一致
void move()			一致
void display()	fill(0, 0, 255); ellipse(x, y, 30, 30);	fill(0, 0, 255); rect(x-15, y-15, 30, 30);	違う

一緒の部分をまとめた
スーパークラス（親クラス）を作る

スーパーなObjectクラス

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Object{
   int x;
   int y;
   int speedX;
   int speedY;

   Object(){
      init();
   }
   void init(){
      x = (int)random(width);
      y = (int)random(height);
      speedX = (int)random(5);
      speedY = (int)random(5);
   }
```

display() は内容が違うので
削除してしまう

```
   void move(){
      x = x + speedX;
      y = y + speedY;
      if ( x+15 > width ) {
         x = width - 15;
         speedX = -speedX;
      }
      if( x - 15 < 0 ){
         x = 15;
         speedX = -speedX;
      }
      if( y + 15 > height ){
         y = height - 15;
         speedY = -speedY;
      }
      if( y - 15 < 0 ){
          y = 15;
          speedY = -speedY;
      }
   }
}
```

一緒の部分をまとめる

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Ball は Object の変数（x, y, speedX, speedY）や機能（移動や初期化）をもち，独自の表示に関する機能（メソッド）をもつクラス
- Square はObject の変数（x, y, speedX, speedY）や機能（移動や初期化）をもち，独自の表示に関する機能（メソッド）をもつクラス

- インスタンス変数や，インスタンスメソッドを引き継ぐことを継承と呼ぶ！

Objectクラスを使うと

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
class Ball extends Object {
    void display(){
        fill( 255, 0, 0 );
        ellipse( x, y, 30, 30 );
    }
}

class Square extends Object {
    void display(){
        fill( 0, 0, 255 );
        rect( x-15, y-15, 30, 30 );
    }
}
```

```
Ball miyashita;
Ball komatsu;
Ball kikuchi;
Square fukuchi;
Square nakamura;
Squre hashimoto;
void setup() {
  size( 400, 300 );
  fill( 255, 0, 0 );
  miyashita = new Ball();
  komatsu = new Ball();
  kikuchi = new Ball();
  fukuchi = new Square();
  nakamura = new Square();
  hashimoto = new Square();
}
    :
```

BallクラスとSquareクラスが劇的に短く！

なぜ動くの？

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 継承すると，親の力をすべて引き継ぐ！
- 継承の方法は extends とやるだけ！

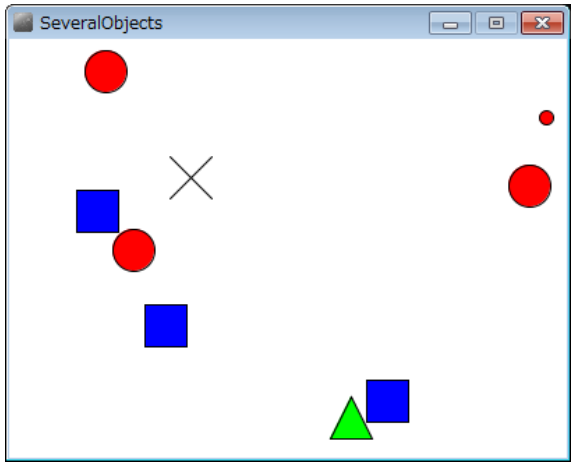
```
class クラス名 extends 親クラス名 {

}
```

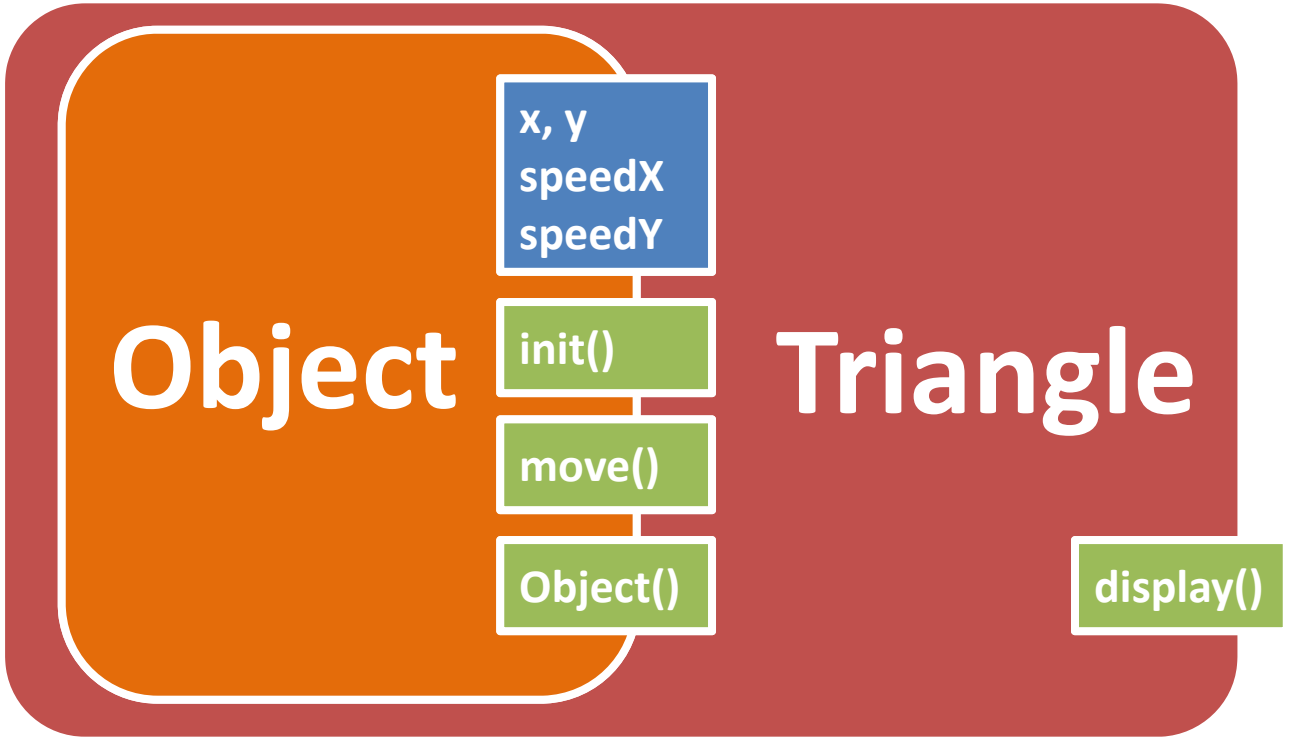
- 継承により親の能力，値はすべて引き継ぎます

じゃあ

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Object クラスを継承して「三角形」を描画するクラスを作るには？（Triangle）
- Object クラスを継承して「×」を描画するクラスを作るには？（Cross）





三角形と×のクラス

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 動く三角形のクラスと動く×のクラス

```
class Triangle extends Object {
    void display(){
        fill( 0, 255, 0 );
        triangle( x, y-15, x-15, y+15, x+15, y+15 );
    }
}

class Cross extends Object {
    void display(){
        fill( 0, 255, 0 );
        line( x-15, y-15, x+15, y+15 );
        line( x-15, y+15, x+15, y-15 );
    }
}
```

三角形は跳ね返らないように

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q3) 先述のObjectを継承したTriangleクラスを改良し，三角形は跳ね返らず右端→左端，左端→右端，上端→下端，下端→上端と移動するようにせよ

- 考え方
  - Object の move メソッドを Triangle クラス内でオーバーライドして，三角形専用のメソッドを作成する

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

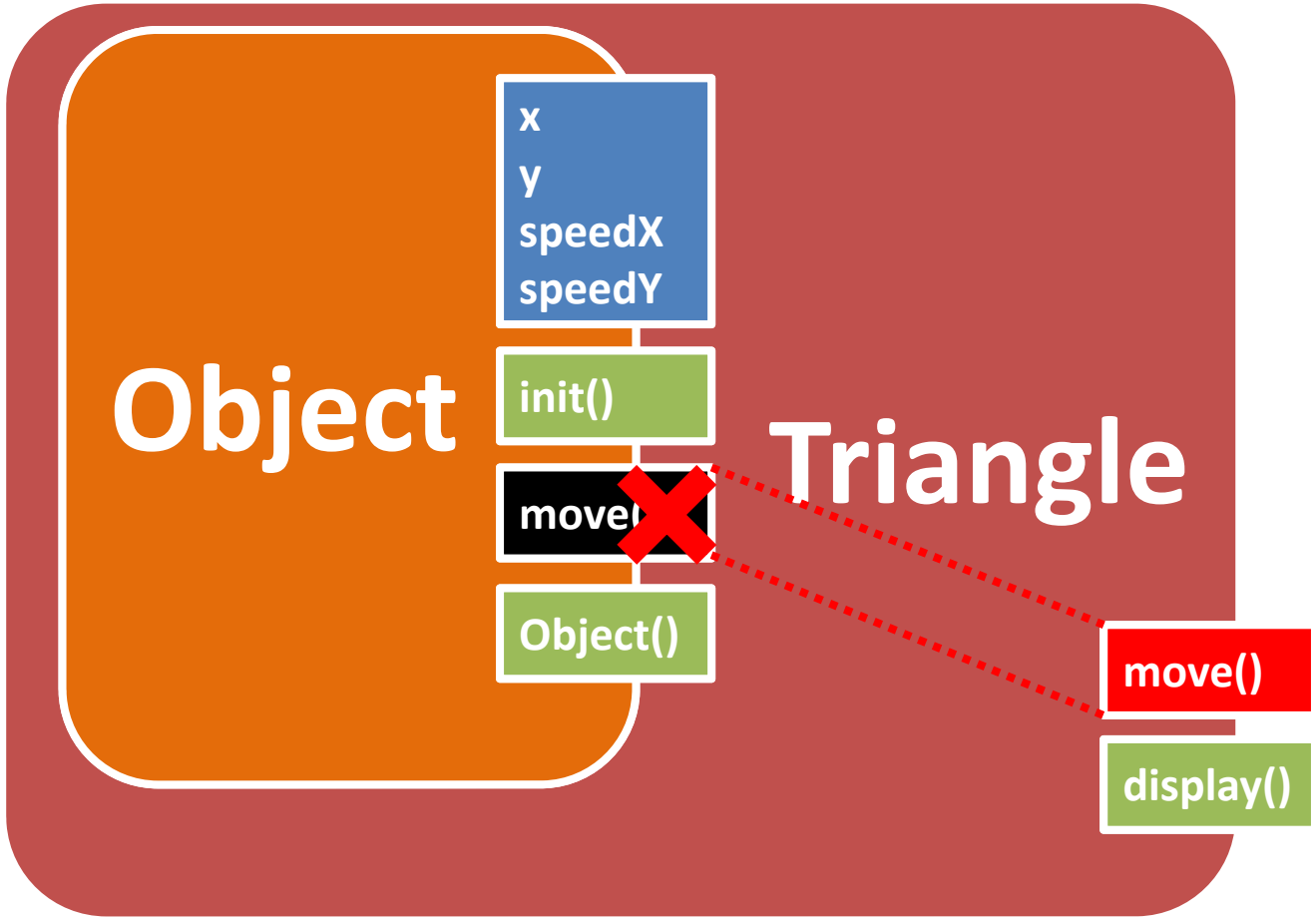

```
class Triangle extends Object {
    void display(){
        fill( 0, 255, 0 );
        triangle( x, y-15, x-15, y+15, x+15, y+15 );
    }
    void move(){
        x = x + speedX;
        y = y + speedY;
        if( x > width ){
            x = x - width;
        } else if( x < 0 ){
            x = width + x;
        }
        if( y > height ){
            y = y - height;
        }
        if( x < 0 ){
            y = width + y;
        }
    }
}
```

move メソッドをオーバーライドして親の move メソッドが呼ばれないようにする

move をオーバーライド

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Triangle の move で，Object の move を上書きしてしまう！



全体の挙動を変更

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q4) Objectクラスのmoveメソッドを変更し，上端及び下端は跳ね返るが，左端と右端では逆側から現れるようにせよ

- 考え方
  - Object の move メソッドのみ変更
  - y 座標の条件で跳ね返り
  - x 座標の条件で逆側から現れるようにする

Object クラスのみ変更

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Object{
    int x;
    int y;
    int speedX;
    int speedY;

    Object(){
        init();
    }
    void init(){
        x = (int)random(width);
        y = (int)random(height);
        speedX = (int)random(5);
        speedY = (int)random(5);
    }
```

```
    void move(){
        x = x + speedX;
        y = y + speedY;
        if( x > width ){
            x = x - width;
        } else if( x < 0 ){
            x = width + x;
        }

        if( y + 15 > height ){
            y = height - 15;
            speedY = -speedY;
        } else if( y - 15 < 0 ){
             y = 15;
             speedY = -speedY;
        }
    }
}
```

惑星と衛星の様なオブジェクト

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q5) 400x300のウインドウ内で，任意の場所x，yから任意の速度で移動する3つの赤色の円を描画し，右端・左端・上端・下端に来ると跳ね返るようにする．また，赤色の円には円の中心から30の距離があるところに1つの衛星があり，10度ずつ円の周りを回転するようにせよ

- 考え方
  - 円の中心（x, y）から衛星の方向の角度（0～360度）を theta とすると，衛星の座標は

（x+30*cos(radians(theta)), y+30*sin(radians(theta)))

惑星と衛星の様なオブジェクト

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

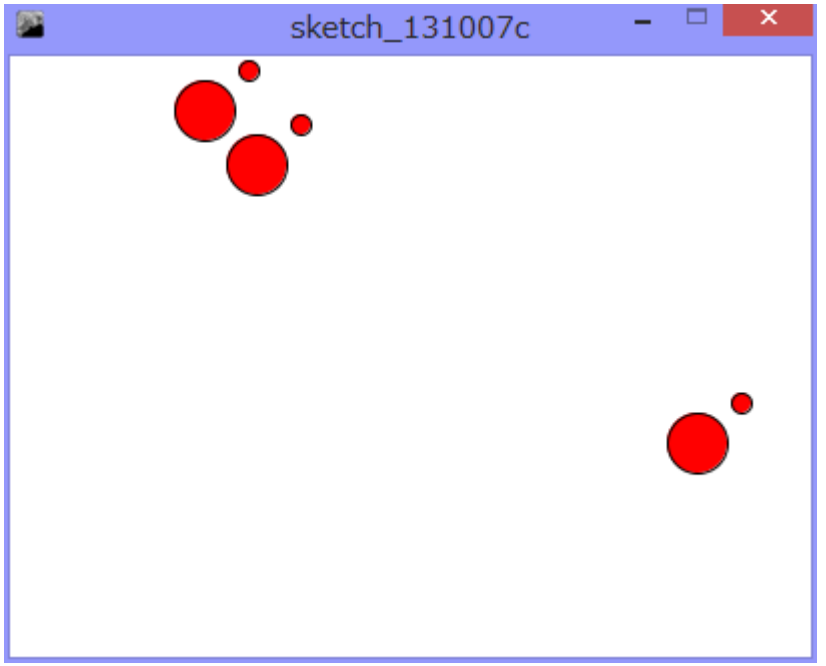
- Objectクラスを継承して，変数を追加する

```
class PlanetSatellite extends Object {
    int theta;
    void display(){
        fill( 255, 0, 0 );
        ellipse( x, y, 30, 30 );
        theta = theta + 10;
        int rx = (int)(x+30*sin(radians(theta)));
        int ry = (int)(y+30*cos(radians(theta)));
        ellipse( rx, ry, 10, 10 );
    }
}
```

惑星と衛星の様なオブジェクト

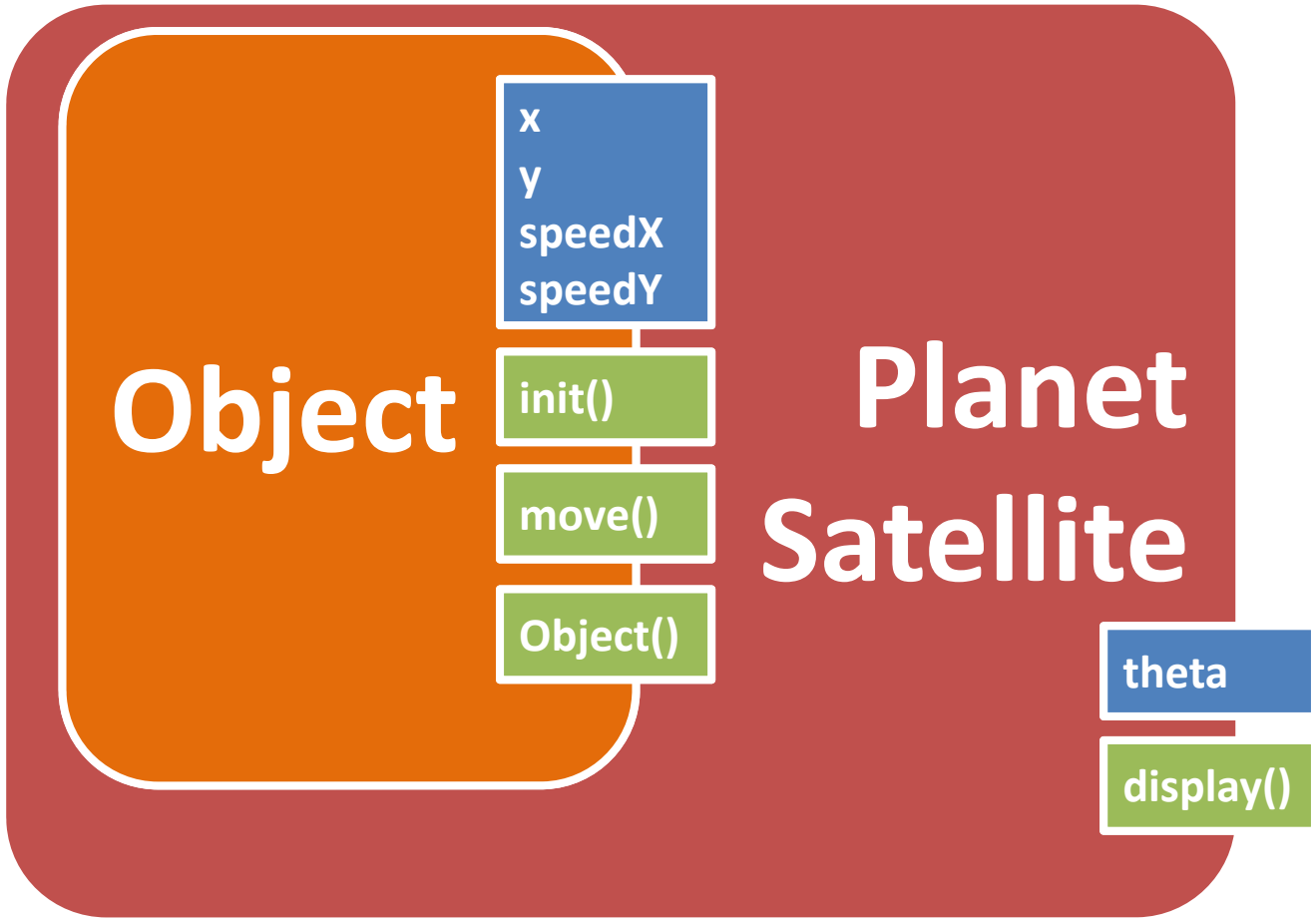
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
PlanetSatellite miyashita;
PlanetSatellite komatsu;
PlanetSatellite kikuchi;
void setup() {
    size( 400, 300 );
    miyashita = new PlanetSatellite();
    komatsu = new PlanetSatellite();
    kikuchi = new PlanetSatellite();
}

void draw() {
    background(255);
    miyashita.move();
    komatsu.move();
    kikuchi.move();
    miyashita.display();
    komatsu.display();
    kikuchi.display();
}
```



継承すると・・・

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室



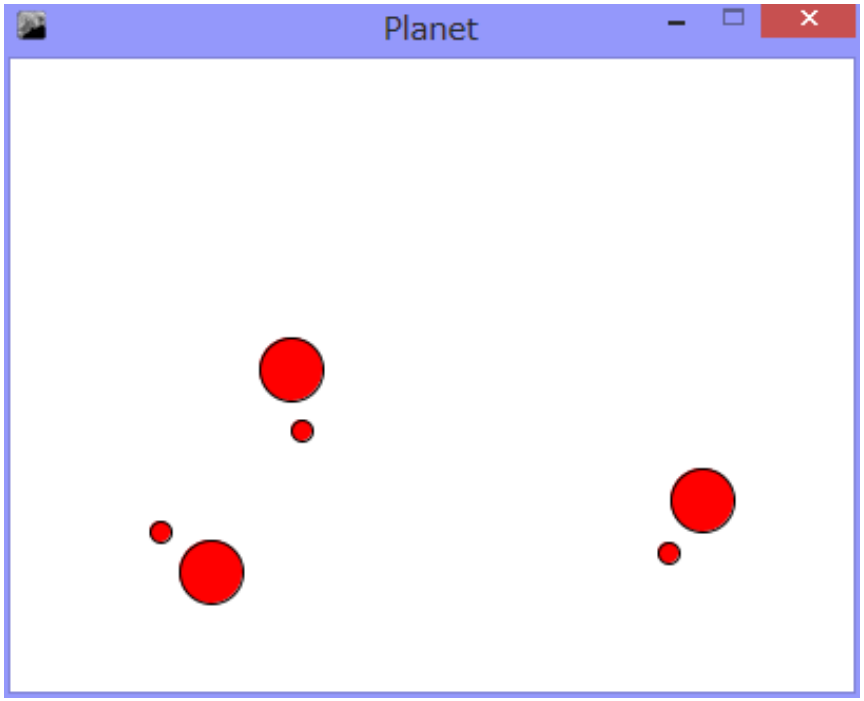
- Object を PlanetSatellite として継承し，theta という変数と，display() というメソッドを追加

惑星と衛星の様なオブジェクト

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q5) Q4を改良し，衛星の開始角を0～360度の任意の場所にしたい．どうするか？

- 考え方
  - void init() というインスタンスメソッドを追加し，改良したらOK？



やってみる

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
class PlanetSatellite extends Object {
    int theta;
    void init(){
        theta  = (int)random(360);
    }
    void display(){
        fill( 255, 0, 0 );
        ellipse( x, y, 30, 30 );
        theta = theta + 10;
        int rx = (int)(x+30*sin(radians(theta)));
        int ry = (int)(y+30*cos(radians(theta)));
        ellipse( rx, ry, 10, 10 );
    }
}
```

init() はコンストラクタで呼ばれる
Objectクラス参照

うまく動作しない

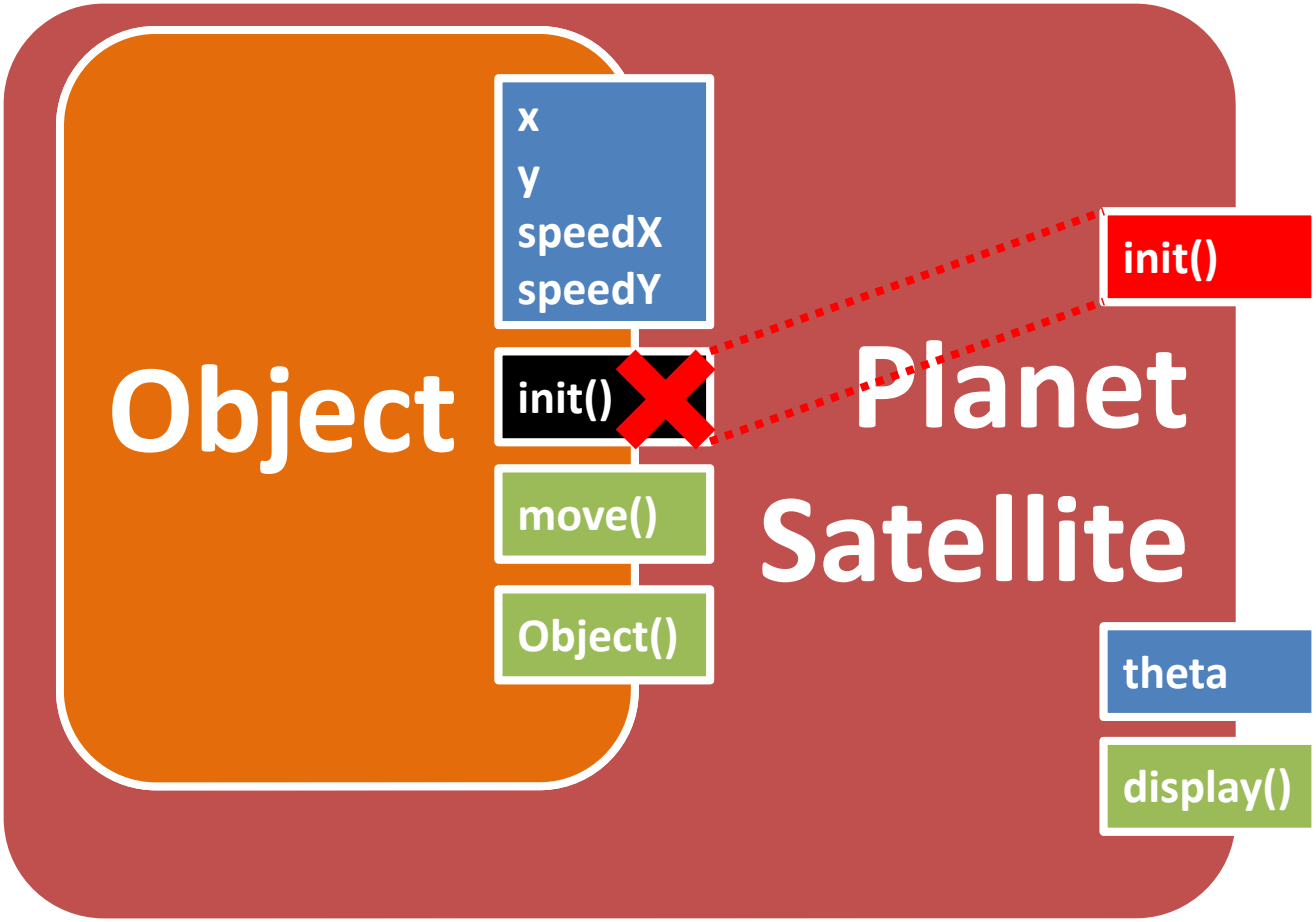
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 画面の左上から動かない．何故か？

上書きしてしまったから

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Object の init() を，PlanetSatellite の init() でオーバーライドている（Objectのinit()が呼び出されない）




```
class PlanetSatellite extends Object {
    int theta;
    void init(){
        theta  = (int)random(360);
        x = (int)random(width);
        y = (int)random(height);
        speedX = (int)random(5);
        speedY = (int)random(5);
    }
    void display(){
        fill( 255, 0, 0 );
        ellipse( x, y, 30, 30 );
        theta = theta + 10;
        int rx = (int)(x+30*sin(radians(theta)));
        int ry = (int)(y+30*cos(radians(theta)));
        ellipse( rx, ry, 10, 10 );
    }
}
```

Object の init() にあるのを
そのままコピペする

動くけど，なんだか
無駄が増えている・・・

Objectのinit()も使いたい

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- クラスの中で，super と書くと，継承元の親を呼び出すことができる！

```
class PlanetSatellite extends Object {
    int theta;
    void init(){
        theta  = (int)random(360);
        super.init();
    }
    void display(){
        fill( 255, 0, 0 );
        ellipse( x, y, 30, 30 );
        theta = theta + 10;
        int rx = (int)(x+30*sin(radians(theta)));
        int ry = (int)(y+30*cos(radians(theta)));
        ellipse( rx, ry, 10, 10 );
    }
}
```

super . メソッド名

衛星を3つにする

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q6) Q4の PlanetSatellite クラスを継承し，衛星の数を3つにする PlanetSatellite3 クラスを作りたい．どうするか？

- 考え方
  - インスタンス変数を追加する
  - init メソッドと，display メソッドをオーバーライドする

さて

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 色々なオブジェクトがあると，配列でまとめることができない．どうする？

- それはまた次のお話