Espèce : Renard

Nombre de population : 6.822.66

| Moyenne morts par jour :

Age yen: 2.

Cau rincipal mortalitée

SIMULATEUR SECOSYSTEM

Infos du cyc

Nombre de morts : (Nombre de naissances : (

PAR LEGRAND ALEXIA

PAR LEGRAND ALEXIA AVEC JÉRÉMY ET ZONGH-Y

1/1/2000

SOMMAIRE



Classe principale



Hiérarchie du programme

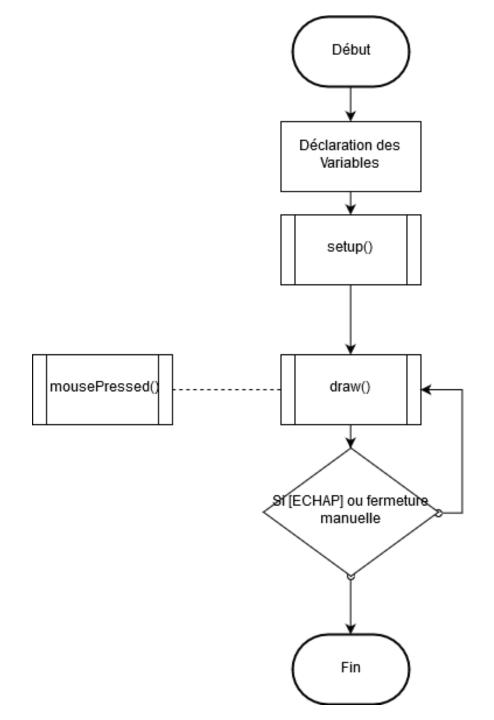


Les classes especes, animaux et plante



Maps_gestion et Maps_Reader

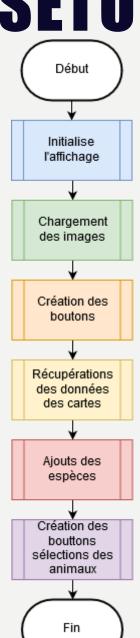
CLASSE PRINCIPALE

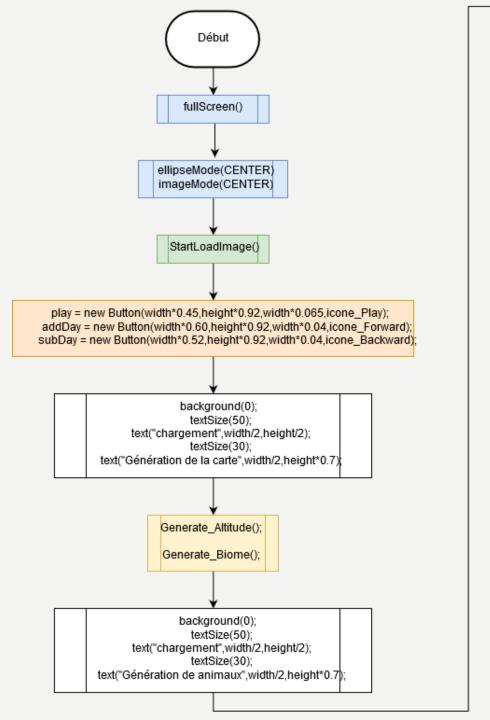


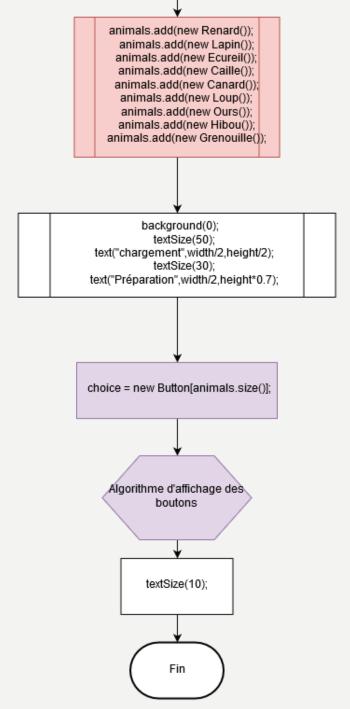
VARIABLES

Туре	Nom	Contenu
ArrayList <animal></animal>	animals	New ArrayList <animal>()</animal>
Int	actualChoice	-1
Int	day	I
Boolean	dayPass	false
Int	Xmouse	-I
Int	Ymouse	-I
Button	play	N/A
Button[]	choice	N/A
Button	addDay	N/A
Button	subDay	N/A

SETUP



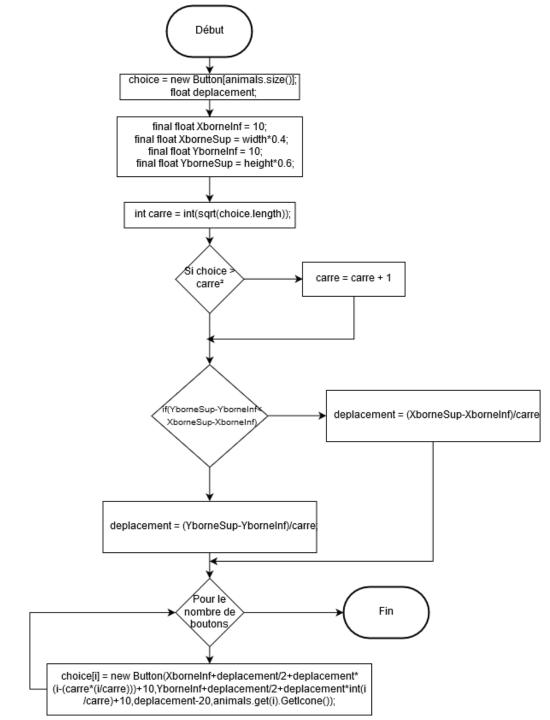




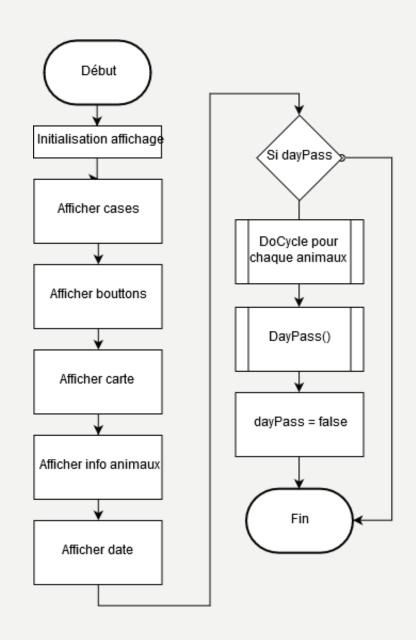
ALGORITHME D'AFFICHAGE DES BOUTTONS

Carre

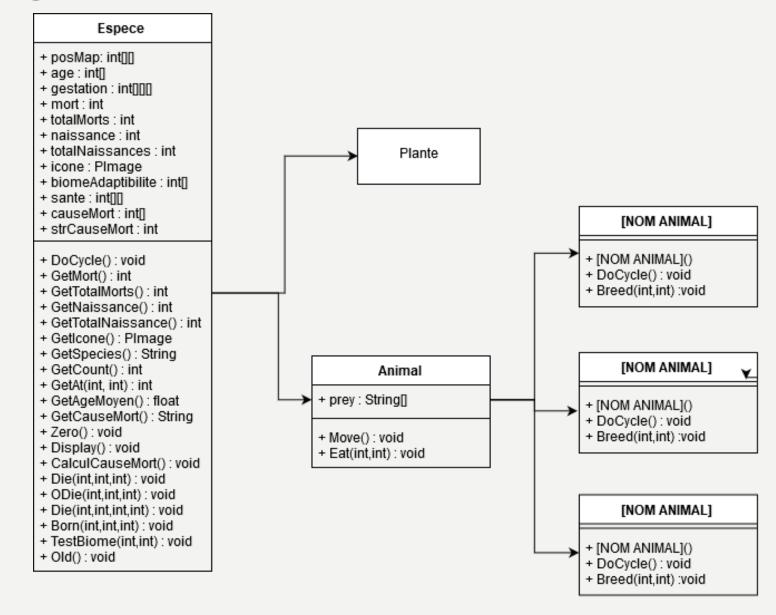




DRAW



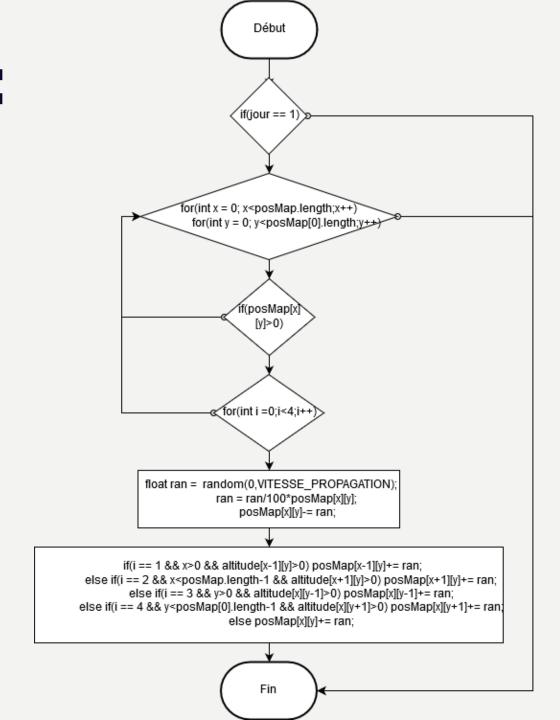
HIERARCHIE



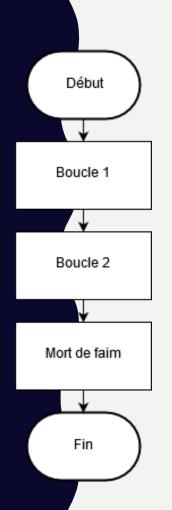
CLASSE ESPÈCE : OLD()

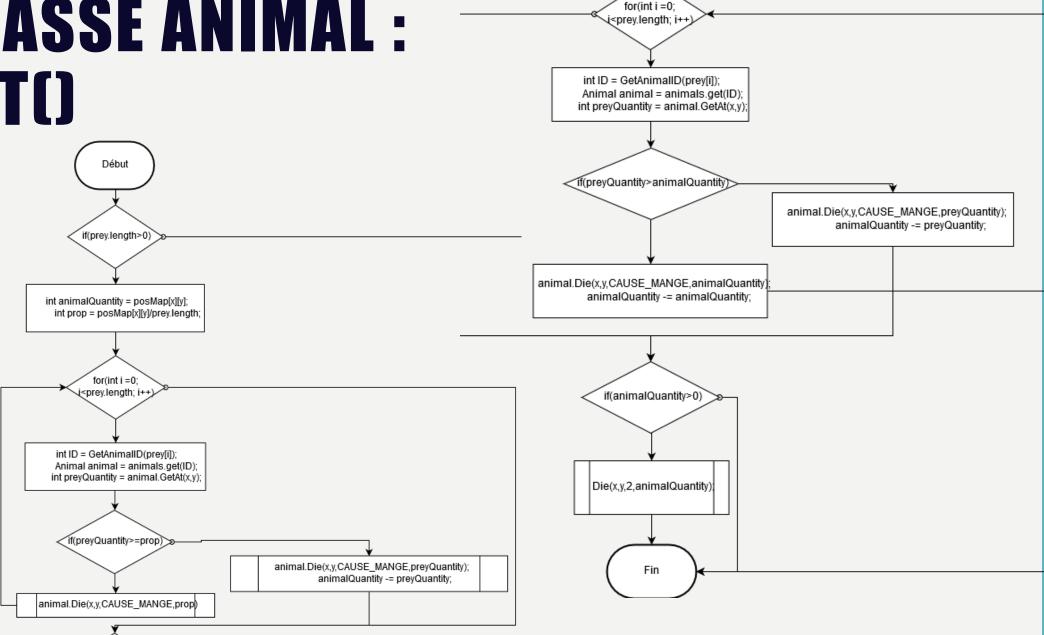
```
Début
                      ArrayList<int[]> presence = new ArrayList<int[]>();
                                for(int i = 0; i<posMap.length;i++)
                              for(int y = 0; y < posMap[0].length;y++){
                          if(posMap[i][y] > 0) presence.add(new int[]{i,y});
                                 if(jour == 31 && mois == 12)
                     for(int i = age[DUREE_VIE-1];i>0;i--) age[i] = age[i-1];
                            for(int i=0;i<age[DUREE_VIE-1];i++)
                            int ran = int(random(0,presence.size()));
              while(posMap [presence.get(ran)[0]][presence.get(ran)[1]]== 0) ran =
                              int(random(0,presence.size()));
                ODie(presence.get(ran)[0],presence.get(ran)[1],DUREE_VIE-1);
             //println("Death at " + presence.get(ran)[0] + ", " + presence.get(ran)[1]);
                        for(int i=VIEILLESSE_MIN;i<DUREE_VIE-2;i++)
                                   float rand = random(0,0.05*i);
                                 for(int j=0;j<age[i]*rand/1000;j++){
                              int ran = int(random(0,presence.size()));
while(posMap [presence.get(ran)[0]][presence.get(ran)[1]]== 0) ran = int(random(0,presence.size())
                          ODie(presence.get(ran)[0],presence.get(ran)[1],i);
               //println("Death at " + presence.get(ran)[0] + ", " + presence.get(ran)[1]);
                                            Fin
```

CLASSE ANIMAL: MOVE()



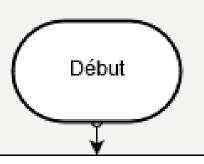
CLASSE ANIMAL: EAT()





MAPS READER: Début **TESTCOLOR()** boolean[][] map = new boolean[image.width][image.height]; image.loadPixels(); for(int i = 0; i< image.width*image.height;i++); int y_map = i/image.width; int x_map = i-(y_map*image.width); if(image.pixels[i] == c) s map[x_map][y_map] = false; Text map[x_map][y_map] = true; return map;

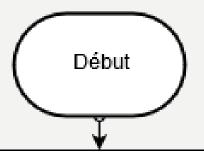
MAPS_READER: MAP_READ()



PImage map = GetMap(Imap); int[][] finalMap = new int[map.width][map.height];

```
for(int i = 0; i < tColor.length; i++)
boolean[]] test = TestColor(map,tColor[i]);
       for(int j = 0; j < test.length; j++)
     for(int k = 0; k < test[0].length; k++)
   if(test[j][k]) finalMap[j][k] = returnValue[i];
            return finalMap;
```

MAPS_READER: RESIZEMAP()

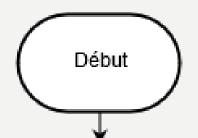


boolean[[[] nouvelle = new boolean[(int) (width*0.4)][(int)(height*0.37)];

for(int i = 0; i < nouvelle.length; i++)
for(int j = 0; j < nouvelle[0].length; j++)
nouvelle[i][j] = original[int(i*(original.length/(float)nouvelle.length))][int(j*
(original[0].length/(float)nouvelle[0].length))];

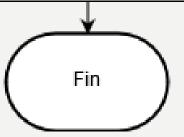
return nouvelle;

MAPS_GESTION: BIOME ET ALTITUDE



altitude = Map_Read(lien carte,liste des couleurs,liste des valeurs);
int[[[] newAltitude = new int [(int) (width*0.4)][(int)(height*0.37)];

 $for(int \ i = 0; \ i < newAltitude.length \ ; \ i++) \\ for(int \ j = 0; \ j < newAltitude[0].length \ ; \ j++) \\ newAltitude[i][j] = altitude[int(i*(altitude.length/(float)newAltitude.length))][int(j*(altitude[0].length/(float)newAltitude[0].length))]; \\ (altitude[0].length/(float)newAltitude;)$



GETBIOMENAME(ID)

