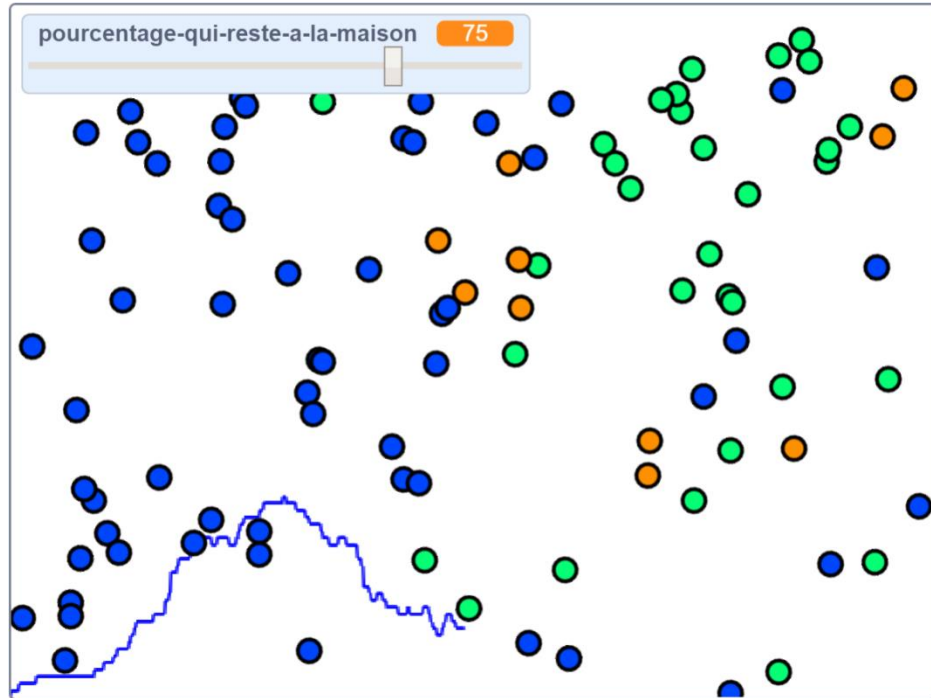


# Simuler la propagation d'une épidémie



## Évaluer l'impact du confinement






Version  
Pierre Huguet

d'après  
Mitchell Resnick






# ANALYSE

On considère ici qu'une épidémie démarre avec un premier malade qui peut contaminer les personnes avec qui il est en contact. Au bout d'un moment, on guérit ...

-  personne malade contaminante (dont le patient 0)
-  personne en bonne santé, non immunisée
-  personne guérie immunisée

# ANALYSE

On considère ici qu'une épidémie démarre avec un premier malade qui peut contaminer les personnes avec qui il est en contact. Au bout d'un moment, on guérit ...

-  personne malade contaminante (dont le patient 0)
-  personne en bonne santé, non immunisée
-  personne guérie immunisée

durée-de-contamination




vitesseDeplacement


nombre-de-malades


pourcentage-qui-reste-a-la-maison


# ANALYSE


On considère ici qu'une épidémie démarre avec un premier malade qui peut contaminer les personnes avec qui il est en contact. Au bout d'un moment, on guérit ...

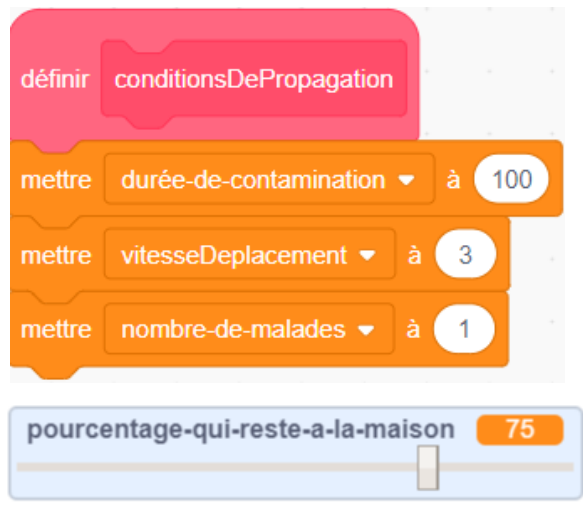
-  personne malade contaminante (dont le patient 0)
-  personne en bonne santé, non immunisée
-  personne guérie immunisée

 durée-de-contamination

 vitesseDeplacement

 nombre-de-malades

 pourcentage-qui-reste-a-la-maison



defining propagation conditions script:

- define conditionsDePropagation
- set duration-de-contamination to 100
- set vitesseDeplacement to 3
- set nombre-de-malades to 1

percentage-qui-reste-a-la-maison slider: 75

# ANALYSE : POUR CHAQUE PERSONNE

Pour chaque personne on a 3 variables locales

santé

bonneSanté, malade ou guérie

temps-depuis-contamination

Le temps depuis qu'elle est malade (contaminée)

reste-a-la-maison-?

Oui ou non selon qu'elle reste confinée  
(si non elle bouge)

# ANALYSE : POUR CHAQUE PERSONNE

Pour chaque personne on a 3 variables locales

santé

bonneSanté, malade ou guérie

temps-depuis-contamination

Le temps depuis qu'elle est malade (contaminée)

reste-a-la-maison-?

Oui ou non selon qu'elle reste confinée  
(si non elle bouge)

les personnes malades  contaminent

les personnes en bonne santé  touchées

Lorsque le  temps-depuis-contamination atteint la  durée-de-contamination

la personnes est guérie  et immunisée.

# ANALYSE : POUR CHAQUE PERSONNE

Pour chaque personne on a 3 variables locales

santé

bonneSanté, malade ou guérie

temps-depuis-contamination

Le temps depuis qu'elle est malade (contaminée)

reste-a-la-maison-?

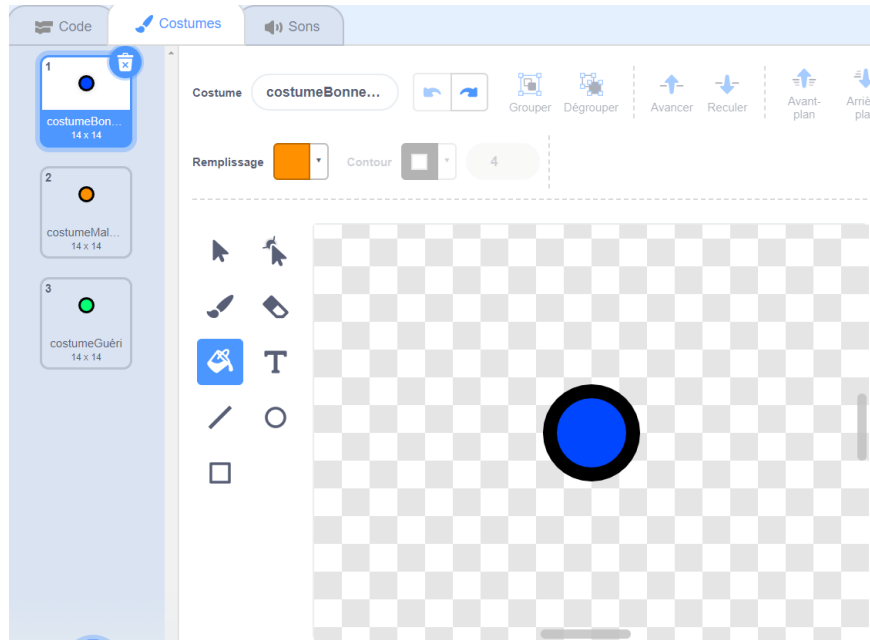
Oui ou non selon qu'elle reste confinée  
(si non elle bouge)

les personnes malades ● contaminent

les personnes en bonne santé ● touchées

# LE SPRITE « PERSONNE » ET SES COSTUMES

Créez "costumeBonneSanté" bien centré de 3 ou 4 pixels  
puis faites 2 copies : "costumeMalade" et "costumeGuéri"  
avec 3 couleurs





# AU DÉMARRAGE : INITIALISATION ET PATIENT 0

The image shows a Scratch script for initializing variables and setting up Patient 0. The script is organized into two main sections: one for global conditions and one for the Patient 0 character.

**Global Conditions:**

- When the green flag is clicked, define a variable named `conditionsDePropagation`.
- Set `conditionsDePropagation` to `Patient0`.
- Set `durée-de-contamination` to 100.
- Set `vitesseDeplacement` to 3.
- Set `nombre-de-malades` to 1.

**Patient 0 Setup:**

- Define a character named `Patient0`.
- Move Patient 0 to x: 0, y: 0.
- Set `reste-a-la-maison-?` to `non`.
- Set `santé` to `malade`.
- Set `temps-depuis-contamination` to 0.
- Switch to the `costumeMalade` costume.

# PUIS CRÉATION DE 100 CLONES EN BONNE SANTÉ

The image shows three columns of Scratch code blocks. The first column starts with a 'when green flag is clicked' event, followed by defining a 'conditionsDePropagation' variable, a 'Patient0' variable, and a loop that repeats 100 times to create clones of 'moi-même'. The second column defines 'Patient0' with 'reste-a-la-maison?' set to 'non', 'santé' set to 'malade', and 'temps-depuis-contamination' set to 0, and then switches to the 'costumeMalade' costume. The third column starts with a 'when I start as a clone' event, moves to a random position, sets 'santé' to 'bonneSanté', and switches to the 'costumeBonneSanté' costume. It then uses an 'if' block to check if a random number between 0 and 99 is less than 'pourcentage-qui-reste-a-la-maison'. If yes, it sets 'reste-a-la-maison?' to 'oui'; otherwise, it sets it to 'non'.

```
quand le drapeau vert est cliqué
conditionsDePropagation
Patient0
répéter 100 fois
  créer un clone de moi-même

définir conditionsDePropagation
mettre durée-de-contamination à 100
mettre vitesseDeplacement à 3
mettre nombre-de-malades à 1

définir Patient0
aller à x: 0 y: 0
mettre reste-a-la-maison-? à non
mettre santé à malade
mettre temps-depuis-contamination à 0
basculer sur le costume costumeMalade

quand je commence comme un clone
aller à position aléatoire
mettre santé à bonneSanté
basculer sur le costume costumeBonneSanté
si nombre aléatoire entre 0 et 99 < pourcentage-qui-reste-a-la-maison alors
  mettre reste-a-la-maison-? à oui
sinon
  mettre reste-a-la-maison-? à non
```

# ET DÉMARRAGE DE L'HORLOGE : 480 COUPS OU HEURES

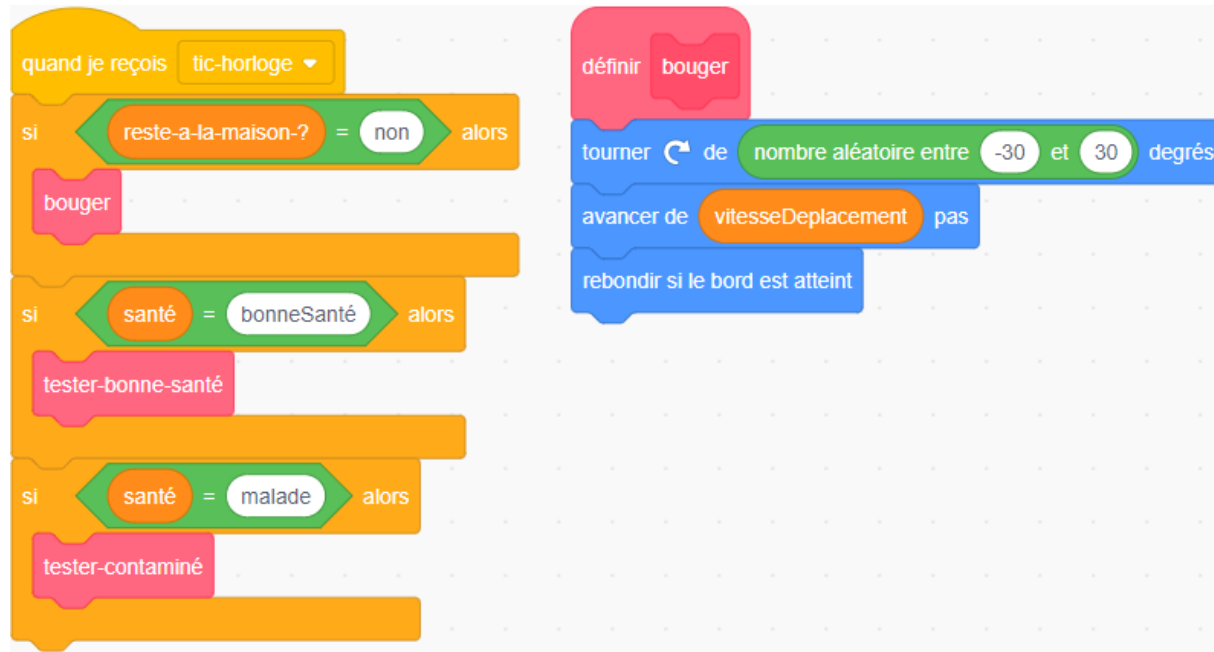


# PUIS CRÉATION DE 100 CLONES EN BONNE SANTÉ

```
quand est cliqué
conditionsDePropagation
Patient0
répéter 100 fois
  créer un clone de moi-même
  répéter 480 fois
    envoyer à tous tic-horloge
```

```
quand je reçois tic-horloge
si reste-a-la-maison-? = non alors
  bouger
  définir bouger
si santé = bonneSanté alors
  tester-bonne-santé
  définir tester-bonne-santé
si santé = malade alors
  tester-contaminé
  définir tester-contaminé
```

# A CHAQUE COUP D'HORLOGE : INTERACTIONS



# A CHAQUE COUP D'HORLOGE : INTERACTIONS

The image displays two columns of Scratch code blocks. The left column contains logic for handling clock ticks and health status. The right column contains logic for movement and infection.

**Left Column:**

- when I receive: tic-horloge
- if: reste-a-la-maison? = non then: bouger
- if: santé = bonneSanté then: tester-bonne-santé
- if: santé = malade then: tester-contaminé

**Right Column:**

- define: bouger
  - turn: de nombre aléatoire entre -30 et 30 degrés
  - move: vitesseDeplacement pas
  - bounce: rebondir si le bord est atteint
- define: tester-bonne-santé
  - if: couleur touchée? then:
    - switch: basculer sur le costume costumeMalade
    - set: santé à malade
    - add: 1 to nombre-de-malades
    - set: temps-depuis-contamination à 0

# A CHAQUE COUP D'HORLOGE : INTERACTIONS

The image displays a Scratch script for a character's interactions. The script is organized into three main sections:

- When I receive 'tic-horloge':** A 'when I receive' block triggers a 'if' block. The condition is 'reste-a-la-maison?' = 'non'. If true, the character 'bouger' (moves).
- Health Logic:** Two 'if' blocks check the character's 'santé' (health).
  - If 'santé' = 'bonneSanté', the character 'tester-bonne-santé' (tests good health).
  - If 'santé' = 'malade', the character 'tester-contaminé' (tests contaminated).
- Movement and Collision:** A 'define' block sets 'bouger' to:
  - 'tourner' (turn) by a 'nombre aléatoire entre -30 et 30' (random number between -30 and 30) degrees.
  - 'avancer de vitesseDeplacement pas' (move forward by 'vitesseDeplacement' steps).
  - 'rebondir si le bord est atteint' (bounce if the edge is reached).

A 'define' block for 'tester-bonne-santé' is also shown, containing a 'if' block for 'couleur touchée?' (color touched?). This block includes a 'basculer' (toggle) block for 'volumeMalade' and a 'mettre à jour' (update) block for 'malades'.

A color picker dialog is open over the 'couleur touchée?' block, showing a color selection interface with 'Couleur 9', 'Saturation 100', and 'Luminosité 100'. An arrow points from the color picker to a small character icon on the right side of the slide.

# A CHAQUE COUP D'HORLOGE : INTERACTIONS

The image displays three columns of Scratch code blocks, representing the logic for an epidemic simulation. The code is organized into three main sections:

- Left Column (Main Loop):**
  - Starts with a "when I receive" block for the "tic-horloge" event.
  - Follows with an "if" block: "if 'reste-a-la-maison?' is 'non', then 'bouger' (move).
  - Then another "if" block: "if 'santé' is 'bonneSanté', then 'tester-bonne-santé' (check health).
  - Finally, a third "if" block: "if 'santé' is 'malade', then 'tester-contaminé' (check for contamination).
- Middle Column (Movement and Health Checks):**
  - Defines the "bouger" block: "turn" by a random angle between -30 and 30 degrees, "move" by "vitesseDeplacement" steps, and "bounce if edge reached".
  - Defines the "tester-bonne-santé" block: "if" the "orange" color is touched, then "switch costume" to "costumeMalade", "set 'santé' to 'malade'", "add 1 to 'nombre-de-malades'", and "set 'temps-depuis-contamination' to 0".
- Right Column (Contamination Logic):**
  - Defines the "tester-contaminé" block: "add 1 to 'temps-depuis-contamination'", "if" "temps-depuis-contamination" is greater than "durée-de-contamination", then "add -1 to 'nombre-de-malades'", "switch costume" to "costumeGuéri", and "set 'santé' to 'guéri'".



# A CHAQUE COUP D'HORLOGE : INTERACTIONS

The image displays a Scratch script for an epidemic simulation, organized into three main sections. The first section, on the left, is a 'when green flag clicked' event handler. It contains three conditional blocks: 1) 'if (reste-a-la-maison? = non) then (bouger)'; 2) 'if (santé = bonneSanté) then (tester-bonne-santé)'; 3) 'if (santé = malade) then (tester-contaminé)'. The second section, in the middle, is a 'when clicked' event handler. It starts with 'define (bouger)'. The main logic is 'if (couleur touchée?) then (turn (direction) degrees) (move (vitesse) steps) (bounce if off edge)'. A color selection dialog is open over this section, showing 'Couleur 9' (red), 'Saturation 100', and 'Luminosité 100'. Below the dialog, the script continues with 'define (tester-bouger)'. The third section, on the right, is a 'when clicked' event handler for 'tester-contaminé'. It includes: 'set (temps-depuis-contamination) to 1', 'if (temps-depuis-contamination > durée-de-contamination) then (set (nombre-de-malades) to (nombre-de-malades) - 1) (switch costume to costumeGuéri) (set (santé) to guéri)'. The background is a light gray grid.

# POUR DESSINER : SPRITE GRAPHIQUE

The image shows two Scratch code snippets on a grid background. The left snippet is triggered by a 'when clicked' event and contains five blocks: 'hide', 'erase everything', 'lift pen', 'go to x: -240 y: -180', and 'pen down'. The right snippet is triggered by a 'when I receive' event with the message 'tic-horloge' and contains one block: 'go to x: abscisse x + 1 y: -180 + nombre-de-malades \* 3.6'. The 'nombre-de-malades' variable is highlighted in orange in the original image.

Left snippet:

- quand est cliqué
- cache
- effacer tout
- relever le stylo
- aller à x: -240 y: -180
- stylo en position d'écriture

Right snippet:

- quand je reçois tic-horloge
- aller à x:  $\text{abscisse } x + 1$  y:  $-180 + \text{nombre-de-malades} * 3.6$