

# OPENCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !



# OPENCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## BUT DU JEU

- MacGyver doit sortir du labyrinthe
- Il doit ramasser 3 objets pour fabriquer une seringue
- Il doit endormir le gardien avant de pouvoir sortir

## CONTRAINTES

- Utiliser le module 2D Pygame
- Un seul parcours
- La fenêtre doit être carrée et contenir 15 sprites
- MacGyver se déplace avec les flèches directionnelles
- Les objets sont placés aléatoirement
- Il y a un compteur listant les objets
- Il y a une barre d'éléments affichant les objets ramassés
- Le labyrinthe doit être dans un fichier séparé
- Un fichier par classe
- Une fonction main
- Utilisation de la PEP8

# OPENCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## Le Labyrinthe

- Lecture de la structure du plateau depuis le fichier
- 0 = Chemin vide
- G = Gardien
- 1 = Murs
- Positionnement du gardien
- Positionnement aléatoire des objets
- Un seul chemin possible pour arriver au gardien

## Le Code

```
# -----  
def lab_struct(self):  
    # -- Init structure of the labyrinth  
    with open('game/labyrinth') as maze:  
        maze = ''.join(maze.read().splitlines())  
        # -- Init empty sprites positions  
        self.extend([divmod(idx, 15) for idx, value in enumerate(maze)  
                    | if value == '0'])  
        # -- Init guardian position  
        self.guardpos = divmod(maze.find('G'), 15)  
        # -- Init the 3 items random positions  
        self.itempos = sample(self[1:], 3)  
  
    return self  
  
# -----  
def draw_objects(self):  
    # -- Display walls as background  
    self.master.blit(self.wallpic, (0, 0))  
    # -- Draw the empty path  
    for emptypos_y, emptypos_x in self:  
        self.master.blit(self.bground, (emptypos_x * 50, emptypos_y * 50),  
                          (emptypos_x * 50, emptypos_y * 50, 50, 50))  
    # -- Display and positioning guardian  
    guardpos_y, guardpos_x = self.guardpos  
    self.master.blit(self.bground, (guardpos_x * 50, guardpos_y * 50),  
                      (guardpos_x * 50, guardpos_y * 50, 50, 50))  
    self.master.blit(self.guardpic, (guardpos_x * 50, guardpos_y * 50))  
    # -- Without it, MacGyver can't get into the same cell with the guard  
    self.extend([self.guardpos])  
    # -- Display and positioning items  
    for item, (pos_y, pos_x) in zip(self.itempos, self.itempos):  
        self.master.blit(item, (pos_x * 50, pos_y * 50))  
        # -- Add item and their position in tuple in list  
        self.image_position_list.append((item, (pos_y, pos_x)))
```

# OPENCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## Déplacements de MacGyver

- Suppression de MacGyver
- Mise à jour de sa position
- Affichage de MacGyver à sa nouvelle position

## Le Code

```
# -----  
def del_mac(self):  
    macpos_y, macpos_x = self.macpos  
    self.screen.master.blit(self.bground, (macpos_x * 50, macpos_y * 50),  
                             (macpos_x * 50, macpos_y * 50, 50, 50))  
  
# -----  
def update_mac(self, key, board):  
    """ Get arrows direction and move MacGyver to the right position """  
    macpos_y, macpos_x = self.macpos  
  
    offset_y, offset_x = self.arrows.get(key, (50, 0))  
    if (macpos_y + offset_y, macpos_x + offset_x) in board:  
        self.macpos = (macpos_y + offset_y, macpos_x + offset_x)  
  
# -----  
def show_mac(self):  
    macpos_y, macpos_x = self.macpos  
    self.screen.master.blit(self.macpic, (macpos_x * 50, macpos_y * 50))
```

# OPENCCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## Ramassage des objets

- Vérification de la position du joueur par rapport à l'objet
- Récupération de l'image correspondant à la position
- Décrémentation du compteur
- Positionnement de l'objet dans la barre d'élément

## Le Code

```
# -----  
def pickup(self, board, counter):  
    if self.macpos in board.itempos:  
        for item, (imgpos_y, imgpos_x) in board.image_position_list:  
            if self.macpos == (imgpos_y, imgpos_x):  
                counter.decrease()  
                board.itembar(item, counter)  
  
        board.itempos.remove(self.macpos)
```

# OPENCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## Victoire et défaite devant le gardien

- Vérification de la position de MacGyver par rapport a celle du gardien
- Vérification du compteur (Initialisé à 3 en début de partie)
- Si compteur = 0 : Affichage de la fenêtre WIN
- Sinon : Affichage de la fenêtre LOOSE

## Le Code

```
# -----  
def win_loose(self, player, counter):  
    if self.guardpos == player.macpos:  
        if counter.get_counter_value() == 0:  
            self.endgame = pg.display.set_mode((cst.WINSIZE, cst.WINSIZE))  
            self.endgame.blit(self.win, (0, 0))  
        else:  
            self.endgame = pg.display.set_mode((cst.WINSIZE, cst.WINSIZE))  
            self.endgame.blit(self.loose, (0, 0))
```



# OPENCCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

## Le fichier principal

- Optimisé au maximum
- Encapsulé dans une fonction `main`
- Elle-même appelée par le fichier `main.py`
- Initialisation du module Pygame et de toutes les classes
- N'est géré que par des appels de fonctions

## Le Code

```
# =====  
def main():  
    screen = maze.GameBoard()  
    board = screen.lab_struct()  
    screen.draw_objects()  
    hero = player.McGyver(screen)  
    counter = itemcounter.ItemCounter()  
  
    pg.display.flip()  
  
    # =====  
    while True:  
        ev = pg.event.wait()  
        key_pressed = pg.key.get_pressed()  
        # -- Close the window game by the CROSS button or ESCAPE key  
        if ev.type == pg.QUIT or key_pressed[pg.K_ESCAPE]:  
            break  
        elif ev.type == pg.KEYDOWN:  
            hero.del_mac()  
            hero.update_mac(ev.key, board)  
            hero.show_mac()  
            hero.pickup(board, counter)  
            screen.win_loose(hero, counter)  
  
    pg.display.flip()
```

# OPENCCLASSROOMS - PROJET 03

Aidez MacGyver à s'évader !

MEKO !