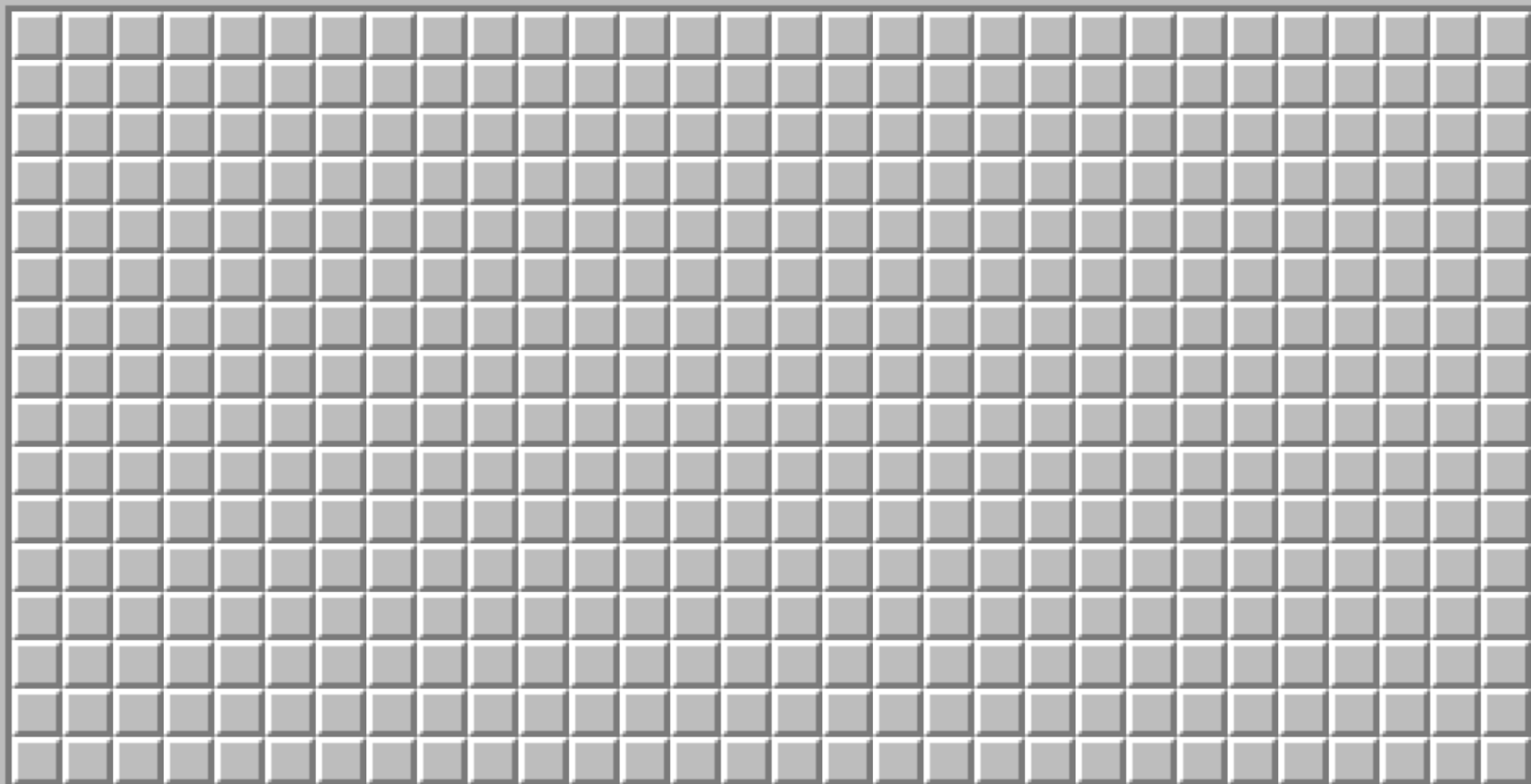


Code For Minesweeper



004



003

			☠				☠					☠			☠	1		1	☠	☠	☠	☠	1		1	☠	1	
										☠	☠	☠	4	☠	4	3	1	2	3	☠	4	2	1		1	1	2	1
	☠						☠	☠	3	4	4	5	☠	☠	2	☠	1	1	1	1				1	1	2	☠	
			☠						2	2	2	1	1	☠	☠	4	2	2	2	1				2	☠	4	2	
	☠					☠			2			1	3	☠	2		2	☠	2				2	☠	3	☠		
			☠		☠	☠	2					1	1	1			3	☠	3			1	1	1	1	1	2	1
	☠	☠	☠		☠		2	1	1	2	2	2	1		1	1	3	☠	3	1	1	1	☠	2	1	1	1	
	☠				☠				☠		☠	☠	1		1	☠	3	2	3	☠	1	1	2	☠	2	3	☠	2
			☠	☠							☠		3	1	3	☠	4	☠	4	2	1		1	2	☠	3	☠	2
							☠	☠			☠				☠	☠	4	☠	☠	1			2	2	3	1	1	
											☠				☠		4	4	4	3	1	2	1	3	☠	2		
☠							☠				☠					2	☠	☠	2	☠	2	☠	3	☠	2	1	1	
				☠	☠		☠			☠					1	1	2	2	2	1	2	2	4	3	2	1	☠	
☠				☠	☠		☠		☠	3	1	1	1	☠	1						1	2	☠	☠	2	3	3	
			☠				☠	☠		☠	2			1	1	1					1	☠		2	3	☠	☠	
			☠		☠					1											1					☠	☠	



	2	1	1	1	1	1	2	3	1	2			1	1	2	1	1	1	2	1							
2	5	1	5	2	1	2	1	2	1	3	2	1	2	1	2	2	1	2	1	4	2	1					
1	1	1	1	3	3	1	3	3	3	4	1	2	1					3	1	1	1			1	2	2	
2	4	1	1	3	1	1	2	1	1	2	1	2	1	1		1	1	2	3	1	4	3	1	1	1	2	1
	1	3	3	4	3	4	2	2	1	1					2	1	3	1	5	1	3	1	2	2	1	3	2
	1	2	1	2	1	3	1	3	1	2	2	2	2	2	1	2	1	4	3	1	1	4	2	1	2	1	2
	2	1	3	2	1	3	1	4	1	3	1	1	2	1	1	2	2	4	1	5	1	3	2	1	2	1	2
	2	1	3	1	1	1	1	3	1	3	2	3	3	3	2	1	1	1	3	1	3	2	1	2	1	1	3
	1	1	2	1	2	1		1	1	1		1	1	2	1		1	1	2	1	1	1	2	1	3	3	2
	1	1	3	3	1							1	2	1						1	3	1	1	1	1	1	
1	3	1	3	1	2	1	1	1	1				1	1	1				1	2	2	2	3	1	3	1	
1	3	1	3	1	1	1	2	1		1	2	2	2	1	1		1	1	2	1	1	3	1	3	2	1	
1	3	3	4	2	1	1	1	2	2	1	3	1	1	3	1	1		2	1	3	3	1	4	2	2	1	
1	2	1	1	1	1	1	1	1	1	1	3	1	4	1	2	1		2	1	3	2	3	1	2	2	1	
1	3	3	4	2	1		1	1	2	1	2	1	2	1	2	1	1	1	1	2	1	3	4	1	2		
1	2	1	1				1	1	1							1	1	1		1	2	1	3	1	2		

Create an empty grid and place bombs

```
import random

def GenerateMineSweeperMap(n, k):
    arr = [[0 for row in range(n)] for column in range(n)]

    for num in range(k):
        x = random.randint(0,n-1)
        y = random.randint(0,n-1)
        arr[y][x] = 'X'
```

Assigning values to spaces touching bombs

```
if (x >=0 and x <= n-2) and (y >= 0 and y <= n-1):
    if arr[y][x+1] != 'X':
        arr[y][x+1] += 1 # center right

if (x >=1 and x <= n-1) and (y >= 0 and y <= n-1):
    if arr[y][x-1] != 'X':
        arr[y][x-1] += 1 # center left

if (x >= 1 and x <= n-1) and (y >= 1 and y <= n-1):
    if arr[y-1][x-1] != 'X':
        arr[y-1][x-1] += 1 # top left

if (x >= 0 and x <= n-2) and (y >= 1 and y <= n-1):
    if arr[y-1][x+1] != 'X':
        arr[y-1][x+1] += 1 # top right
```

```
if (x >= 0 and x <= n-1) and (y >= 1 and y <= n-1):
    if arr[y-1][x] != 'X':
        arr[y-1][x] += 1 # top center

if (x >=0 and x <= n-2) and (y >= 0 and y <= n-2):
    if arr[y+1][x+1] != 'X':
        arr[y+1][x+1] += 1 # bottom right

if (x >= 1 and x <= n-1) and (y >= 0 and y <= n-2):
    if arr[y+1][x-1] != 'X':
        arr[y+1][x-1] += 1 # bottom left

if (x >= 0 and x <= n-1) and (y >= 0 and y <= n-2):
    if arr[y+1][x] != 'X':
        arr[y+1][x] += 1 # bottom center
```

```
def GeneratePlayerMap(n):
    arr = [['-' for row in range(n)] for column in range(n)]
    return arr
```

Checks

```
def CheckWon(map):
    for row in map:
        for cell in row:
            if cell == '-':
                return False
    return True

def CheckContinueGame(score):
    print("Your score: ", score)
    isContinue = input("Do you want to try again? (y/n) :")
    if isContinue == 'n':
        return False
    return True
```

Assigning Difficulty

```
def Game():
    GameStatus = True
    while GameStatus:

        difficulty = input("Select your difficulty (b, i, h):")
        if difficulty.lower() == 'b':
            n = 5
            k = 3
        elif difficulty.lower() == 'i':
            n = 6
            k = 8
        else:
            n = 8
            k = 20
```


Ending the game

```
minesweeper_map = GenerateMineSweeperMap(n, k)
player_map = GeneratePlayerMap(n)
score = 0

while True:

    if CheckWon(player_map) == False:

        print("Enter your cell you want to open :")
        x = input("X (1 to 5) :")
        y = input("Y (1 to 5) :")
        x = int(x) - 1 # 0 based indexing
        y = int(y) - 1 # 0 based indexing
```

```
        if (minesweeper_map[y][x] == 'X'):

            print("Game Over!")
            DisplayMap(minesweeper_map)
            GameStatus = CheckContinueGame(score)
            break

        else:

            player_map[y][x] = minesweeper_map[y][x]
            DisplayMap(player_map)
            score += 1

    else:

        DisplayMap(player_map)
        print("You have Won!")
        GameStatus = CheckContinueGame(score)
        break
```

Select your difficulty (b(beginner), i(intermediate), h(hard)):b

Enter your cell you want to open :

X (1 to 5) :1

Y (1 to 5) :1

```
1   -   -   -   -  
-   -   -   -   -  
-   -   -   -   -  
-   -   -   -   -  
-   -   -   -   -
```

Enter your cell you want to open :

X (1 to 5) :1

Y (1 to 5) :2

```
1   -   -   -   -  
  
1   -   -   -   -  
-   -   -   -   -  
-   -   -   -   -  
-   -   -   -   -
```

Enter your cell you want to open :

X (1 to 5) :2

Y (1 to 5) :1

Game Over!

```
1   X   1   0   0  
  
1   2   2   2   1  
  
0   1   X   2   X  
  
0   1   1   2   1  
  
0   0   0   0   0
```

Your score: 2

Do you want to try again? (y/n) :

Sources

<https://medium.com/swlh/this-is-how-to-create-a-simple-minesweeper-game-in-python-af02077a8de>

<https://www.askpython.com/python/examples/create-minesweeper-using-python>

<https://www.freecodecamp.org/news/if-name-main-python-example/>