
WELCOME TO
CODEWARS
Skull and Crossbones

RESOURCES

WOOD

RUM

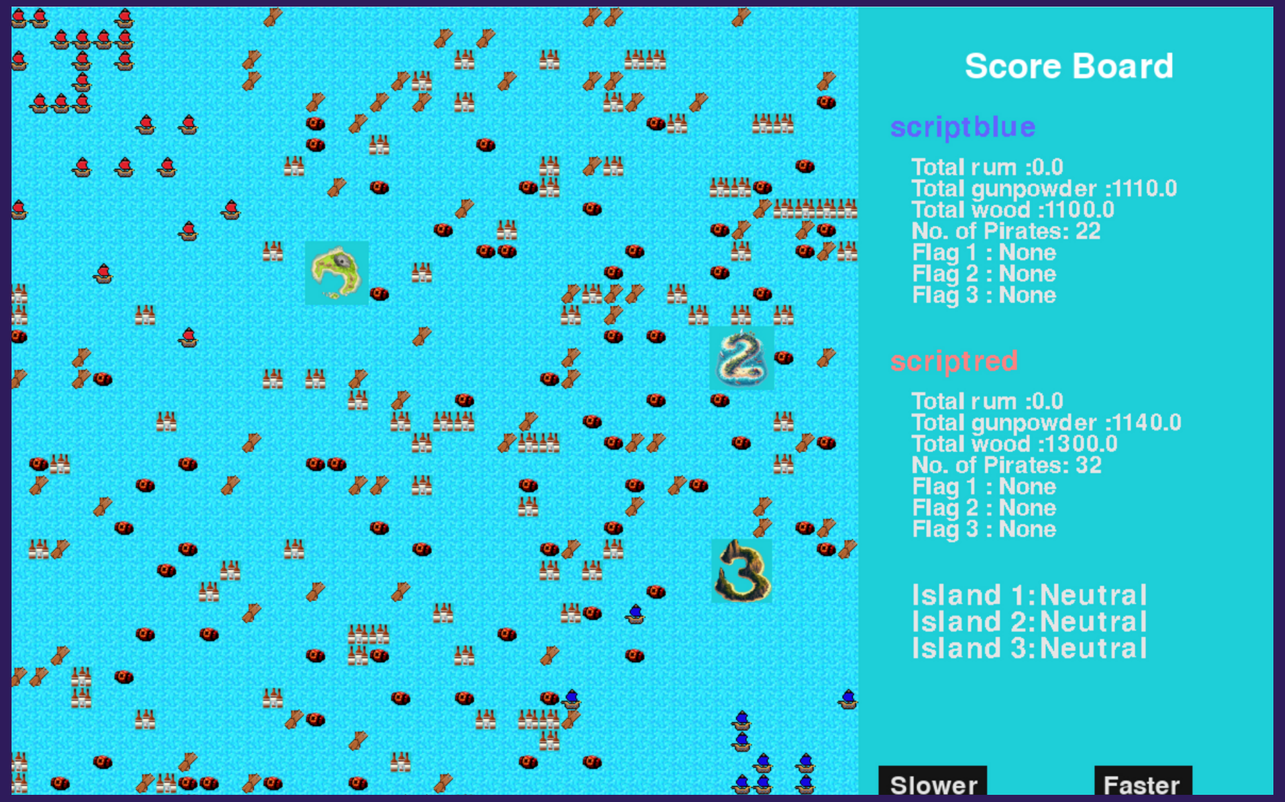
GUNPOWDER

GAME RULES

Each Team, i.e., their scripts compete head to head in the game.
The teams are marked as Blue and Red.

Each team is given some initial number of pirates and some amount of wood and gunpowder (initially, no rum is given)

3 islands spawn in the map as different locations each game.
The pirates are supposed to move around and explore the sea (i.e., the map) to
1.) Discover Islands
2.) Conquer the islands and defend them
3.) Gather resources

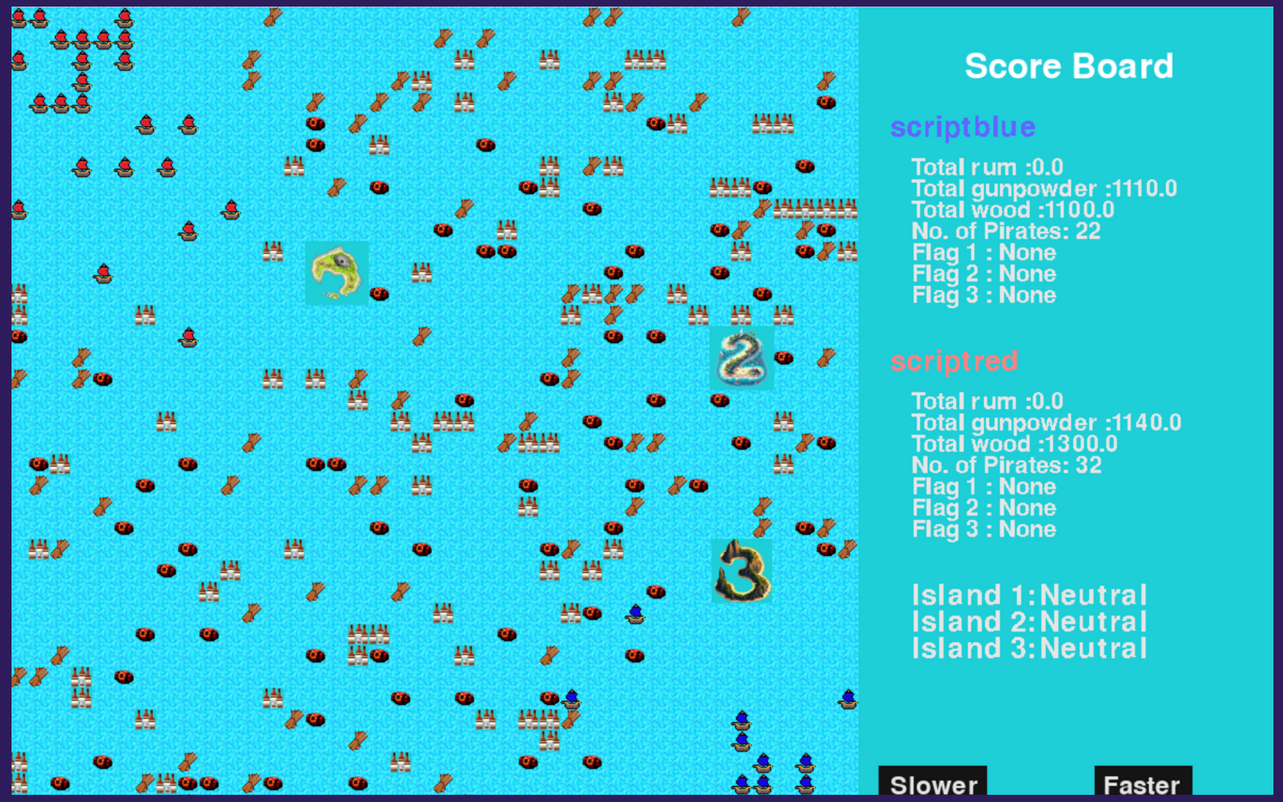


GAME RULES

All the resources are common to the entire team and usable by any pirate of the team. In other words, all the pirates share a common pool of resources. No pirate has any individual count of any resource.

The map tiles may contain rum, gunpowder or wood, which gets collected by the pirate (i.e., the pirate's team) when the pirate steps on to that tile.

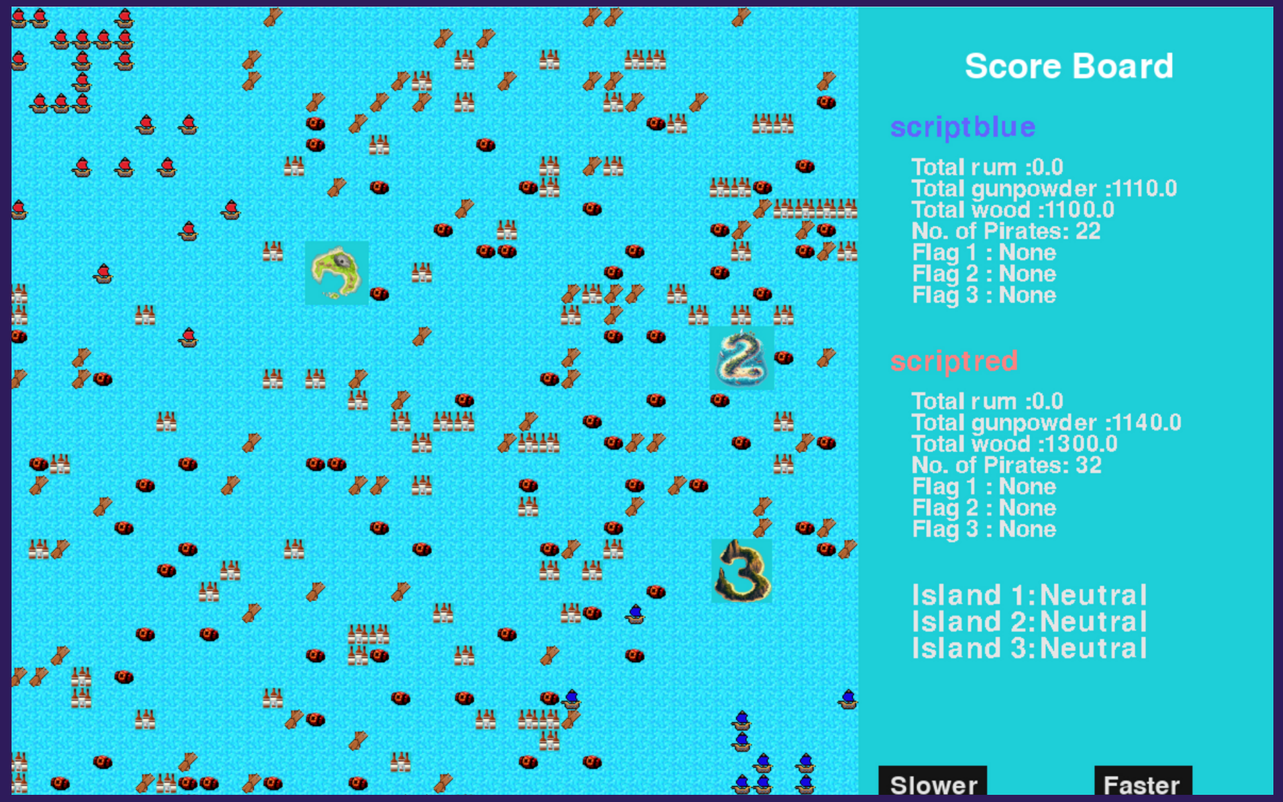
Only gunpowder gets randomly regenerated in the map while rum and wood do not. So, only gunpowder is unlimited.



GAME RULES

Gunpowder : 100 amount of gunpowder is needed to kill an enemy pirate. Pirates fight ONLY when then are on the SAME tile (not when they are adjacent to each other).

- If both the fighting pirates' team have more than than 100 amount of gunpowder, then both dies.
- Otherwise, if one team has more than 100 amount of gunpowder and the other doesn't then, only the later team's pirate die on fighting.
- If both team have less than 100 amount of gunpowder, then the pirate cannot kill each other and both the fighting pirates survive.



GAME RULES

Rum : 50 rum gunpowder is needed to generate a new pirate. A pirate automatically gets spawned as the amount of rum exceeds the required amount.

Wood: 50 wood is needed to build walls on an island. Walls serve a defensive purpose. When there are walls on an island, pirates (irrespective of the team which built the walls) can neither enter nor exit the island.

GAME RULES

WALLS

Walls can be built by a team only when atleast one pirate of that team is present and no pirate of the enemy team is present on the island.

Walls stay built 50 game frames and get destroyed automatically. No pirate can damage/destroy the wall (they get destroyed only when the timer ends). There is a cooldown timer of 35 frames after the walls get destroyed. During this period, NEITHER of the teams can build and wall on the island.

GAME RULES

HOW TO DECIDE MOVEMENT

We have signals, which are strings, which can be defined by the user. There are 2 types of signals:

1. Each pirate has its own signal
2. There is a signal for each team. All the pirates of that team can access that signal.

The team can read the signals of all the pirates at any point of time.

PIrates can access and change their own signal as well as the signal of the team but not the individual signal of other pirates.

GAME RULES

WINNER

1. If any team captures all 3 islands, then that team instantly wins.
2. If no team can capture all 3 islands and the time runs out, then the team with more number of captures islands at the end wins.
3. If both teams have same number of captured islands, the team with more number of surviving pirates win.
4. When all the above 3 criteria are tied, then the team with more surviving pirates win.
5. Lastly, the winner will be decided as per the combinations of resources they have.

PYTHON

Data Types: There are no data types in python

Ex: x= 5

Ex: y= 2.5

Ex: s= ”hello”

Ex: f= True

Ex: c1 = s[0] # c1 == ‘h’

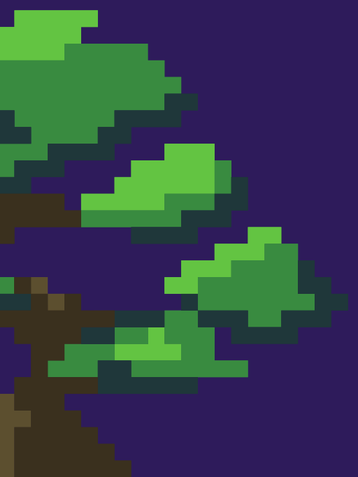
Ex: c2 = s[0:3] # c2 == “hel”

Ex: c3 = s[-1] # c3 == ‘o’

Print: **‘+’** is used to concatenate strings. So be careful about convert the variables which are not strings to string first

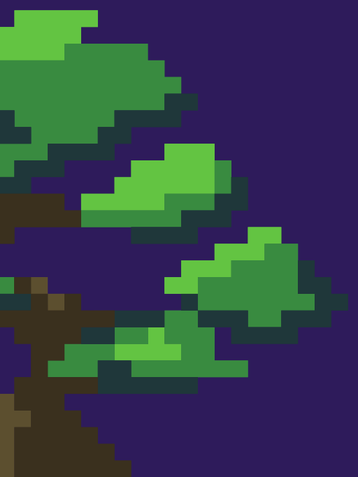
Ex: t = s + str(x) + str(x+2)
t == “hello57”

Ex: print(“hello”)
same as print(s)



Ex: print(“hello” + str(x))
outputs “hello5”

Ex: print(“hello”, x)
outputs “hello 5”



Lists: Its an array of variables

Ex. list = [ ]

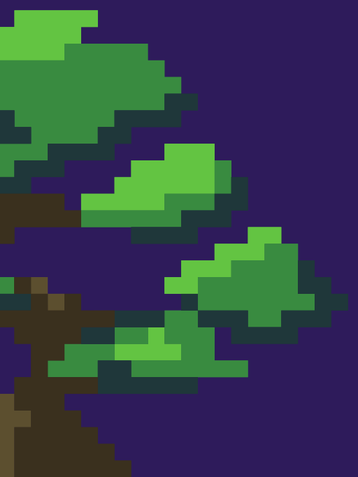
Ex: list1 = [1,2,3,4]

Ex: list2 = [”hello1", “hello2”, “hello3”, “hello4”]

Ex: list2.append(“hello5”)
Now the **list2** became [”hello1", “hello2”, “hello3”, “hello4”, “hello5”]

Ex: s1 = list2[3]
s1 == “hello4”

Ex: s2 = list2[-1]
s2 == “hello5”



Ex: print(list2[1])
outputs “hello2”

Ex: list3 = [100, 99, 98, 97]
list1.append(list3[3])

Now **list1** = [1,2,3,4,97]

Ex: list1[0] = list1[0] + 2
Now **list1** becomes [3,2,3,4]

Ex. del list1[1]
Now **list1** becomes [3,2,4]

Ex. del list1
Deletes the entire list

Ex. list4 = [2,5,7, “wncc”, 10, “codewars”]
A list can contain values of different data types

Tuples: Very similar to lists, expect that tuples cannot be changed after they are created. We use **(.)** for tuples.

Ex: t1 = (1,2,3)
print(t1[1])
t1[0] = 6 # this will give error

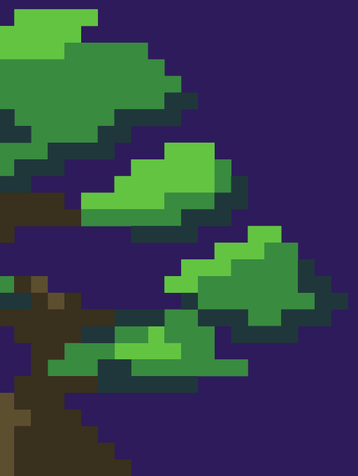
Ex. del t1[0] # this will give error
del t1 # works

NOTE: All the coordinates will be in the form of tuples

NOTE: All the coordinates will be in the form of tuples

First understand the purpose and return values of the given functions before reading the below section
So, use

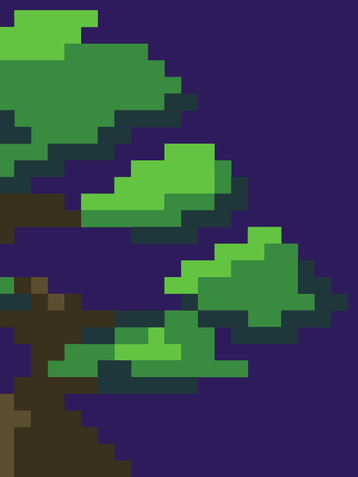
```
x, y = pirate.getPosition()
        or
t = pirate.getPosition()
x = t[0]
y = t[1]
```



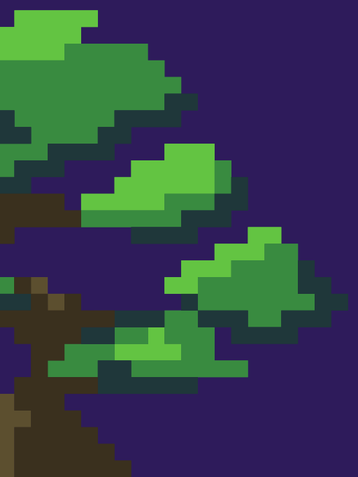
Loops: In python there are no **{.}** to mark the start and end of blocks (if blocks, loop blocks, etc.). **Blocks** are represented by **indentation (spaces)** and **colons.**

Ex: for s in list2:
    print(s)

Ex: for i in range(5):
    print(i)
(**range(5)** returns [0,1,2,3,4], i.e., a list from 0 to 4)
(**reversed(range(5))** returns [4,3,2,1,0])



Ex:
```
i = 0
while i < 5:
    print(i)
    i += 1
```



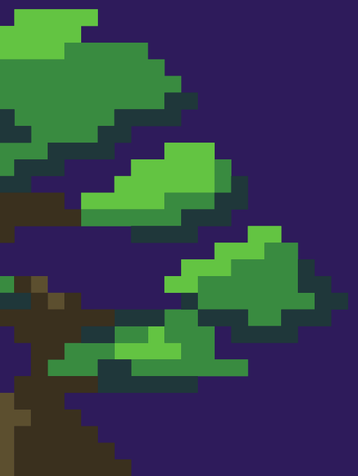
Ex:
```
i = 0
while i < len(list3):
    print(i)
    i += 1
```

(this loop will continue till i < 4)

If blocks: Similar to what we saw in case of loops, if blocks are also marked using **colon** and **indentation.**

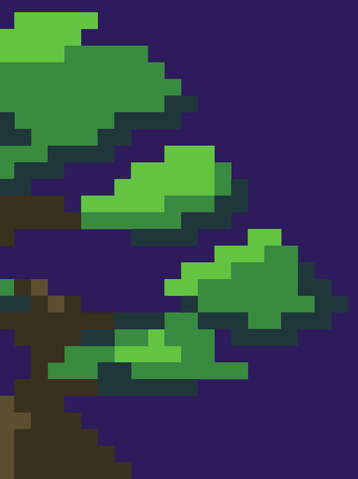
Ex: if list[0] == “hello0”:
    print(“found hello0 at index 0”)

Ex: if “hello0” in list1:
    print(“found hello0 but don’t know its index”)



Ex: if x > y:

```
        print("x is greater than y")
    elif x < y:
        print("x is less than y")
    else:
        print("x and y are equal")
```



Functions: Similar to what we saw in case of loops, if blocks are also marked using **colon** and **indentation.**

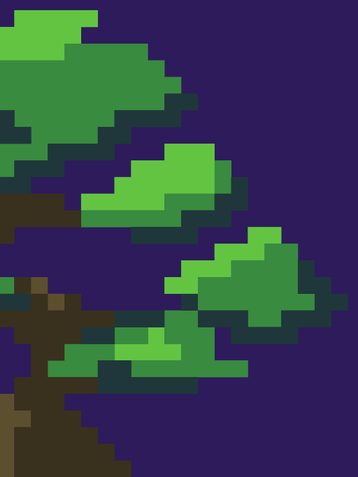
Ex: def greet(name):
print("Hello, “, name)

Ex: def findSum(name=”buddy”):
print("Hello, “, name)

Here, we have set a default string to the name argument. So, if no **name** argument is passed at function call, then **name** is set to “buddy” by default.

Ex: greet(“WnCC”)

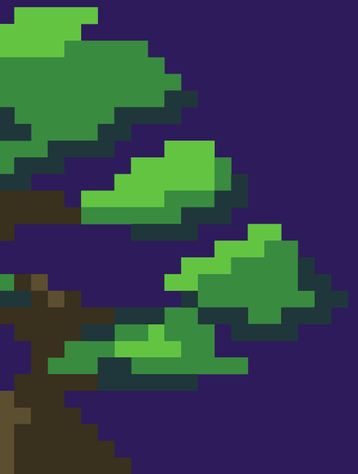
Ex: z = findSum(3, 5)
So, **z** == 8



Split: This function is used to break a string into parts about a particular character (whitespace by default). It returns a list of those parts.

Ex: sentence = “Codewars is onnn”

```
listOfWords = sentence.split()
print(“Sentence has “, len(listOfWords), “ words)
for word in listOfWords:
    print(word)
```



This outputs: “Sentence has 3 words

Codewars

is

onnn”

Ex:

```
x = 10
y = 5
islandNo = 2
pos = str(x) + ”,” + str(y) + ”,” + str(islandNo)
```

```
islandInfo = pos.split(“,”)
islandX = int(islandInfo[0])
islandY = int(islandInfo[1])
islandIndex = int(islandInfo[2])
```

Here, **islandInfo** = [”10", “5”, “2”]


THANKYOU FOR
PLAYiNG