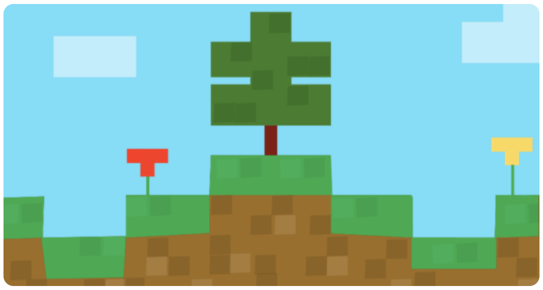
Projects

CodeCraft

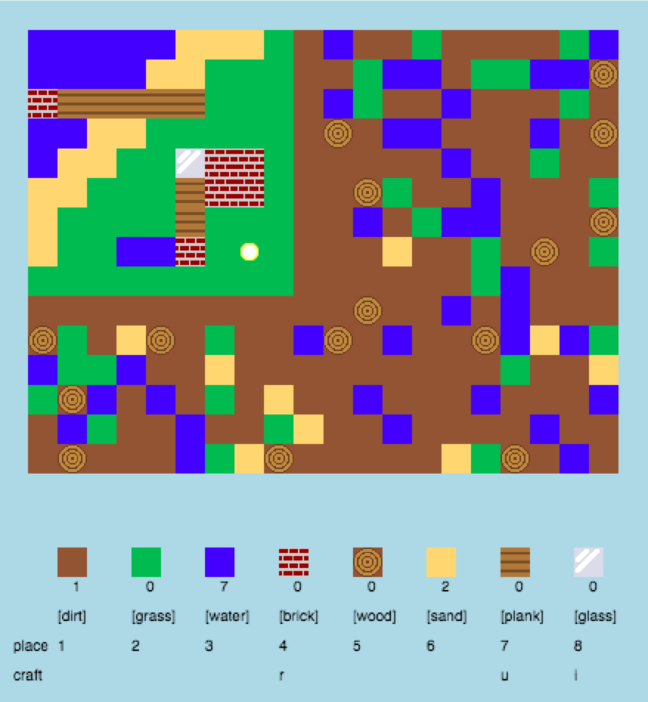
Design and code improvements to a 2D version of Minecraft.

Python



Step 1 Introduction

In this project, you’ll design and code improvements to a 2D version of Minecraft.



	🟫	🟩	🟦	▦	◎	🟨	▤	▨
	1	0	7	0	0	2	0	0
	[dirt]	[grass]	[water]	[brick]	[wood]	[sand]	[plank]	[glass]
place	1	2	3	4	5	6	7	8
craft				r			u	i

Additional information for club leaders

If you need to print this project, please use the **Printer friendly version** (https://projects.raspberrypi.org/en/projects/codecraft/print).

Club leader notes

Introduction:

In this project, children will learn aspects of graphics and game design by making improvements to a basic 2D Minecraft clone. Children will create new resources, as well as crafting rules for combining resources to make new ones. This will be achieved by understanding and manipulating variables, lists and dictionaries.

Online Resources

This project uses Python 3. We recommend using **trinket** (https://trinket.io/) to write Python online. This project contains the following Trinkets:

- **'CodeCraft' starter project – rpf.io/codecraft-on** (http://rpf.io/codecraft-on)

There is also a trinket containing the completed project:

- **'CodeCraft' Finished – trinket.io/python/ebc5b0148b** (https://trinket.io/python/ebc5b0148b)

Offline Resources

This project can be **completed offline** (https://www.codeclubprojects.org/en-GB/resources/python-working-offline/) if preferred. You can access the project resources by clicking the 'Project Materials' link for this project. This link contains a 'Project Resources' section, which includes resources that children will need to complete this project offline. Make sure that each child has access to a copy of these resources. This section includes the following files:

- codecraft/codecraft.py
- codecraft/variables.py
- codecraft/brick.gif
- codecraft/dirt.gif
- codecraft/glass.gif
- codecraft/grass.gif
- codecraft/plank.gif
- codecraft/player.gif

- codecraft/sand.gif
- codecraft/water.gif
- codecraft/wood.gif

You can also find a completed version of this project in the 'Volunteer Resources' section, which contains:

- codecraft-finished/codecraft.py
- codecraft-finished/variables.py
- codecraft-finished/brick.gif
- codecraft-finished/dirt.gif
- codecraft-finished/glass.gif
- codecraft-finished/grass.gif
- codecraft-finished/plank.gif
- codecraft-finished/player.gif
- codecraft-finished/sand.gif
- codecraft-finished/water.gif
- codecraft-finished/wood.gif

(All of the resources above are also downloadable as `.zip` files.)

Learning Objectives

- Creating and editing graphics;
- Game design;
- Editing:
  - Variables;
  - Lists;
  - Dictionaries.

This project covers elements from the following strands of the **Raspberry Pi Digital Making Curriculum** (http://rpf.io/curriculum):

- **Design basic 2D and 3D assets.** (https://www.raspberrypi.org/curriculum/design/creator)

- **Combine programming constructs to solve a problem.** (https://www.raspberrypi.org/curriculum/programming/builder)

Challenges

- "Build your world" - Playing the game, placing and crafting existing blocks;

- "Change your world size" - editing the `MAPWIDTH` and `MAPHEIGHT` variables to change the world size;
- "Creating sand" - Creating a new sand resource, along with associated game data.
- "Crafting glass from sand" - Creating a new craftable glass resource.
- "Create more resources" - Use what's been learnt to create more blocks and crafting rules.

Frequently Asked Questions

- Children may need reminding that elements of a dictionary/list are separated by a comma. For example, when adding inventory items, graphics and crafting rules to the game.

Project materials

Project resources

- **.zip file containing all project resources** (https://projects-static.raspberrypi.org/projects/codecraft/7fb82e483c4281eb405b52aac026ded63eab447f/en/resources/codecraft-resources.zip)
- **Online Trinket containing all 'CodeCraft' project resources** (http://rpf.io/codecraft-on)

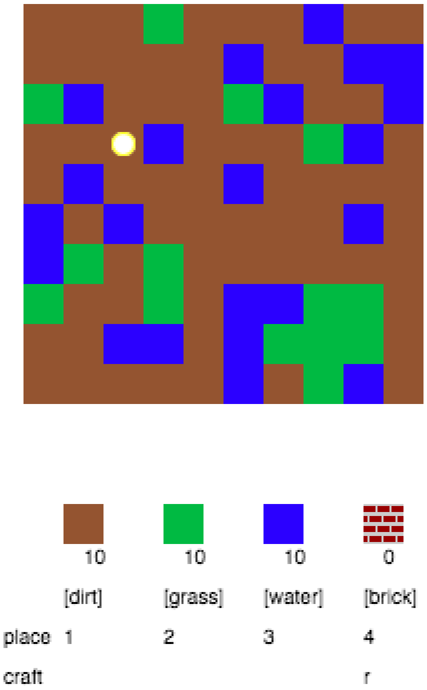
Club leader resources

- **.zip file containing all completed project resources** (https://projects-static.raspberrypi.org/projects/codecraft/7fb82e483c4281eb405b52aac026ded63eab447f/en/solutions/codecraft-solution.zip)
- **Online completed Trinket project** (https://trinket.io/python/ebc5b0148b)

Step 2 Playing the game

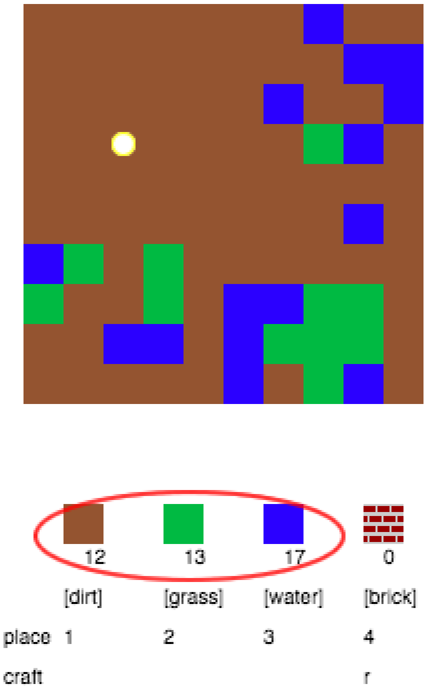
- Open this trinket: **rpf.io/codecraft-on** (http://rpf.io/codecraft-on).

- Use the WASD keys to move your player around the world, which is full of different resources (dirt, grass and water).



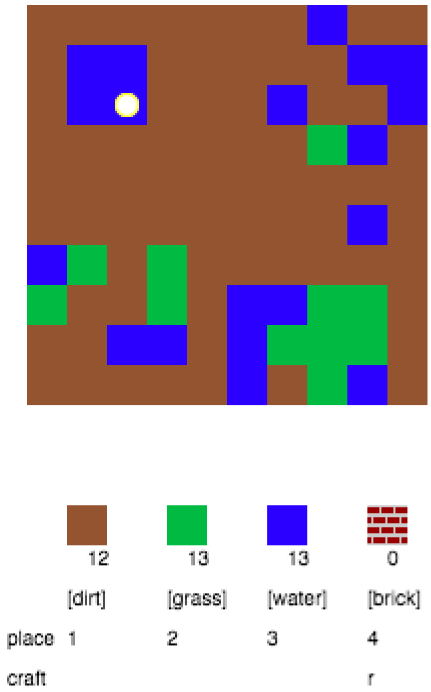
	🟫	🟩	🟦	▨
	10	10	10	0
	[dirt]	[grass]	[water]	[brick]
place	1	2	3	4
craft				r

- You can press the spacebar to collect resources. Pick up a few of each type of resource, and you'll see them added to your inventory.



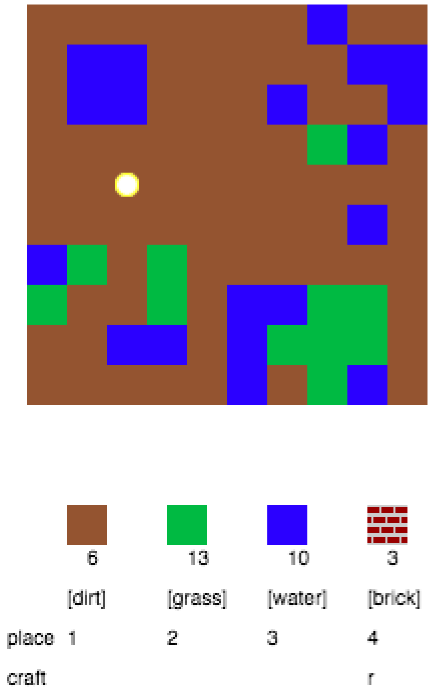
	🟫	🟩	🟦	▨
	12	13	17	0
	[dirt]	[grass]	[water]	[brick]
place	1	2	3	4
craft				r

- Press the number keys (1 to 3) to place a resource on the map. For example, press 3 to place some water on the map. This will only work if you have some water in your inventory.



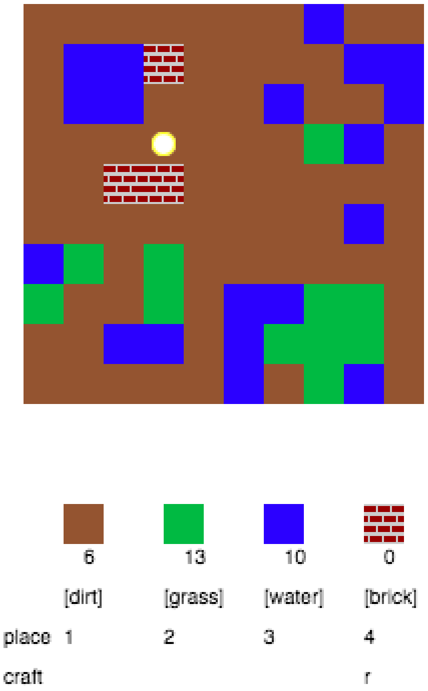
	🟫	🟩	🟦	🧱
	12	13	13	0
	[dirt]	[grass]	[water]	[brick]
place	1	2	3	4
craft				r

- You can craft an item by pressing the key displayed in the menu. Crafting means combining items you already have in your inventory to create new ones. Try pressing the ‘r’ key to craft a new brick (as long as you have 2 dirt and 1 water in your inventory).



	🟫	🟩	🟦	🧱
	6	13	10	3
	[dirt]	[grass]	[water]	[brick]
place	1	2	3	4
craft				r

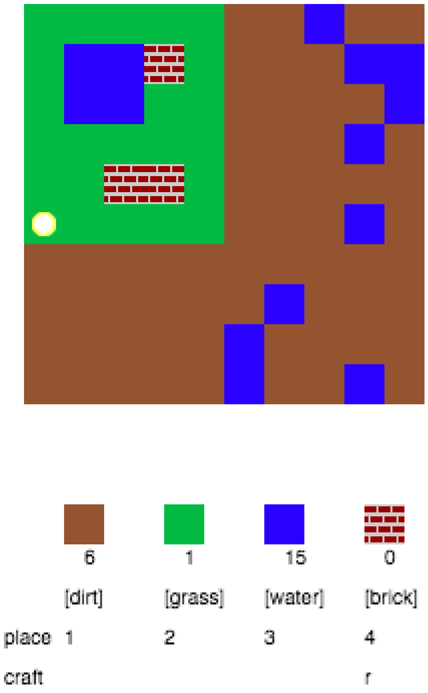
- You can then press they ‘4’ key to place your crafted bricks.



	🟫	🟩	🟦	▨
	6	13	10	0
	[dirt]	[grass]	[water]	[brick]
place	1	2	3	4
craft				r

Step 3 Challenge: Build your world

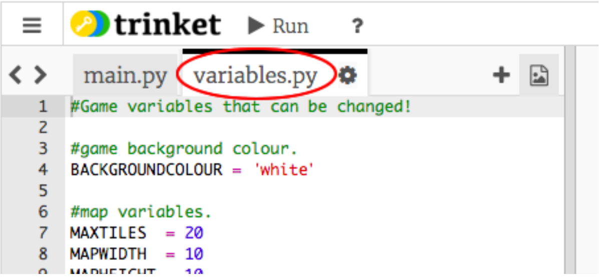
Can you build a house, with a garden and a swimming pool? What else can you create?



Step 4 Customising your game

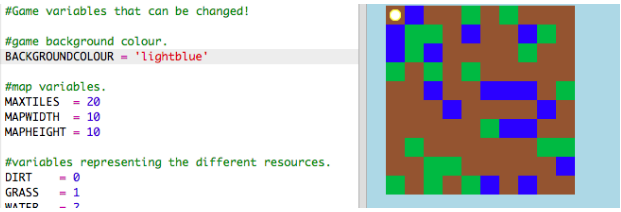
Let’s modify some variables to change how your game works.

- Click the `variables.py` file to see some variables that can be changed.



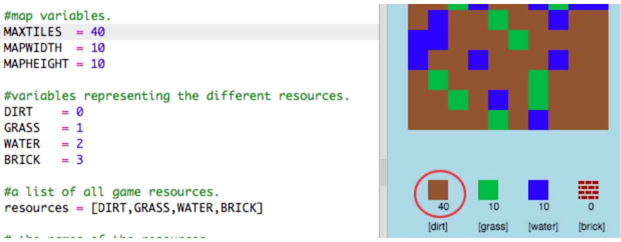
```
#Game variables that can be changed!

#game background colour.
BACKGROUNDCOLOUR = 'white'

#map variables.
MAXTILES   = 20
MAPWIDTH   = 10
MAPHEIGHT  = 10
```

- Change the value of your `BACKGROUNDCOLOUR` variable, and click ‘Run’ to see the change to your game.



```
#Game variables that can be changed!

#game background colour.
BACKGROUNDCOLOUR = 'lightblue'

#map variables.
MAXTILES  = 20
MAPWIDTH  = 10
MAPHEIGHT = 10

#variables representing the different resources.
DIRT    = 0
GRASS   = 1
WATER   = 2
```

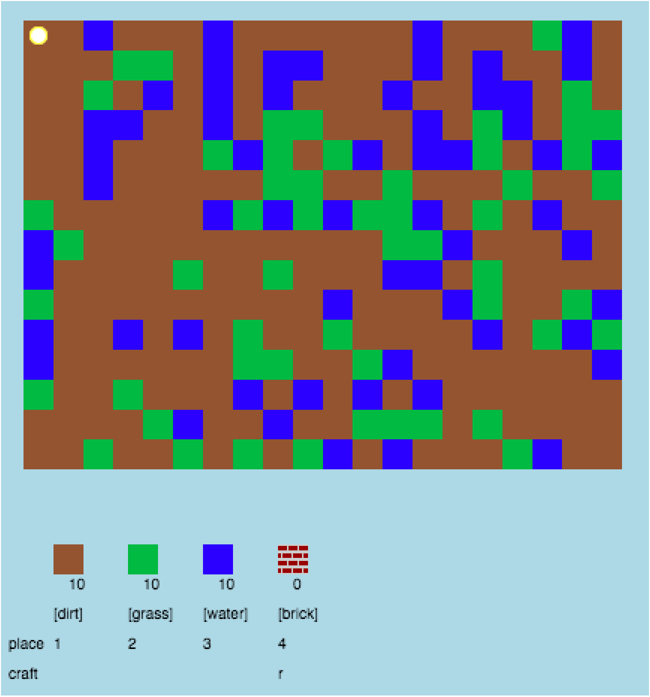
- The variable `MAXTILES` is the amount of each resource that can be held in your inventory. Change this variable if you want to store more (or less) than 20 of each resource.



```
#map variables.
MAXTILES  = 40
MAPWIDTH  = 10
MAPHEIGHT = 10

#variables representing the different resources.
DIRT    = 0
GRASS   = 1
WATER   = 2
BRICK   = 3

#a list of all game resources.
resources = [DIRT,GRASS,WATER,BRICK]

#the names of the resources
```

Step 5 Challenge: Change your world size

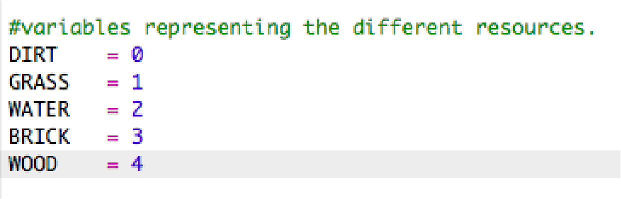
Can you change the values of your `MAPWIDTH` and `MAPHEIGHT` variables to change the size of the world?



Step 6 Creating a new wood resource

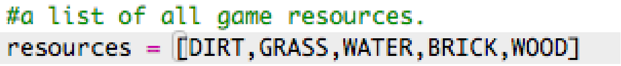
Let's create a new wood resource. To do this, you'll need to add to some variables in your `variables.py` file.

- First, you need to give your new resource a number. You'll then be able to use the word `WOOD` in your code instead of the number 4.



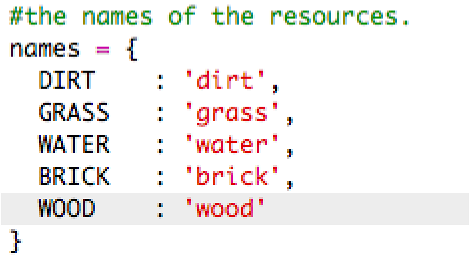
```
#variables representing the different resources.
DIRT    = 0
GRASS   = 1
WATER   = 2
BRICK   = 3
WOOD    = 4
```

- You should add your new `WOOD` resource to your list of `resources`.



```
#a list of all game resources.
resources = [DIRT,GRASS,WATER,BRICK,WOOD]
```

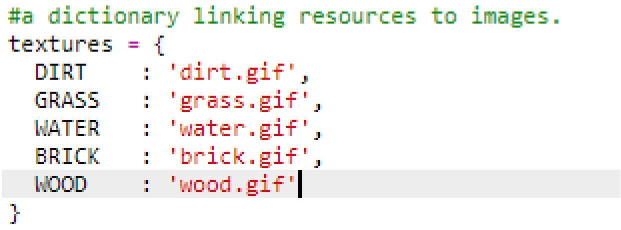
- You should also give your resource a name, which will be displayed in the inventory.



```
#the names of the resources.
names = {
  DIRT    : 'dirt',
  GRASS   : 'grass',
  WATER   : 'water',
  BRICK   : 'brick',
  WOOD    : 'wood'
}
```

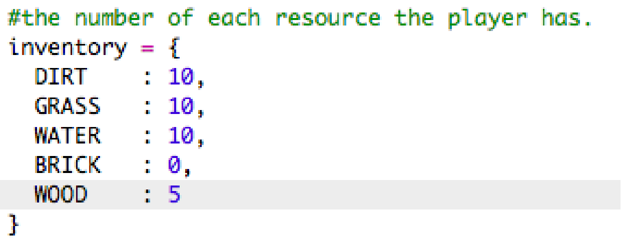
  Notice the comma `,` at the end of the line above.

- Your resource will also need an image. The project already includes an image called `wood.gif`, which you should add to the `textures` dictionary.



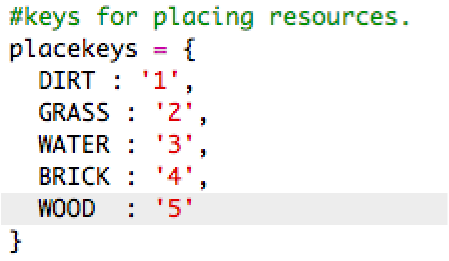
```
#a dictionary linking resources to images.
textures = {
  DIRT    : 'dirt.gif',
  GRASS   : 'grass.gif',
  WATER   : 'water.gif',
  BRICK   : 'brick.gif',
  WOOD    : 'wood.gif'
}
```

- Add the number of your resource that should be in your `inventory` to start with.



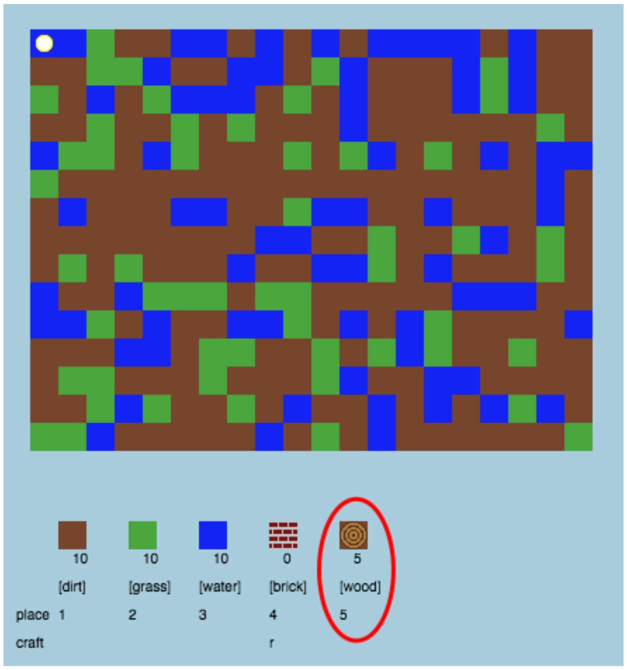
```
#the number of each resource the player has.
inventory = {
  DIRT    : 10,
  GRASS   : 10,
  WATER   : 10,
  BRICK   : 0,
  WOOD    : 5
}
```

- Finally, add the key that you'll press to place wood in the world.



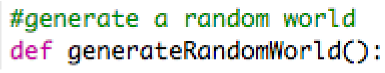
```
#keys for placing resources.
placekeys = {
  DIRT : '1',
  GRASS : '2',
  WATER : '3',
  BRICK : '4',
  WOOD  : '5'
}
```

- Run your project to test it. You'll see that you now have a new 'wood' resource in your inventory.



```
        10      10      10      0       5
        [dirt]  [grass] [water] [brick] [wood]
place   1       2       3       4       5
craft                           r
```

- There's no wood in your world! To fix this, click on your `main.py` file and find the function called `generateRandomWorld()`.

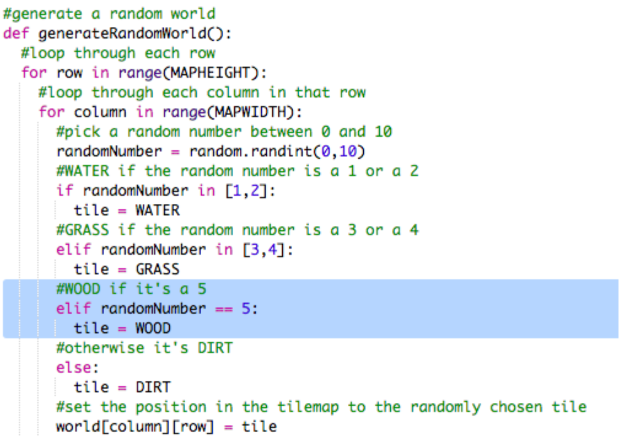


```
#generate a random world
def generateRandomWorld():
```

This code generates a random number between 0 and 10, and uses the number to decide which resource to place:

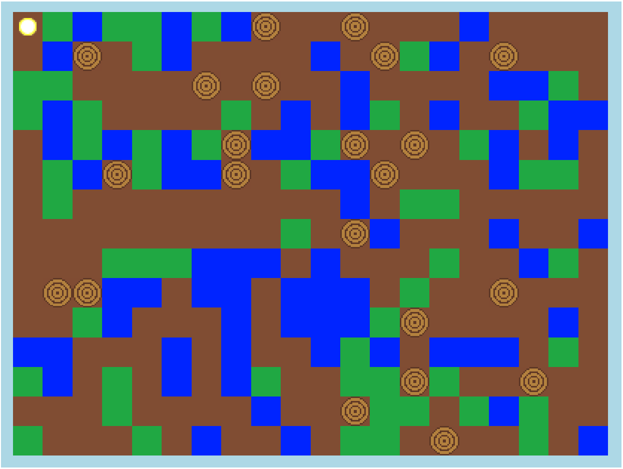
- 1 or 2 = water
- 3 or 4 = grass
- anything else = DIRT

- Add this code to add wood to your world whenever the `randomNumber` is 5.



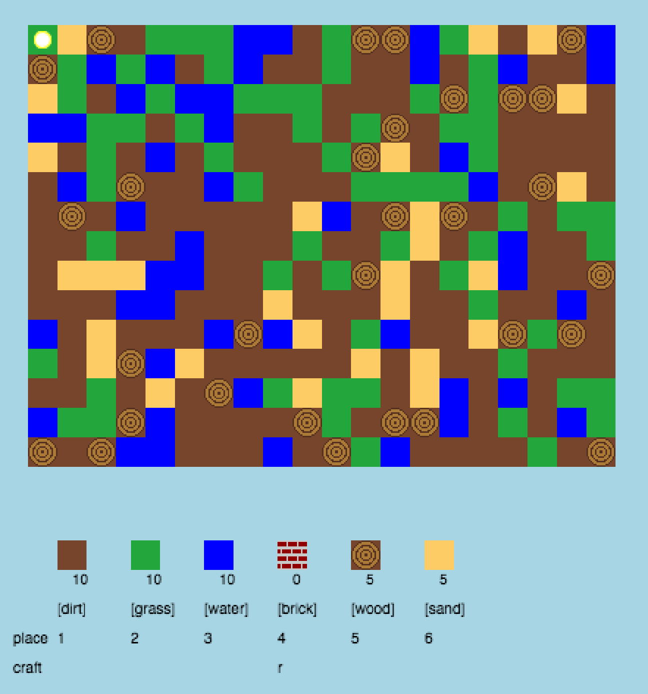
```
#generate a random world
def generateRandomWorld():
  #loop through each row
  for row in range(MAPHEIGHT):
    #loop through each column in that row
    for column in range(MAPWIDTH):
      #pick a random number between 0 and 10
      randomNumber = random.randint(0,10)
      #WATER if the random number is a 1 or a 2
      if randomNumber in [1,2]:
        tile = WATER
      #GRASS if the random number is a 3 or a 4
      elif randomNumber in [3,4]:
        tile = GRASS
      #WOOD if it's a 5
      elif randomNumber == 5:
        tile = WOOD
      #otherwise it's DIRT
      else:
        tile = DIRT
      #set the position in the tilemap to the randomly chosen tile
      world[column][row] = tile
```

- Test your project again. This time, you should see some wood appear in your world.



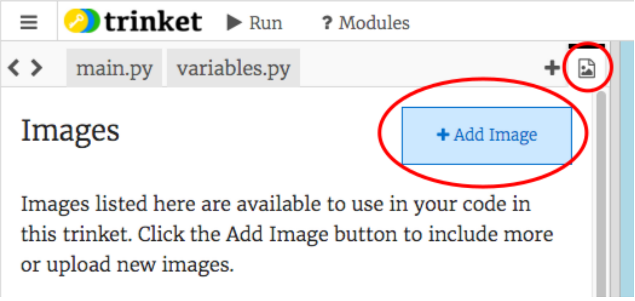
Step 7 Challenge: Creating sand

Can you add a `SAND` resource to your game? You can use the steps above to help you.



	🟫	🟩	🟦	▦	◎	🟨
	10	10	10	0	5	5
	[dirt]	[grass]	[water]	[brick]	[wood]	[sand]
place	1	2	3	4	5	6
craft				r		

The project already includes a `sand.gif` image, but you can create and upload your own if you prefer.

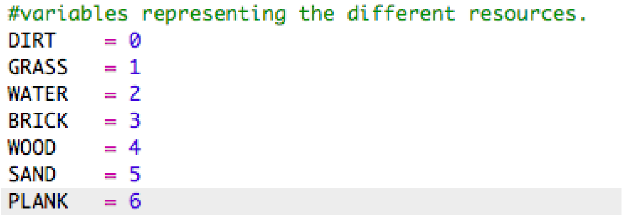


Images listed here are available to use in your code in this trinket. Click the Add Image button to include more or upload new images.

Step 8 Crafting planks from wood

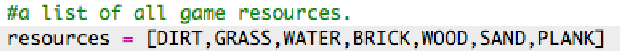
Let's create a new plank resource that be crafted from wood.

- First, add a new `PLANK` variable to your game.



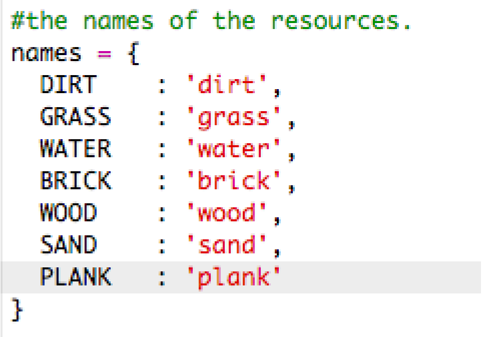
```
#variables representing the different resources.
DIRT    = 0
GRASS   = 1
WATER   = 2
BRICK   = 3
WOOD    = 4
SAND    = 5
PLANK   = 6
```

- Add a new `PLANK` variable to your game.



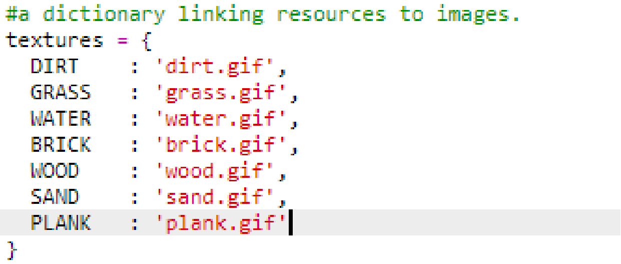
```
#a list of all game resources.
resources = [DIRT,GRASS,WATER,BRICK,WOOD,SAND,PLANK]
```

- Name the resource `'plank'`.



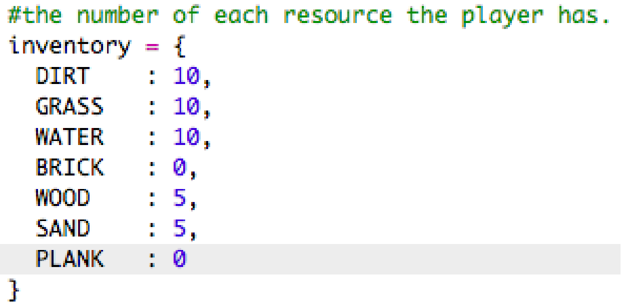
```
#the names of the resources.
names = {
  DIRT    : 'dirt',
  GRASS   : 'grass',
  WATER   : 'water',
  BRICK   : 'brick',
  WOOD    : 'wood',
  SAND    : 'sand',
  PLANK   : 'plank'
}
```

- Give your `PLANK` resource an image. The project already contains a `plank.gif` image, but you can create your own if you prefer.



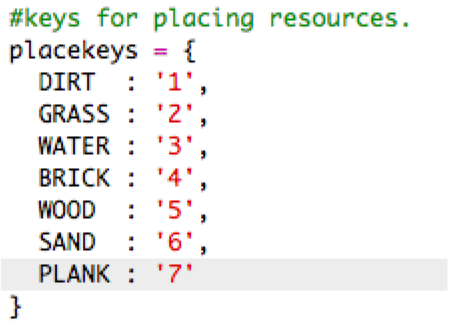
```
#a dictionary linking resources to images.
textures = {
  DIRT    : 'dirt.gif',
  GRASS   : 'grass.gif',
  WATER   : 'water.gif',
  BRICK   : 'brick.gif',
  WOOD    : 'wood.gif',
  SAND    : 'sand.gif',
  PLANK   : 'plank.gif'
}
```

- Add planks to your inventory.



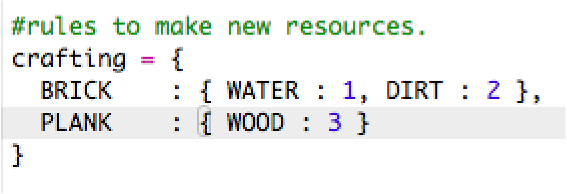
```
#the number of each resource the player has.
inventory = {
  DIRT    : 10,
  GRASS   : 10,
  WATER   : 10,
  BRICK   : 0,
  WOOD    : 5,
  SAND    : 5,
  PLANK   : 0
}
```

- Set a key for placing planks.



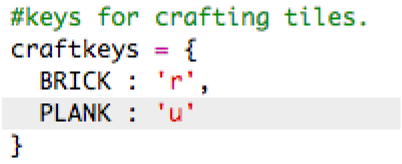
```
#keys for placing resources.
placekeys = {
  DIRT  : '1',
  GRASS : '2',
  WATER : '3',
  BRICK : '4',
  WOOD  : '5',
  SAND  : '6',
  PLANK : '7'
}
```

- As this resource can be crafted, you need to create a crafting rule, which is that a plank can be made from 3 wood tiles. Add this code to the `crafting` dictionary.



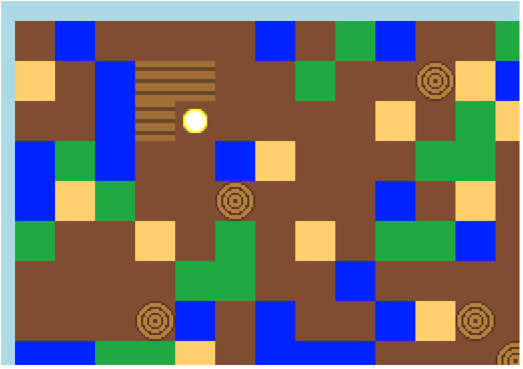
```
#rules to make new resources.
crafting = {
  BRICK    : { WATER : 1, DIRT : 2 },
  PLANK    : { WOOD : 3 }
}
```

- Finally, you need to set a key for crafting new planks.



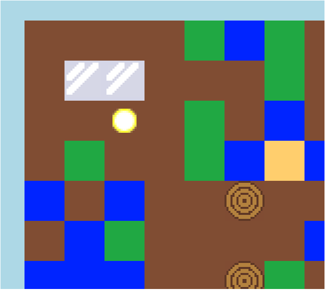
```
#keys for crafting tiles.
craftkeys = {
  BRICK : 'r',
  PLANK : 'u'
}
```

- To test your new plank resource, gather up a few wood tiles and then craft some planks from your wood. You can then place your new planks in your world.



Step 9 Challenge: Crafting glass from sand

Can you create a new glass resource, that can be crafted from sand? You can follow the steps above to help you.



The project already contains a `glass.gif` image that you can use, or you can create your own image if you prefer.

Step 10 Challenge: Create more resources

Can you add more resources and crafting rules to your game?