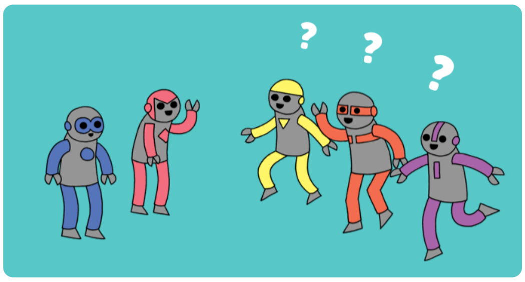
Projects

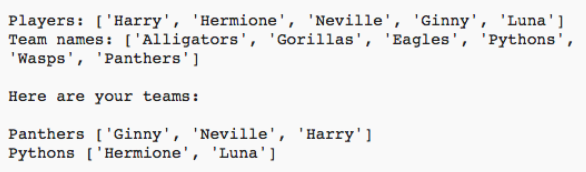
Team Chooser

Make a program to split players into 2 random teams.



Step 1 Introduction:

In this project, you’ll learn how to create 2 random teams from a list of players.



```
Players: ['Harry', 'Hermione', 'Neville', 'Ginny', 'Luna']
Team names: ['Alligators', 'Gorillas', 'Eagles', 'Pythons',
'Wasps', 'Panthers']

Here are your teams:

Panthers ['Ginny', 'Neville', 'Harry']
Pythons ['Hermione', 'Luna']
```

Additional information for club leaders

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

Club leader notes

Introduction:

In this project, children will learn how to make a program to split a list of players into 2 random teams. This project teaches lists and using files.

Online Resources

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

- **New (blank) Python Trinket – jumpto.cc/python-new** (http://jumpto.cc/python-new)

There is also a trinket containing the completed project:

- **'Team Chooser' Finished – trinket.io/python/a699c44ce6** (https://trinket.io/python/a699c44ce6)

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:

- team/team.py

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

- team-finished/team.py

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

Learning Objectives

- Lists;
- Loading list data from a file.

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

- **Use basic programming constructs to create simple programs.** (https://www.raspberrypi.org/curriculum/programming/creator)

Challenges

- "Adding more players" - adding elements to a `players` list;
- "Choosing for team B" - creating a new `teamB` list to add random players to;
- "Random team names" - creating and using a new `teamNames` list to assign random names to teams;
- "Storing team names" - storing team names in a file, and loading them into a `teamNames` variable;
- "More teams" - splitting players into 3 teams instead of 2.

Project materials

Project resources

- You can **find the resources for this project here** (http://rpf.io/p/en/team-chooser-go).
- **Online blank Python Trinket** (http://jumpto.cc/python-new)
- **Offline blank Python file** (https://projects-static.raspberrypi.org/projects/team-chooser/6d85ec006baf59e1e7799299fa72bd1d9ab2cd05/en/resources/new-new.py)

Club leader resources

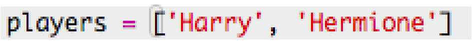
- You can **find the solutions for this project here** (http://rpf.io/p/en/team-chooser-get).
- **Online completed Trinket project** (https://trinket.io/python/a699c44ce6)
- **team-chooser-finished/team-chooser.py** (https://projects-static.raspberrypi.org/projects/team-chooser/6d85ec006baf59e1e7799299fa72bd1d9ab2cd05/en/resources/team-chooser-finished-team-chooser.py)

Step 2 Players

Let’s start by creating a list of players to choose from.

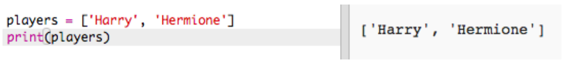
- Open the blank Python template Trinket: **jumpto.cc/python-new** (http://jumpto.cc/python-new).
- You can use a variable to store a **list** of players. The list should be in square brackets `[ ]`, with a comma between each item in the list.

  Start by adding a list of players to your program.



```
players = ['Harry', 'Hermione']
```

- Add this code to print your `players` variable:

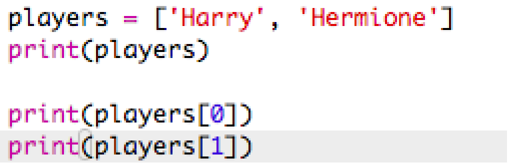


```
players = ['Harry', 'Hermione']
print(players)
```

```
['Harry', 'Hermione']
```

- You can get to an item in the list by adding its position in square brackets after the variable name.

  The first item in the list is at **position 0**. This is different to Scratch, which starts at position 1.



```
players = ['Harry', 'Hermione']
print(players)

print(players[0])
print(players[1])
```

Step 3 Challenge: Adding more players

Can you add more players to your list? You can add as many players as you like, but make sure that there is an **even** number of players.

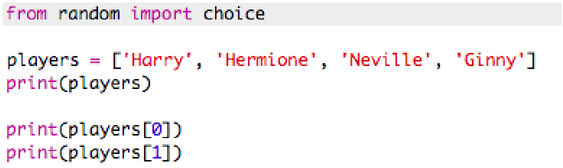
You can also change the names of the first 2 players if you prefer.

Can you add code to print **just one** of your new players?

Step 4 Random players

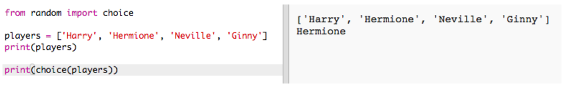
Let’s choose random players!

- To be able to get a random player from your `players` list, first you’ll need to import the `choice` part of the `random` module.



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

print(players[0])
print(players[1])
```

- To get a random player, you can use `choice`. (You can also delete the code to print individual players.)

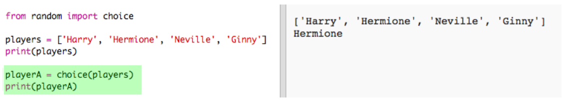


```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

print(choice(players))
```

```
['Harry', 'Hermione', 'Neville', 'Ginny']
Hermione
```

- Test your `choice` code a few times and you should see a different player being chosen each time.

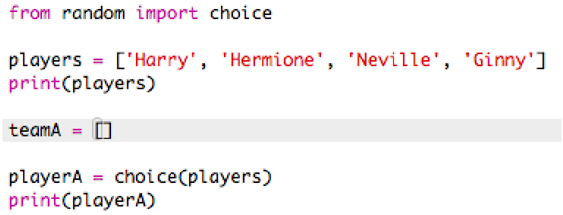
- You can also create a new variable called `playerA`, and use it to store your random player.



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

playerA = choice(players)
print(playerA)
```

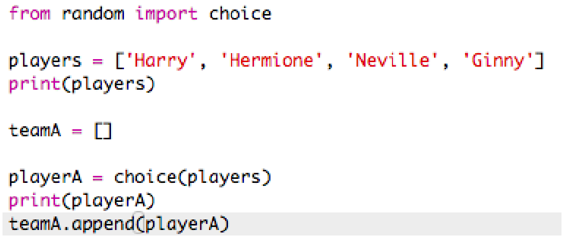
```
['Harry', 'Hermione', 'Neville', 'Ginny']
Hermione
```

- You’ll need a new list to store all of the players in team A. To start with, this list should be empty.



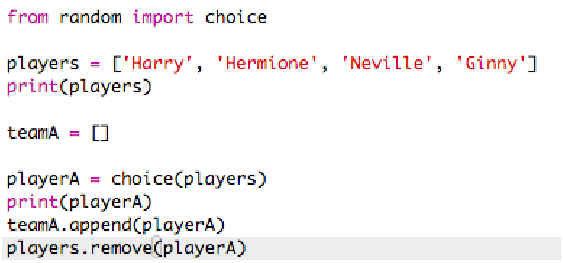
```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []

playerA = choice(players)
print(playerA)
```

- You can now add your randomly chosen player to `teamA`. To do this, you can use `teamA.append` (**append** means add to the end).



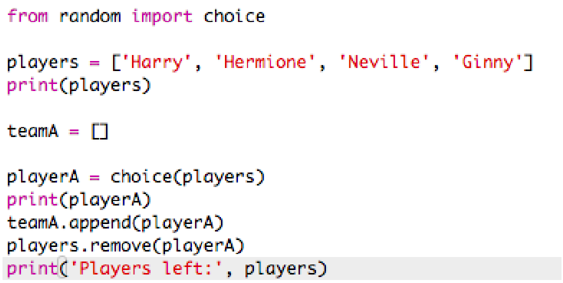
```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []

playerA = choice(players)
print(playerA)
teamA.append(playerA)
```

- Now that your player has been chosen, you can remove them from your list of `players`.



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []

playerA = choice(players)
print(playerA)
teamA.append(playerA)
players.remove(playerA)
```

- Test this code by adding a `print` command, to show the `players` left to choose from.



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []

playerA = choice(players)
print(playerA)
teamA.append(playerA)
players.remove(playerA)
print('Players left:', players)
```

In the example above, Hermione has been chosen for `teamA`, and so has been removed from the list of `players`.

Step 5 Challenge: Choosing for team B

Can you add code to choose a player at random for team B? You'll need to:

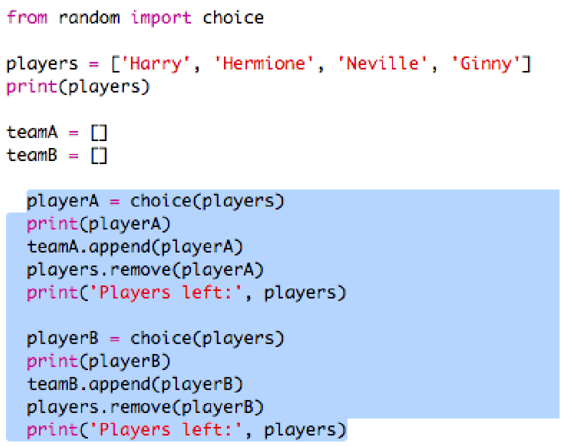
- Create a new `teamB` list
- Choose a random player for team B (called `playerB`)
- `append` the chosen player to your `teamB` list
- `remove` the chosen player from your list of `players`

The code you'll need for `teamB` will be **very** similar to the code you've already written for `teamA`!

Step 6 Choosing lots of players

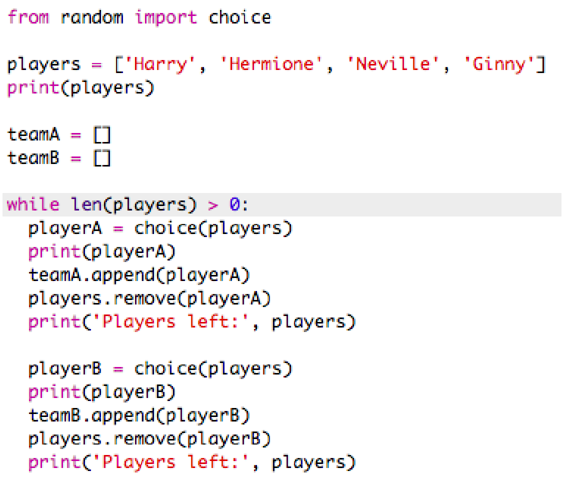
Next you'll need to make sure that every player has been chosen for a team.

- Highlight your code for choosing players for team A and team B and press the tab key to indent the code.



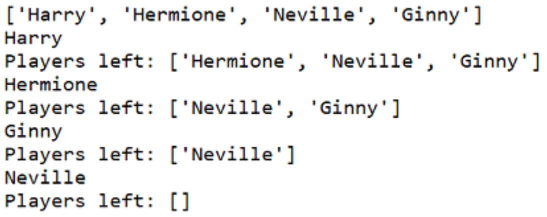
```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []
teamB = []

  playerA = choice(players)
  print(playerA)
  teamA.append(playerA)
  players.remove(playerA)
  print('Players left:', players)

  playerB = choice(players)
  print(playerB)
  teamB.append(playerB)
  players.remove(playerB)
  print('Players left:', players)
```

- Add a **while** loop to keep choosing players until the length of the `players` list is 0.



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []
teamB = []

while len(players) > 0:
  playerA = choice(players)
  print(playerA)
  teamA.append(playerA)
  players.remove(playerA)
  print('Players left:', players)

  playerB = choice(players)
  print(playerB)
  teamB.append(playerB)
  players.remove(playerB)
  print('Players left:', players)
```

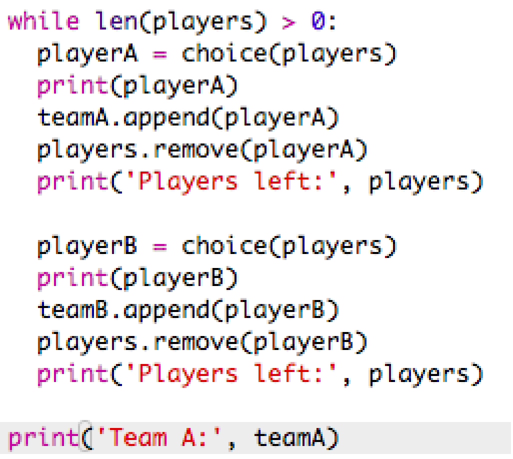
- Run your code to test it. You should see players being chosen for team A and team B until there are no more players left.



```
['Harry', 'Hermione', 'Neville', 'Ginny']
Harry
Players left: ['Hermione', 'Neville', 'Ginny']
Hermione
Players left: ['Neville', 'Ginny']
Ginny
Players left: ['Neville']
Neville
Players left: []
```

- Add code to print your `teamA` list **after** your `while` loop (making sure it is not indented).

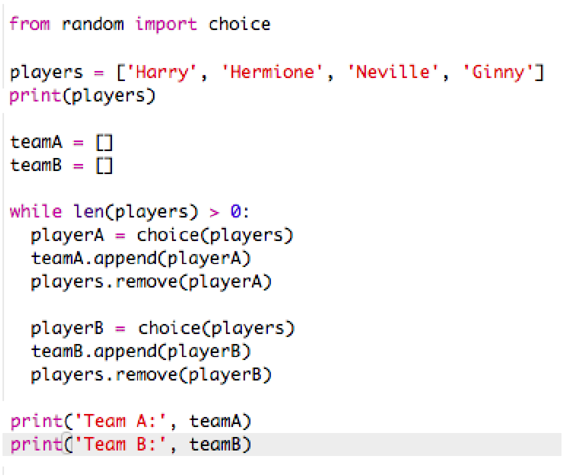
  This means that `teamA` will only be printed once, after all the players have been chosen.



```
while len(players) > 0:
  playerA = choice(players)
  print(playerA)
  teamA.append(playerA)
  players.remove(playerA)
  print('Players left:', players)

  playerB = choice(players)
  print(playerB)
  teamB.append(playerB)
  players.remove(playerB)
  print('Players left:', players)

print('Team A:', teamA)
```

- You can do the same for `teamB`, and you can also delete the other print commands, as they were only there to test your code.

  Here's how your code should look:



```
from random import choice

players = ['Harry', 'Hermione', 'Neville', 'Ginny']
print(players)

teamA = []
teamB = []

while len(players) > 0:
  playerA = choice(players)
  teamA.append(playerA)
  players.remove(playerA)

  playerB = choice(players)
  teamB.append(playerB)
  players.remove(playerB)

print('Team A:', teamA)
print('Team B:', teamB)
```

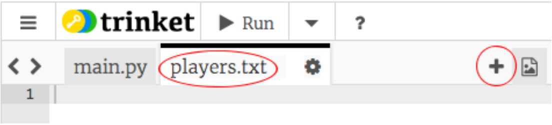
- Test your code again and you should just see your list of players as well as your final teams.

```
['Harry', 'Hermione', 'Neville', 'Ginny']
Team A: ['Hermione', 'Ginny']
Team B: ['Harry', 'Neville']
```

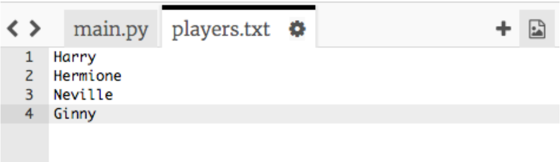
Step 7 Files

You can use a file to store your list of players.

- Click the + icon and create a new file called `players.txt`.

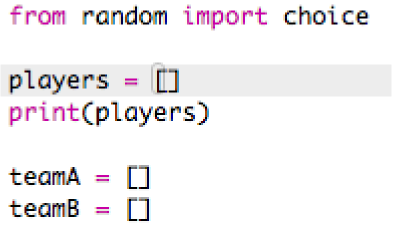


- Add your players to your new file. Make sure that there is no blank line after your last player.



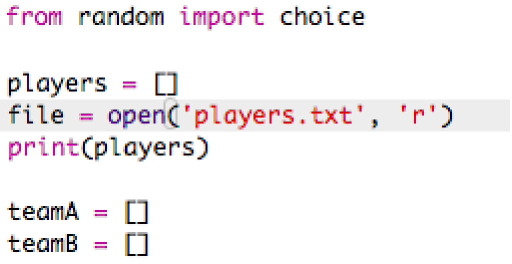
```
Harry
Hermione
Neville
Ginny
```

- Change your `players` list so that it is empty.



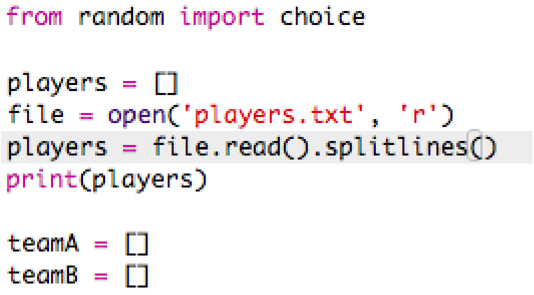
```
from random import choice

players = []
print(players)

teamA = []
teamB = []
```

- Open your `players.txt` file (the `'r'` means read-only).



```
from random import choice

players = []
file = open('players.txt', 'r')
print(players)

teamA = []
teamB = []
```

- Read the list from the file and add to your `players` list. (The `splitlines` code means that every line in the file is a new item in the `players` list).



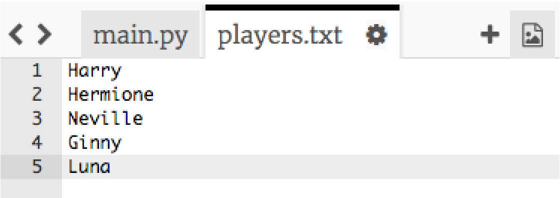
```
from random import choice

players = []
file = open('players.txt', 'r')
players = file.read().splitlines()
print(players)

teamA = []
teamB = []
```

- If you test your code, it should work exactly the same as before. However, now it’s much easier to add players to your `players.txt` file.

Step 8 Odd players

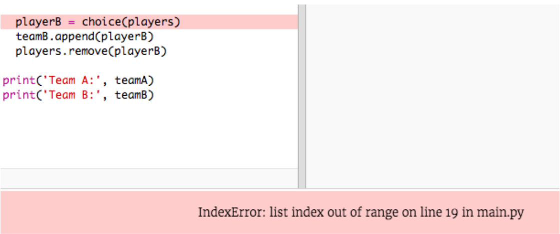
Let’s improve your program to work with an odd number of players.

- Add another name to your `players.txt` list, so that you have an odd number of players.



```
Harry
Hermione
Neville
Ginny
Luna
```

- If you test your code, you’ll see that you get an error message.

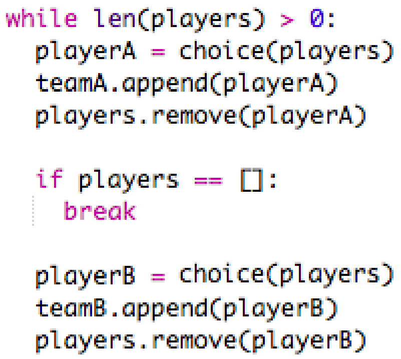


```
  playerB = choice(players)
  teamB.append(playerB)
  players.remove(playerB)

print('Team A:', teamA)
print('Team B:', teamB)
```

```
IndexError: list index out of range on line 19 in main.py
```

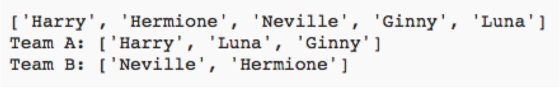
- The error is because your program keeps choosing random players for team A and then team B. However, if there is an odd number of players then after choosing a player for team A there are no players left to choose from for team B.

  To fix this bug, you can tell your program to `break` out of your `while` loop if your `players` list is empty.



```
while len(players) > 0:
  playerA = choice(players)
  teamA.append(playerA)
  players.remove(playerA)

  if players == []:
    break

  playerB = choice(players)
  teamB.append(playerB)
  players.remove(playerB)
```

- If you test your code again, you should see that it now works with an odd number of players.



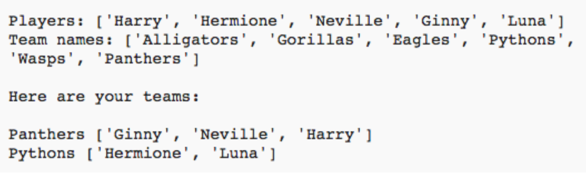
```
['Harry', 'Hermione', 'Neville', 'Ginny', 'Luna']
Team A: ['Harry', 'Luna', 'Ginny']
Team B: ['Neville', 'Hermione']
```

Step 9 Challenge: Random team names

Can you give both of your teams a random team name?

You can create a list called `teamNames` containing the names to choose from.

You can then choose (and display) a random name for each team.



```
Players: ['Harry', 'Hermione', 'Neville', 'Ginny', 'Luna']
Team names: ['Alligators', 'Gorillas', 'Eagles', 'Pythons',
'Wasps', 'Panthers']

Here are your teams:

Panthers ['Ginny', 'Neville', 'Harry']
Pythons ['Hermione', 'Luna']
```

Step 10 Challenge: Storing team names

Can you store your list of team names in a file?

Step 11 Challenge: More teams

Can you improve your program to split players into 3 teams instead of 2?