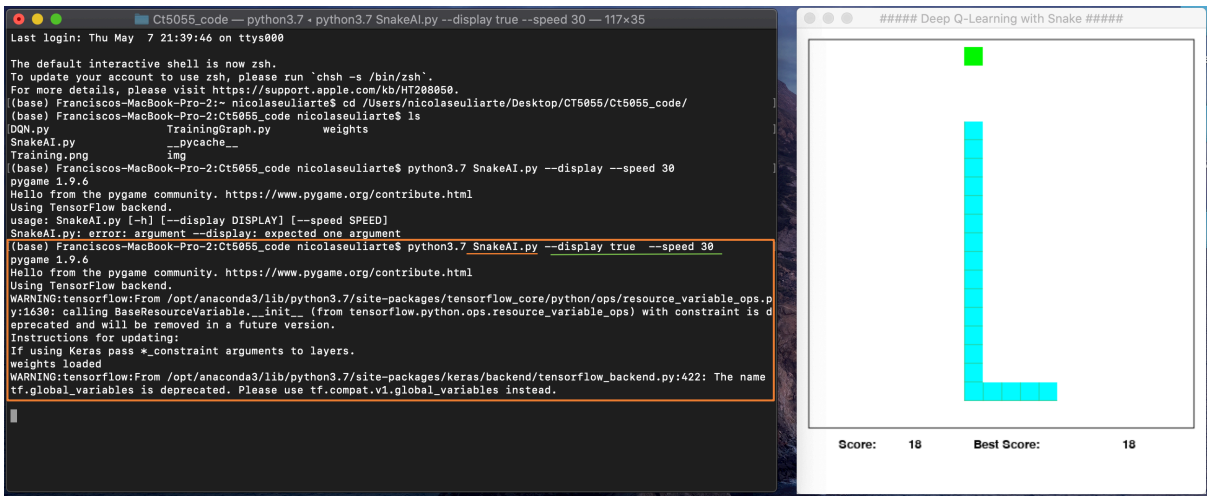
Running

Run the ‘SnakeAI’ class and the user will be greeted with the snake class with visuals of the current and best scores alongside the display if applicable. The snakeAI.py can also be ran through terminal as well:

- MacOS using the command:
  - python3.7 SnakeAI.py
- Windows using the command:
  - Python SnakeAI.py
- Ubuntu using the command:
  - Python3 SnakeAI.py

Here the user is able to add in the display and options as well for the application on start:

- MacOS using the command:
  - python3.7 SnakeAI.py - - display true - - speed 30
- Windows using the command:
  - Python SnakeAI.py - - display true - - speed 30
- Ubuntu using the command:
  - Python3 SnakeAI.py - - display true - - speed 30



Loading Weights or Training

Assure the 'load_weights' is true, and 'train' is false in the 'SnakeAI' class. The opposite is done when training.

```
# Parameters Used
def par_def():
    # parameters dictionary
    par = dict()

    # Change between for training or trained
    par['load_weights'] = True
    par['train'] = False

    par['epsilon_decay_linear'] = 1 / 75
    par['learning_rate'] = 0.0005

    # attempted layer sizing numbers
    # 100, 150, 200
    par['first_layer_size'] = 200  # neurons in the first layer
    par['second_layer_size'] = 150  # neurons in the second layer
    par['third_layer_size'] = 100  # neurons in the third layer

    # number of games, memory and batch sizing
    #     # attempted sizes
    #     # episodes 400, 1000, 2000
    #     # memory 3000, 2500, 4000
    #     # batch sizes 500, 550, 600, 650

    # reduce for the actual weights loading for an easier visualisation
    # 400-500 minimum when training
    par['episodes'] = 400
    par['memory_size'] = 2500   # switch to 4000 when training to see what may occur
    par['batch_size'] = 500

    # model for trained
    par['weights_path'] = 'weights/weights.hdf5'

    return par
```

To display the snake, assure that the '—display' is set to True and that the '—speed' is set to 20-30. For the training it is advised to keep the display hidden and the speed as 0 for faster training sessions.

```
if __name__ == '__main__':
    # font initialising for the game display
    pygame.font.init()

    # arguments passing
    pas = argparse.ArgumentParser()
    par = par_def()

    # set options to activate or deactivate the game view, and its speed
    # use these to change speed and display options
    # 50 is a regular speed 20 is the visual training speed
    pas.add_argument("--display", type=bool, default=False)
    pas.add_argument("--speed", type=int, default=0)

    args = pas.parse_args()

    start(args.display, args.speed, par)
```