CSc 110 Scorekeeper saving and loading

Benjamin Dicken

Score Keeper

- What if we wanted a program that we could use to keep track of player scoring during a sporting event
  - Each time points were scored, associated the player with the points scored
  - Be able to check how many points a player has scored

- What are the challenges?

Score Keeper

```
Enter command: ADD Booker 3
Enter command: ADD James 2
Enter command: ADD Ayton 2
Enter command: ADD Booker 2
Enter command: ADD Booker 2
Enter command: GET Booker
Booker has 7 points.
Enter command: GET James
James has 2 points.
Enter command: GET Ayton
Ayton has 2 points.
Enter command: EXIT
(show scores)
```

```
def get_command():
    user_input = input('Cmd: ')
    return user_input.split(' ')

def get_index(players, points, player):
    if player not in players:
        players.append(player)
        points.append(0)
    return players.index(player)

def show_scores(players, points):
    for i in range(len(players)):
        print(players[i] + ': ' + points[i])
```

```
def main():
    players = []
    points = []
    while True:
        command = get_command()
        command_type = command[0]
        if command_type == 'ADD':
            index = get_index(players, points, command[1])
            points[index] += int(command[2])
        elif command_type == 'GET':
            index = get_index(players, points, command[1])
            print(command[1], 'has', points[index], 'points.')
        elif command_type == 'EXIT':
            return
        else:
            print('Huh?')


main()
```

Weaknesses

- Does not SAVE the game log info after the program ends
- What if something is entered incorrectly?
  - Is there a way to fix?

Weaknesses

- Does not SAVE the game log info after the program ends
- What if something is entered incorrectly?
    - Is there a way to fix?

Can we use file I/O to resolve this?

```
def load_points_file(players, points):
    # implement!

def main():
    players = []
    points = []
    load_points_file(players, points)
    while True:
        command = get_command()
        command_type = command[0]
        # The code where we handle the command types
        # . . .
```

Implement load_points_file

```
def load_points_file(players, points):
    '''
    This function should ask the user for the name of a file to load,
    then read the file and load the info into the players and points
    lists. You can assume the file will have one line per player.
    Each line will have the player name, and the the points that
    player has scored. For example:

    Booker 12
    James 20
    Jackson 7

    '''
```

Implement load_points_file

```
def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
```

Implement load_points_file

```
def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
    for line in points_file:
        line_split = line.split(' ')
        players.append( line_split[0] )
        points.append( line_split[1] )
```

Implement load_points_file

```
def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
    for line in points_file:
        line_split = line.split(' ')
        players.append( line_split[0] )
        points.append( int(line_split[1]) )
```

```
def get_command():
    user_input = input('Cmd: ')
    return user_input.split(' ')

def get_index(players, points, player):
    if player not in players:
        players.append(player)
        points.append(0)
    return players.index(player)

def show_scores(players, points):
    for i in range(len(players)):
        print(players[i] + ': ' + points[i])

def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
    for line in points_file:
        line_split = line.split(' ')
        players.append( line_split[0] )
        points.append( int(line_split[1]) )
```

```
def main():
    players = []
    points = []
    load_points_file(players, points)
    while True:
        command = get_command()
        command_type = command[0]
        if command_type == 'ADD':
            index = get_index(players, points, command[1])
            points[index] += int(command[2])
        elif command_type == 'GET':
            index = get_index(players, points, command[1])
            print(command[1], 'has', points[index], 'points.')
        elif command_type == 'EXIT':
            return
        else:
            print('Huh?')

main()
```

Score Keeper

- What if we wanted a program that we could use to keep track of player scoring during a sporting event
  - Each time points were scored, associated the player with the points scored
  - Be able to check how many points a player has scored

- What are the challenges?

```
def get_command():
    user_input = input('Cmd: ')
    return user_input.split(' ')

def get_index(players, points, player):
    if player not in players:
        players.append(player)
        points.append(0)
    return players.index(player)

def show_scores(players, points):
    for i in range(len(players)):
        print(players[i] + ': ' + points[i])

def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
    for line in points_file:
        line_split = line.split(' ')
        players.append( line_split[0] )
        points.append( int(line_split[1]) )
```

```
def main():
    players = []
    points = []
    load_points_file(players, points)
    while True:
        command = get_command()
        command_type = command[0]
        if command_type == 'ADD':
            index = get_index(players, points, command[1])
            points[index] += int(command[2])
        elif command_type == 'GET':
            index = get_index(players, points, command[1])
            print(command[1], 'has', points[index], 'points.')
        elif command_type == 'EXIT':
            return
        else:
            print('Huh?')

main()
```

```
def save_points_file(players, points):
    # implement!

def main():
    players = []
    points = []
    load_points_file(players, points)
    while True:
        command = get_command()
        command_type = command[0]
        # . . .
        elif command_type == 'EXIT':
            save_points_file(players, points)
            break
        # . . .
```

Implement save_points_file

```
def save_points_file(players, points):
    '''
    This function should ask the user for the name of a file to save
    to, then save the points info to the file.
    It should write the output in the same format that the
    save_points_file function expects. For example:

    Booker 12
    James 20
    Jackson 7

    '''
```

Implement save_points_file

```
def save_points_file(players, points):
    # ?
```

Implement save_points_file

```
def save_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'w')

    # ?

    points_file.close()
```

Implement save_points_file

```
def save_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'w')
    for i in range(len(players)):
        points_file.write(players[i] + ' ' +  str(points[i]))
        points_file.write('\n')
    points_file.close()
```

```
def get_command():
    user_input = input('Cmd: ')
    return user_input.split(' ')

def get_index(players, points, player):
    if player not in players:
        players.append(player)
        points.append(0)
    return players.index(player)

def show_scores(players, points):
    for i in range(len(players)):
        print(players[i] + ': ' + str(points[i]))

def load_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'r')
    for line in points_file:
        line_split = line.split(' ')
        players.append( line_split[0] )
        points.append( int(line_split[1]) )

def save_points_file(players, points):
    file_name = input('Enter points file name: ')
    points_file = open(file_name, 'w')
    for i in range(len(players)):
        points_file.write(players[i] + ' ' + \
                          str(points[i]))
        points_file.write('\n')
    points_file.close()
```

```
def main():
    players = []
    points = []
    load_points_file(players, points)
    while True:
        command = get_command()
        command_type = command[0]
        if command_type == 'ADD':
            index = get_index(players, points, command[1])
            points[index] += int(command[2])
        elif command_type == 'GET':
            index = get_index(players, points, command[1])
            print(command[1], 'has', points[index], 'points.')
        elif command_type == 'EXIT':
            save_points_file(players, points)
            show_scores(players, points)
            return
        else:
            print('Huh?')
```