PokemonGo Report

MSIS 615 Group Project

Chris ParlowAndy QUALCHChristian Thaxton

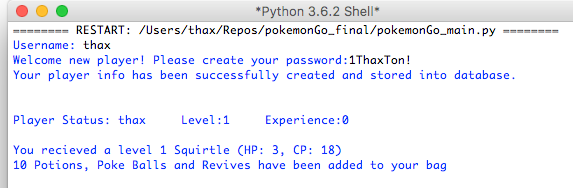
2017

Pokémon Go Manual

# Introduction

Pokémon Go is a Python application based off the mobile PokemonGo Application. In this app, a player can interact with Pokémon. A player can encounter wild Pokémon to battle, or they can choose to capture wild Pokémon to use them in a later battle. Some Pokémon are evolvable, and Pokémon evolve through gaining battle experience. Players can take care of their Pokémon by giving them potions to heal or revive the Pokémon. Players can also gain experience and increase their level. Finally, players can exit the game and save their progress to return to their game at a later time.

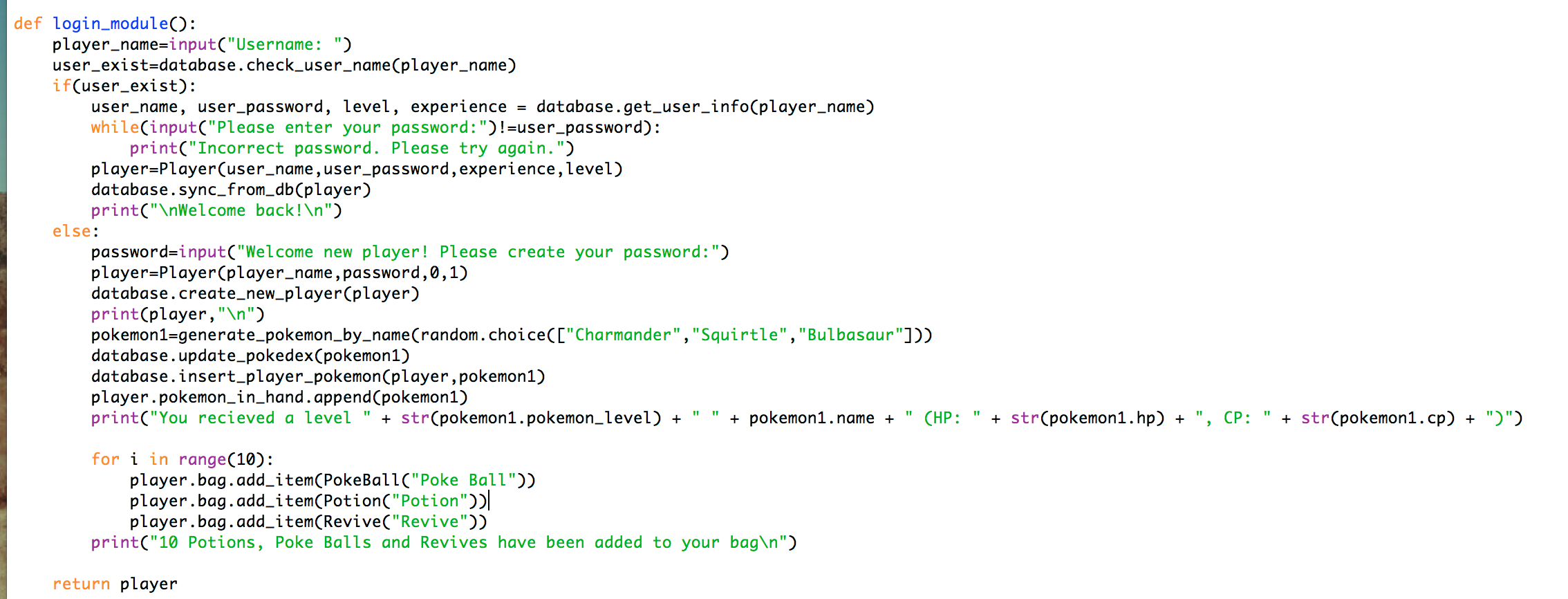
# Player Account



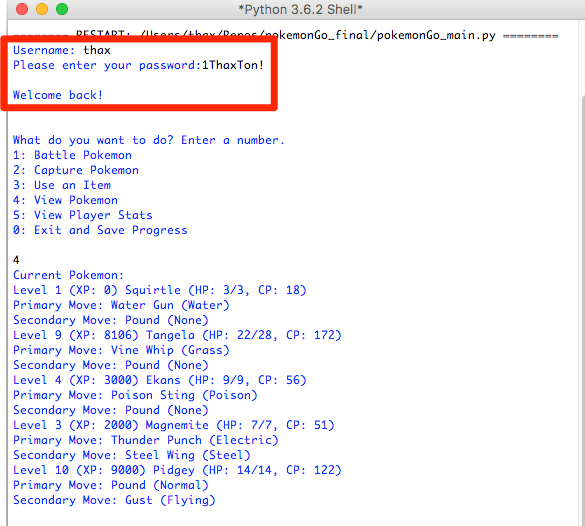
Create an account

1. Enter a username
2. Enter a password. It must contain the following:
   1. Minimum of 8 characters
   2. At least 1 uppercase letter
   3. At least 1 lowercase letter
   4. At least 1 number
   5. At least 1 symbol

Once your account has been created, you will receive a player status message. This will alert you of your username, level, and experience. A player will also be notified if a random starter Pokémon has been given to them, and the player will receive 10 pokeballs, potions, and revives to start.

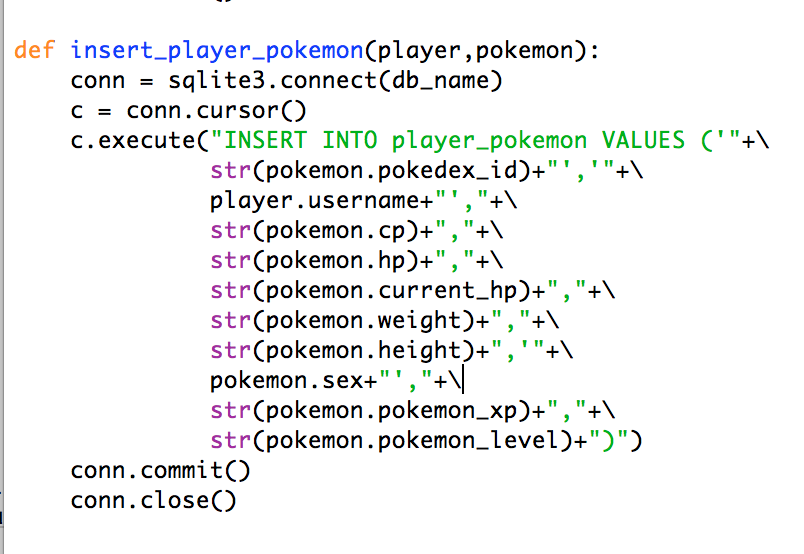


If you’ve previously created an account, enter your username and password, and it will resume where you ended the game.



Prior Pokemon captured is retrieved from the database and Pokemon are instantiated again using data that was written from the CSV into the database when the player played previously.



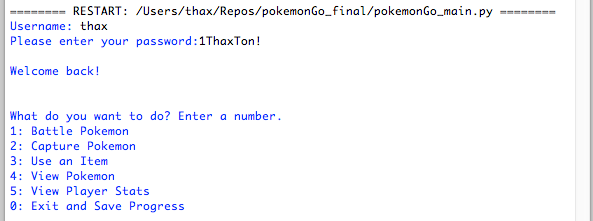




# Main Player Menu

Select from the following options by entering the number associated with each:

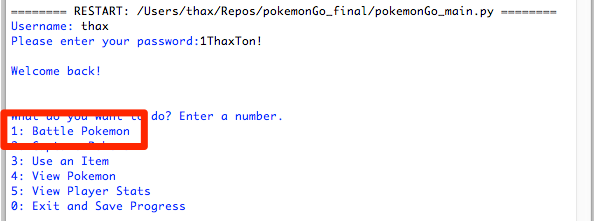
1. Battle Pokémon
2. Capture Pokémon
3. Use an Item
4. View Pokémon
5. View Player Status



## Pokémon Battle

Once you’ve decided to battle a Pokémon (option 1), you will receive the following alert:

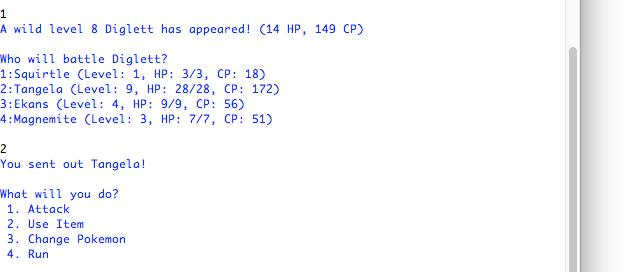
* “A wild [Pokémon] has appeared! (HP, CP)”
* “Who will battle [Pokémon]?”

At this point, the player will then be shown a list of available Pokémon to battle the wild Pokémon. Again, select a Pokémon to battle the wild Pokémon by entering a number. 

NOTE: Certain Pokémon types are super effective or ineffective against other types of Pokémon. This will lead to a damage multiplier. Be Cautious of this when selecting a Pokémon to battle.

### Battle Menu – What will you do?

After the player has selected a Pokémon to fight in the battle, the player needs to specify how to proceed.



1. Attack
   * Engage and attack the wild Pokémon
2. Use Item
   * Select an item from your bag to use during the battle
3. Change Pokémon
   * Select a different Pokémon to attack with.
4. Run
   * Run from the wild Pokémon and do not battle it.

#### Attack

By selecting option 1, the player decides to attack the wild Pokémon. The player can accomplish several things by attacking a wild Pokémon.

1. Battle the Pokémon until it’s been killed
2. Battle the Pokémon to level up captured Pokémon

Each Pokémon has 2 moves. The player must select which move to attack with first. After selecting a move, a battle sequence will occur.

1. The player’s Pokémon will attack. It will use the selected move, and the app will alert the player how much damage the move dealt. The game will also show the remaining HP of the wild Pokémon.
2. The wild Pokémon will attack immediately after the player’s attack. The player will see a message saying what move was used, how much damage was dealt, and how much HP the player’s Pokémon has remaining.
3. After the sequence, the player will again see the battle menu. The battle sequence will continue until one of the Pokémon has been defeated, or the player selects a different option.

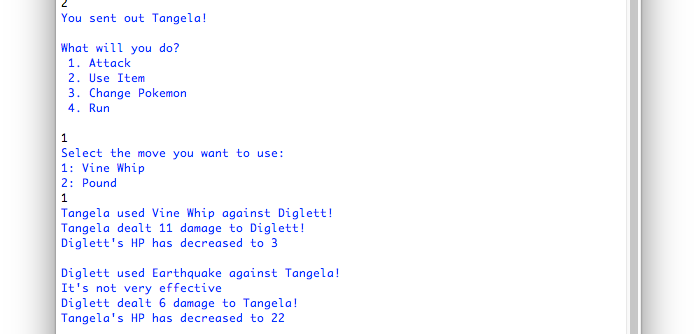
Once one of the Pokémon in the battle reaches 0 HP, the player will receive another message. It will say:

“[pokemon1] has been defeated”

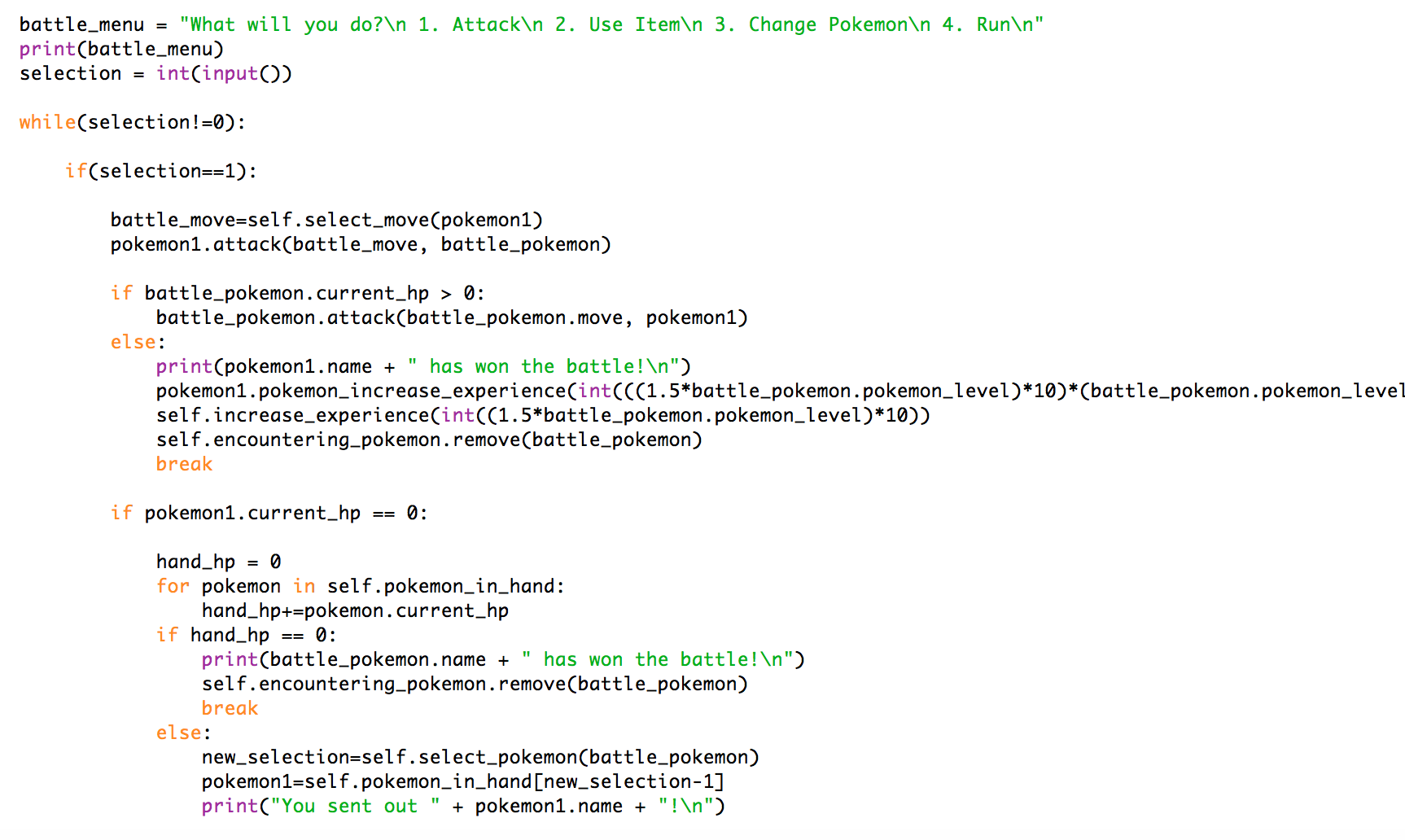
“[pokemon2] has won the battle!”

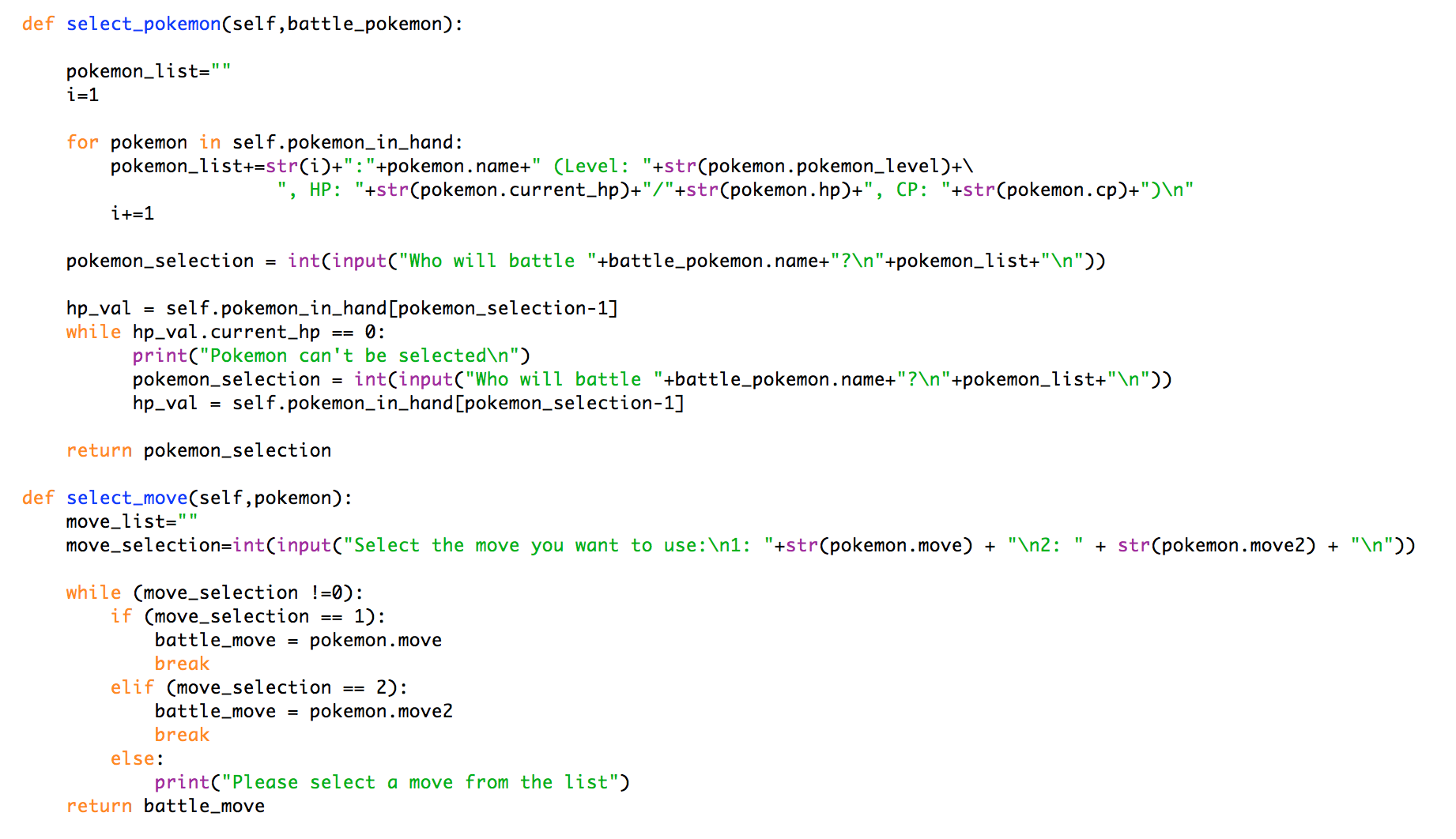
* At this point, the game returns the player to the main menu to perform another action from above.

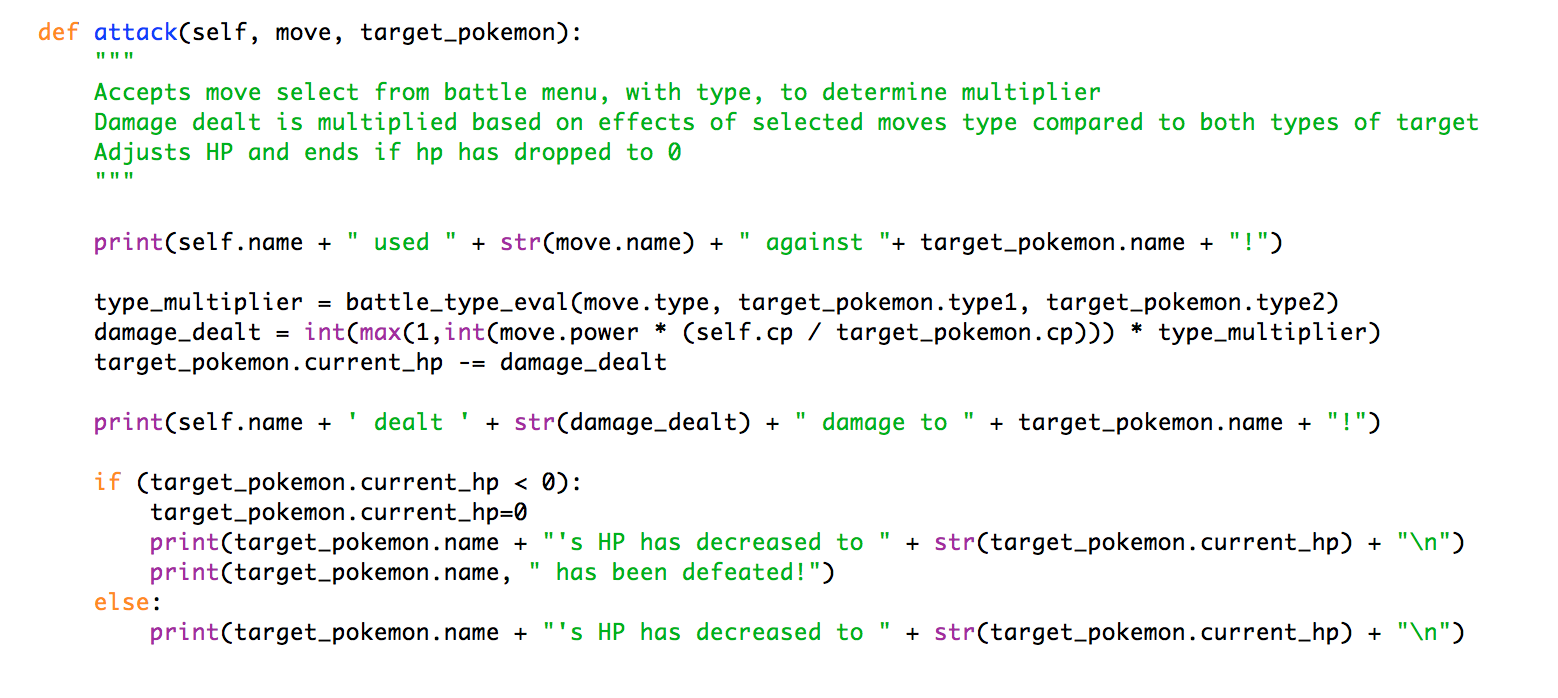
If the player’s Pokémon wins the battle, the Pokémon will gain experience. The player will also gain experience form winning a battle.

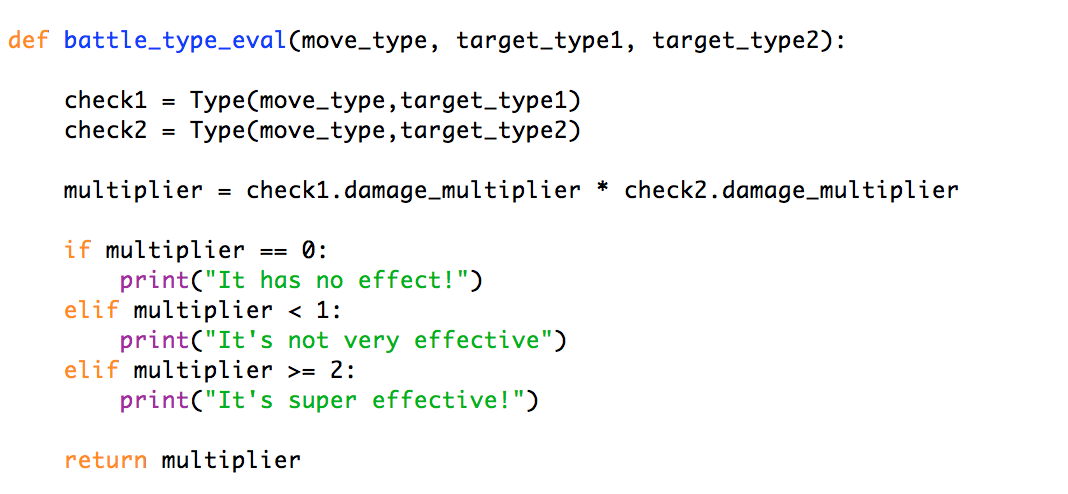


The following dictates how the attack is generated and how the type comparison is made.







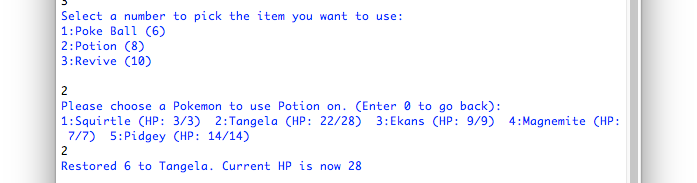




#### Use Item

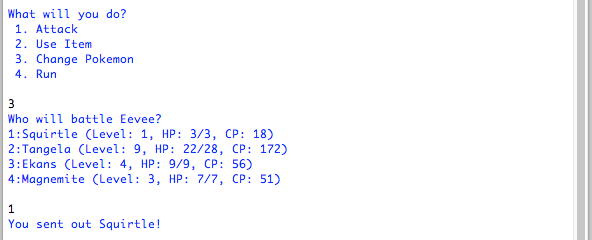
In the course of a battle, you may want to use an Item. There can be several reasons for needing to use an item.

* Revive a defeated Pokémon
* Potion to heal a wounded Pokémon
* PokeBall to capture wild Pokémon



#### Change Pokémon

During the course of a battle, you may decide to change Pokémon. Select this option to recall the current fighting Pokémon and send out a different Pokémon. This is handled using the select\_pokemon method detailed above.

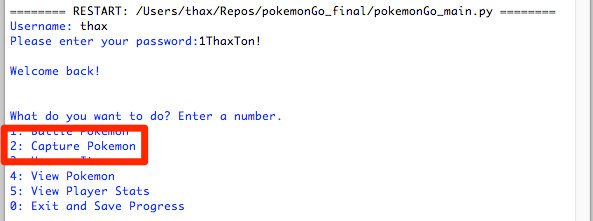


#### Run

Finally, a player may choose to run from a battle. Doing so will exit the battle sequence and return the player to the main menu.



## Capture Pokémon



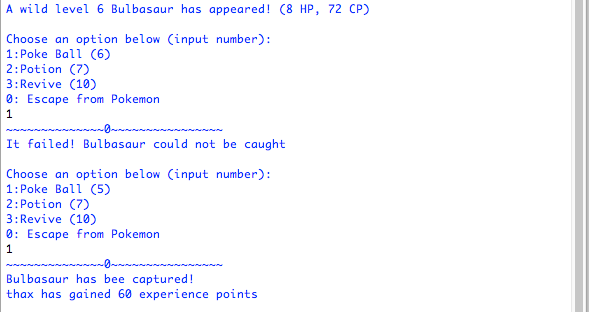
From the main menu, the player may also decide to try to capture a wild Pokémon (option 2). Once selected, the player will receive a similar message as the battle method:

* “a wild [level] [Pokémon] has appeared! (HP, CP)”

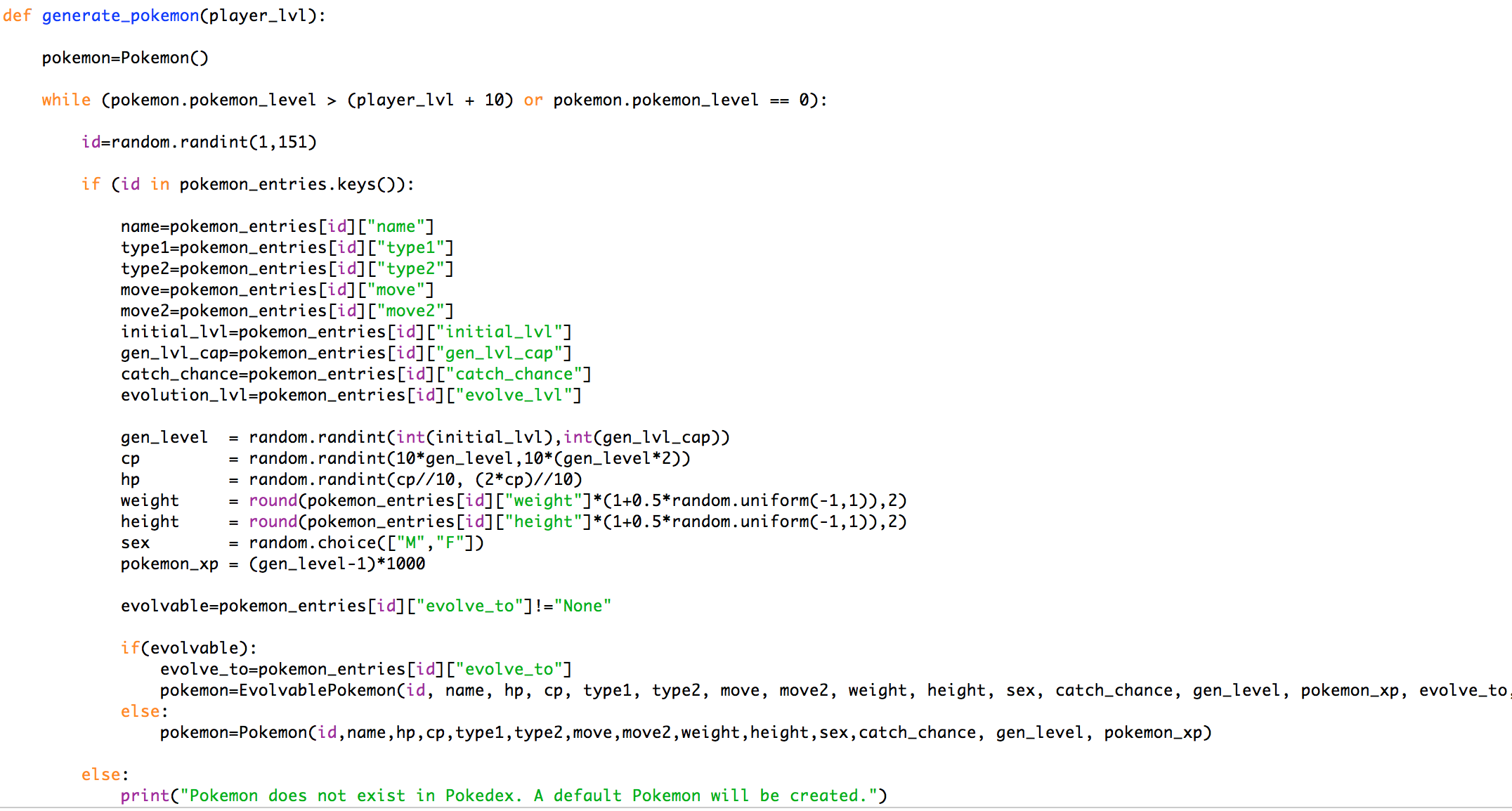
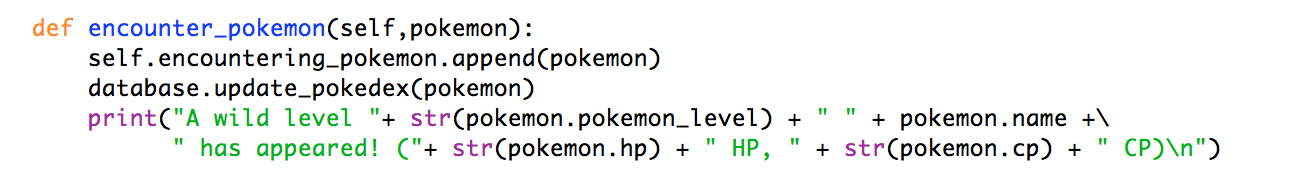
the player is then prompted to select an item to use. To capture the Pokémon, select a pokeball. If the Pokémon was captured, the game will alert the player:

* “[Pokémon] has been captured!”

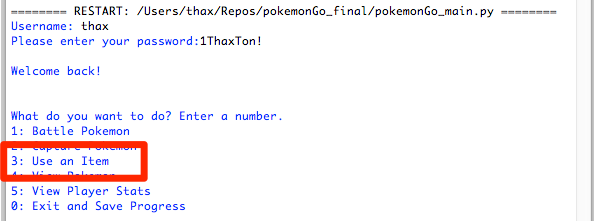
After capturing a Pokémon, the player will also receive experience points.



Both in battles and capturing, a wild Pokemon is generated using the same function, which generates Pokemon based on the player’s current level (can not be generated above the player’s level plus 10). Each Pokemon has a specific range they can be generated within, which allows for player’s to see greater variety as they play more but also make battle’s fair early on.



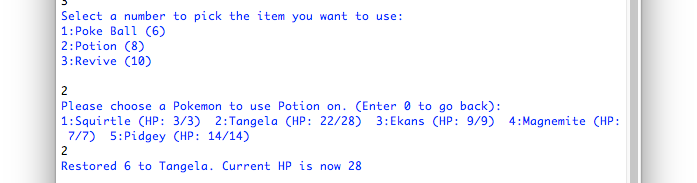
## Use an Item



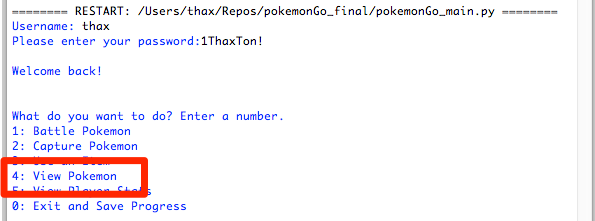
If the player selects to use an item, a list of items in the player’s bag will be displayed.

1. The player needs to select an item to use.
2. If applicable, the player must then select a Pokémon on which to use the item
3. For potions, revives, etc., the player will receive a message stating how many HP has been restored, and what the Pokémon’s HP is after using the potion or other item on the Pokémon.

The game then returns to the main menu.

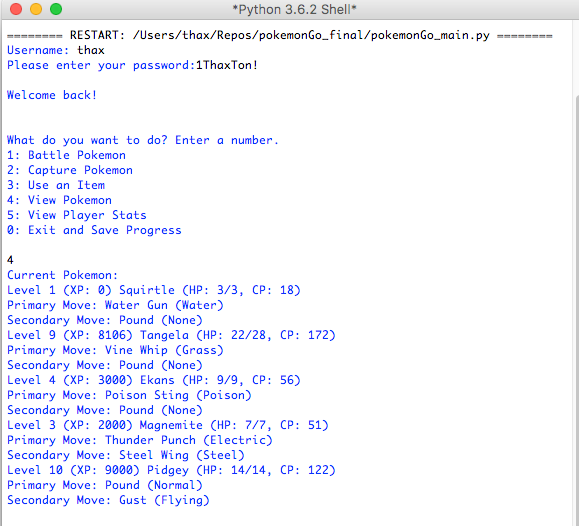


## View Pokémon

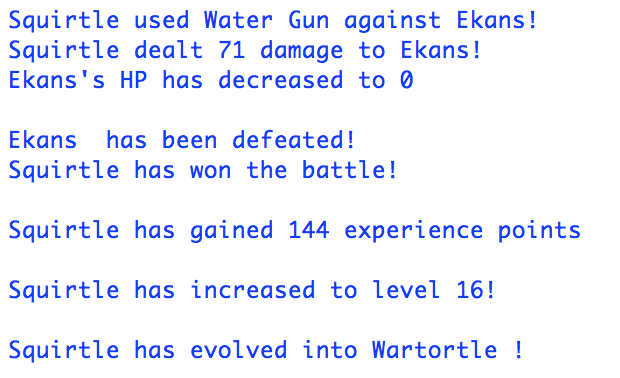


Select this option to view the Pokémon you’ve previously captured. This option displays useful information on your Pokémon, such as:

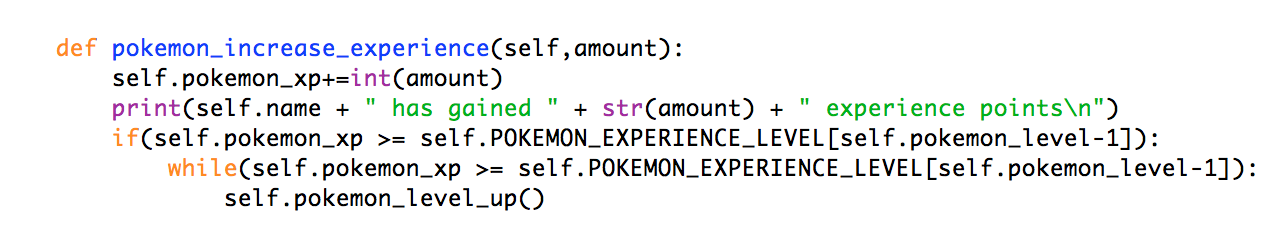
* Pokémon name
* Pokémon Level
* Pokémon experience
* Pokémon HP
* Pokémon CP
* Pokémon moves

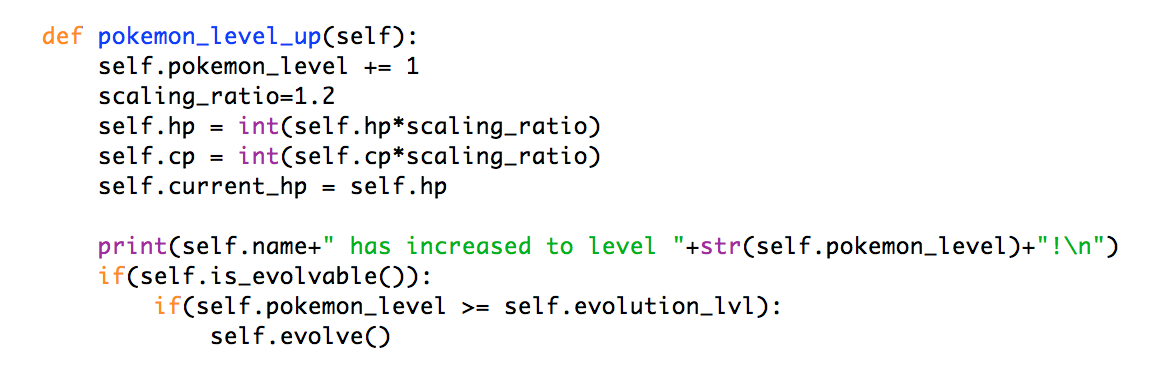


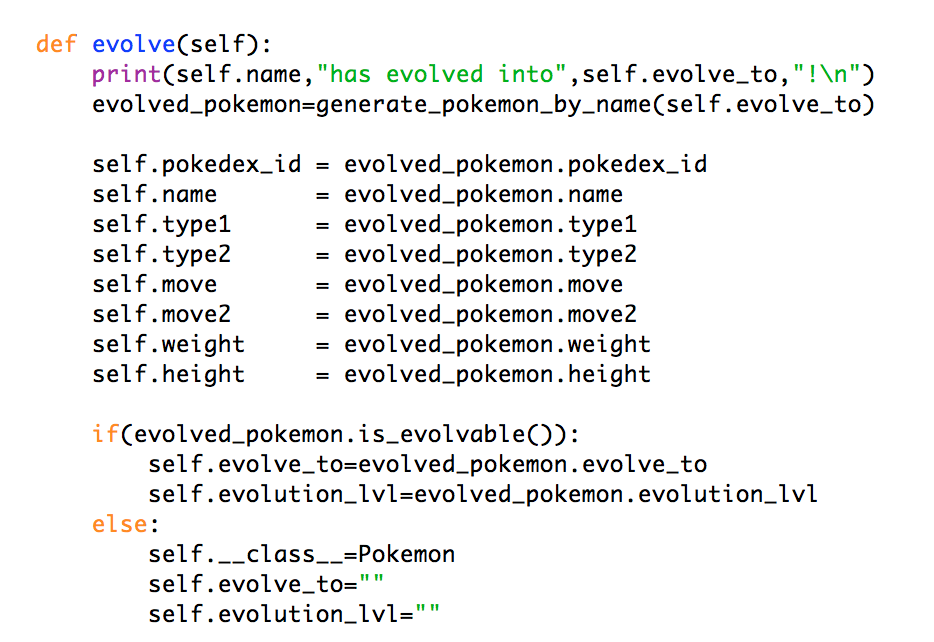
Certain Pokémon can evolve. To evolve a Pokémon, they must gain experience and increase their level. Each Pokémon has a distinct level at which they evolve.



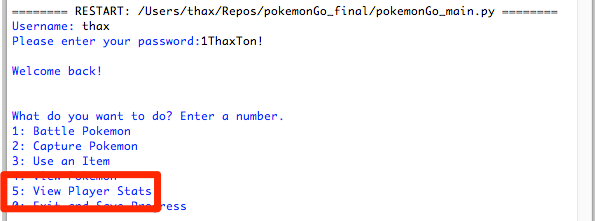
Upon adding experience, in addition to checking if the Pokemon has increased enough experience to reach the next level, it will check if their preset evolution level was reached and call the evolve() method, if they are an EvolvablePokemon class.





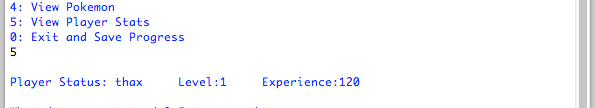


## View Player Status



This option will display information pertaining to the player. Information includes:

* Player Username
* Player Level
* Player Experience

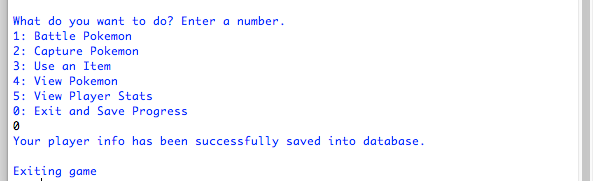


A player can increase their level through gaining experience. Experience is gained in one of two ways:

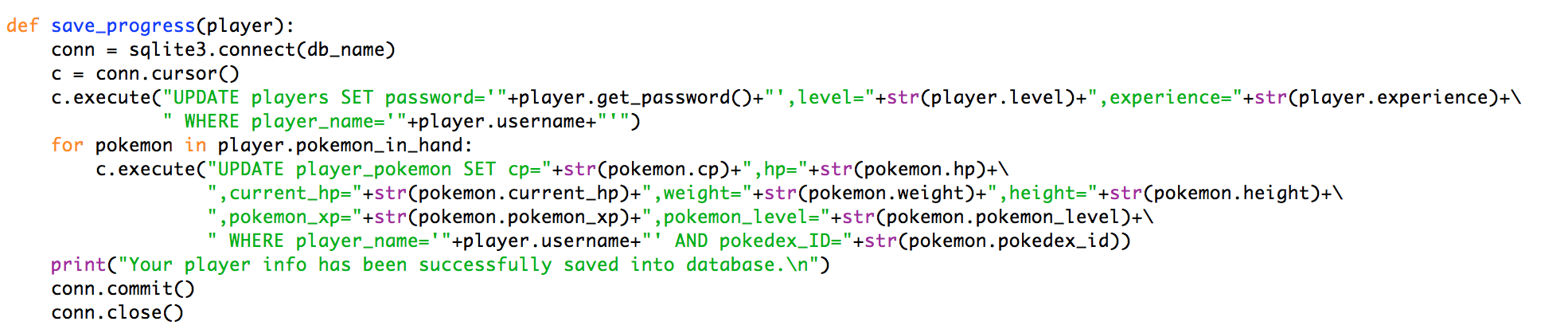
1. Through defeating wild Pokémon in battle
2. Through capturing wild Pokémon

# Save & Exit

At the player menu, you have the option to save and exit the program. This will end the game, and your progress will be saved.



The save\_progress() function called will update the player’s specific Pokemon statistics and player’s specific statisitcs. As detailed above, common data on Pokemon (name, types, moves, etc) is written at the time of encounter and variable data specific to the player is written when captured. When saving, this variable data is updated based on what the player has done in their most recent session.

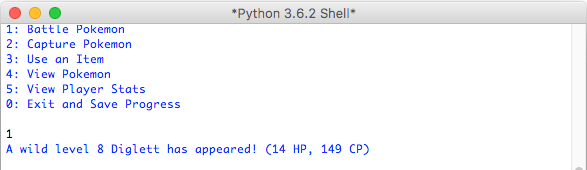


Program Overview

# Pokémon Encounter

*Functionality implemented to determine how a player encounters Pokémon in both battles and capture.*

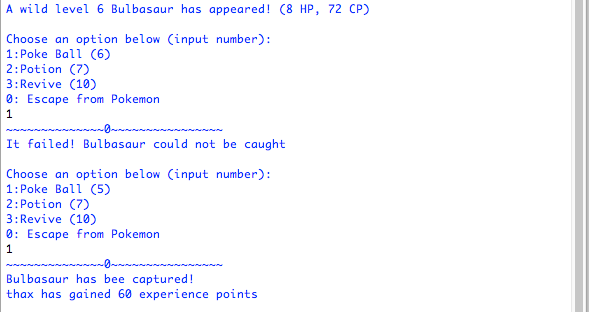
* The generate\_pokemon function is used in both capture and battle menus and will randomly generate a Pokémon based on the *player’s* current level.
  + It’s only parameter is an instantiated player.
  + A while statement creates a loop that will generate several Pokémon until 1 is generated within the range: player level 🡪 player level + 10
  + Each Pokémon has a specific min and max level it can be generated at within pokedex.csv, which also controls which Pokémon are generated, not just the level/strength (this is also used to manage evolutions below).
  + This allows the user to encounter more basic Pokémon at early levels and, as they level up, encounter more and more advanced Pokémon.



# Capture Pokémon

*Functionality implemented to capture new Pokémon to add new Pokémon to a player’s hand that they can battle with.*

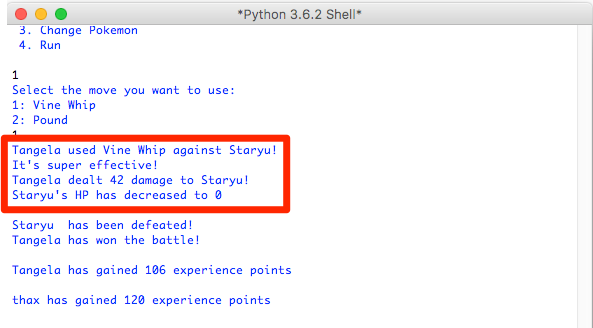
* The capture module utilizes the provided code as its base with the following extensions/variations:
  + The Pokémon generated use the generate\_pokemon function outlined above.
  + *Pokémon do not have a flat 1 for catch chance. More advanced Pokémon have a value less than 1.*
  + The chance to catch (catch) is not just a randomly generated number but is a randomly generated based on the target level.
    - The range is the Pokémon’s current level and the cap is the current level + 50. Once a Pokémon is level 50, that cap becomes 100 (or 1) by default.
    - If a capture fails, a random number is generated based on target’s catch chance and compared to catch variable to determine if they escape. It is designed so higher level Pokémon with a lower catch\_chance escape much more frequently.
  + Successful capture of the Pokémon will gain the player experience based on the captured Pokémon’s level.



# Type System

*Functionality implemented to generate a multiplier for battles based on a given move’s type compared to both types of the target.*

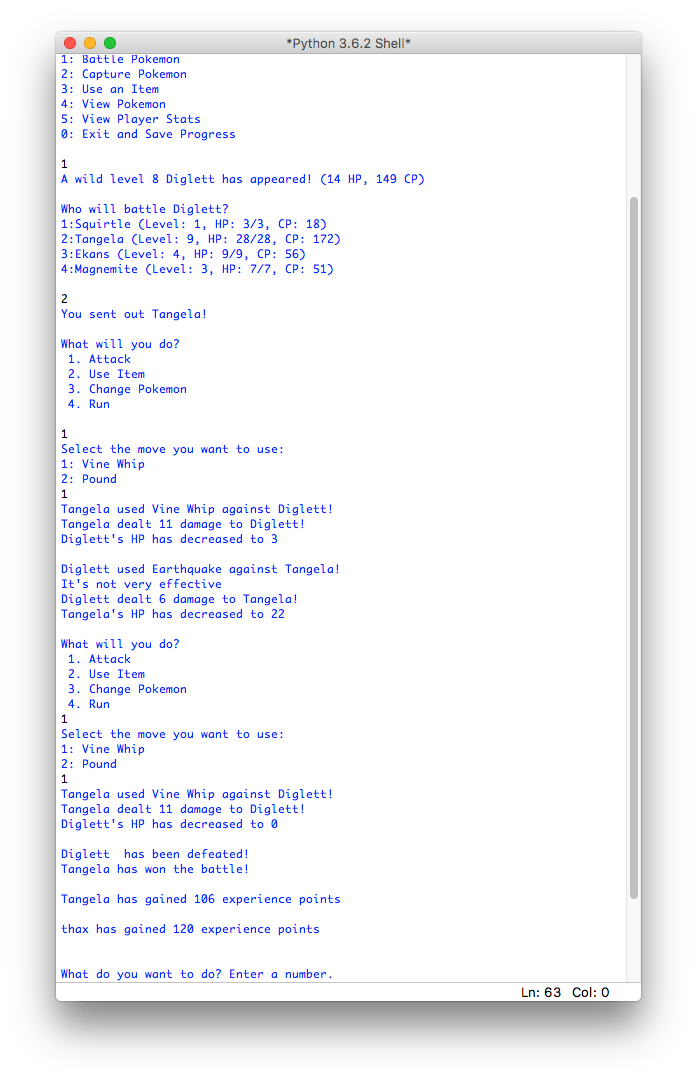
* Each Pokémon is assigned 2 moves, 1 for each of their types.
* A separate module was created to determine compare that type to all other types.
* A separate function was created to call the type comparison twice, one for each of target’s types.
  + Each type can either do no damage (0 multiplier), be less effective (0.5) or super effective (2). If there is no relation between types, a 1 is returned so damage dealt is unaffected.
  + The function multiplies the results of both checks to return 1 multiplier to be used in the attack.
* This is then multiplied by the damage dealt, which a function of the Pokémon’s CP, to either increase or decrease the attack.



# Battle System

*Functionality implemented to not only allow 2 Pokémon to battle and also allow user to change Pokémon use items during the battle and select a move to use for each attack round.*

* When battle called, a player will choose which Pokémon to send out based on Pokémon generated (as detailed above).
* Player then can attack, use an item, run or switch Pokémon.
  + By switching a new Pokémon from their hand can enter the battle.
  + The target will only use their default move and is only triggered when player attacks.
  + Damage is a function of both Pokémon’s CP and type multiplier detailed above.
* If a Pokémon is successfully defeated, the player will receive experience based on the target’s level.
  + If a Pokémon sent out gets defeated, will check to ensure remaining Pokémon have HP and give player an option to send out another.
* Unlike the player, the Pokémon will receive experience based on the difference in level between itself and the target.
  + This means defeating weaker Pokémon will result in less experience and stronger Pokémon in more experience.



# Evolution

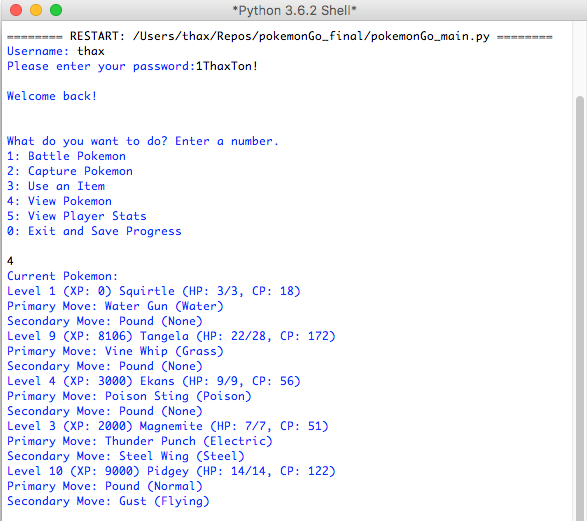
*Functionality implemented to control how a Pokémon evolves.*

* Pokémon evolution is controlled by level. If a Pokémon is evolvable, they will have a preset level at which they will evolve.
  + When experience is added and they gain enough to increase level, if the new level is reached, the evolve function will be called.
  + At each level up for a Pokémon, their HP and CP is increased.
* As a player levels up, they are given new items based on their current level.
  + Each time they increase in level, they have a random chance to receive a candy to speed up the evolution of their Pokémon. This is designed so a higher number will need to be generated to receive it so most leveling will need to be done through battles.

# Database

*Functionality implemented to control how player information is saved/stored.*

* The database is used for player creation and saving progress.
* The only modifications made are reflective of the changes made in the other functions to ensure all information is stored.
  + The ‘get\_pokemon’ function has been removed as it is handled in the pokemon.py module to enable to produce Pokémon based on a player’s level.



# Location

*NOT IMPLETEMEND*

# Egg/Incubator

*NOT IMPLEMENTED*

# Faction/Gym

*NOT IMPLEMENTED*

# GUI

*NOT IMPLEMENTED*

**MSIS 615 Final Project**

(Please attach this page to your final report)

*I affirm that I have neither given, received, nor witnessed unauthorized aid on this deliverable and have completed this work honestly and according to the professor’s guidelines. Please PRINT and sign below.*

**Group Member: Andy Quach**

**Group Member: Christian Thaxton**

**Group Member \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**