For this assignment, you need to emulate a small Pokemon battle between two Pokemon. You will need many classes to achieve this, all of which are working in conjunction. The following is what you need to receive a full grade on the assignment:

1. You must have a Pokemon abstract class. It must have, in the least, the following methods, that are also abstract:
   1. public abstract void attack(Pokemon other, int

attackIndex)

* 1. public abstract void speak()

1. Think about the data a Pokemon would have. As it is an abstract class, it is permitted to have member variables. In the least, it should have hit points, power points, a name, a type, and a List of Attacks. All of these need to be passed into a constructor.
2. You need an Attack class. (This has been highly recommended by my students through the years). An Attack object has a String description of the attack, an int power points cost (which can be 0), a type (which can be “none”), which is a String, and base damage, which is an int. You will also need accessor methods for each data item. This class is meant to simplify the act of attacking (trust me on this – I used to leave this wide open for interpretation and people spent far too much time figuring this out!)
3. You need an abstract Player class. This Player is not going to have much information, other than the **ONE** Pokemon they have prepared for battle and **ONE** Item they may have. You will need accessor methods to access the Pokemon and Item from the runner. The Player class should have a constructor that takes in one Pokemon and one Item. The Player class should also have the following abstract methods:
   1. public abstract void run()
4. Create two concrete Player classes. One is a HumanPlayer, whose actions are controlled in the runner, and ComputerPlayer, whose actions are **random**.
5. You need to create at least **two** unique concrete Pokemon classes of differing types. The values are up to you. The following is a table of Pokemon types you can use (choose *two* different ones!):

A screenshot of a cell phone

Description automatically generated

For your assignment, super effective is double the base damage of an Attack, not very effective is half the base damage of the Attack, and no effect is no damage.

1. Create an Item interface with one method: public void use(Pokemon p). Create at least *two* Items, one that the Player has, and one that the ComputerPlayer has.

Your runner needs to create instantiate all of the data you will need. It needs to create each Pokemon’s list of Attacks, the two concrete Item classes, the two concrete Pokemon classes, and the two concrete Player classes. The rest of the runner should call each of the Player Pokemon’s methods (attack, useItem, run) in order. The ComputerPlayer needs to do three of the following, all with equal probability:

1. Call it’s Pokemon’s attack method against the HumanPlayer’s Pokemon. It should call a random attack by choosing a random number based on the List’s size.
2. Call it’s getItem method.
3. Call it’s run method.

For turn-in, I need all of your code, and your output, as well as a UML diagram as to how all of these classes are related. Comment your code as per Javadoc standards.