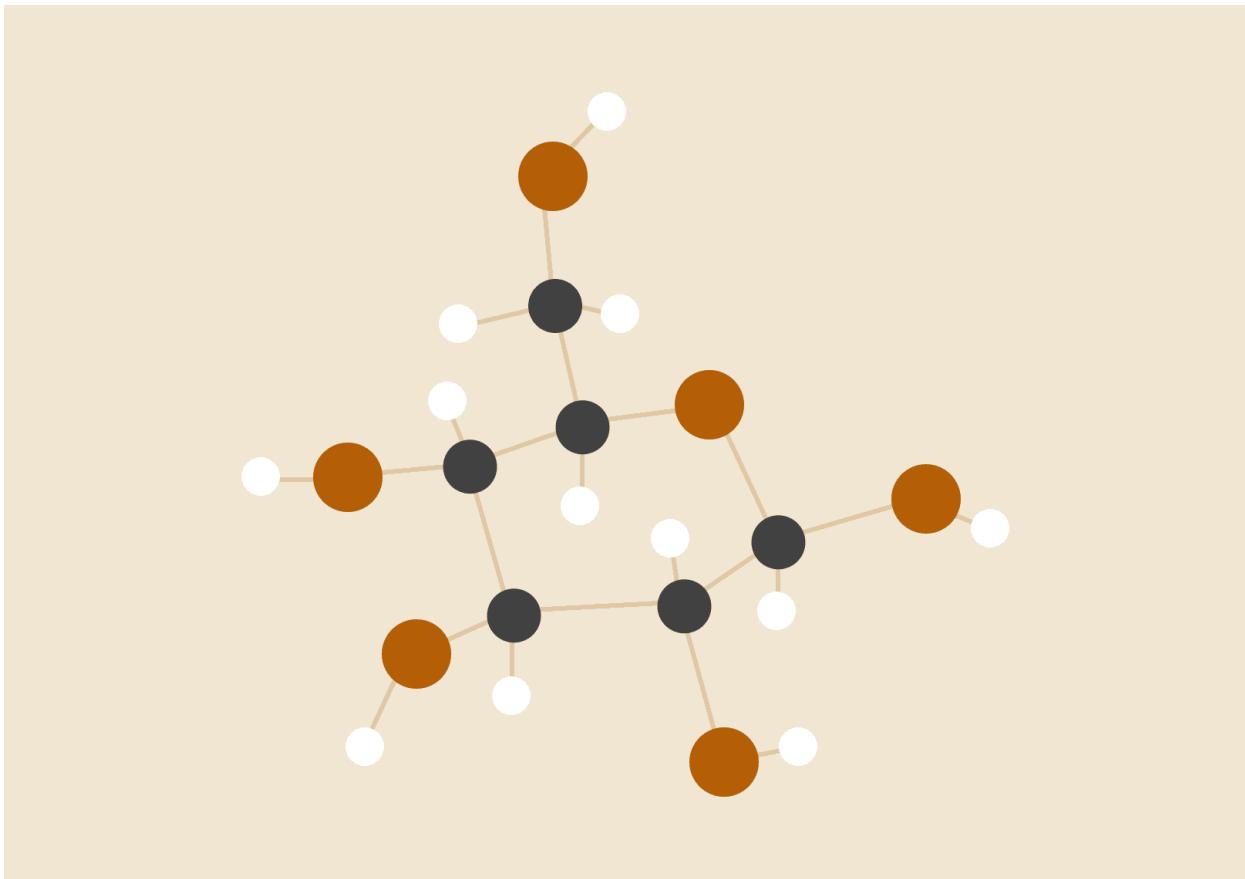


Mobile App of Pokemon Game



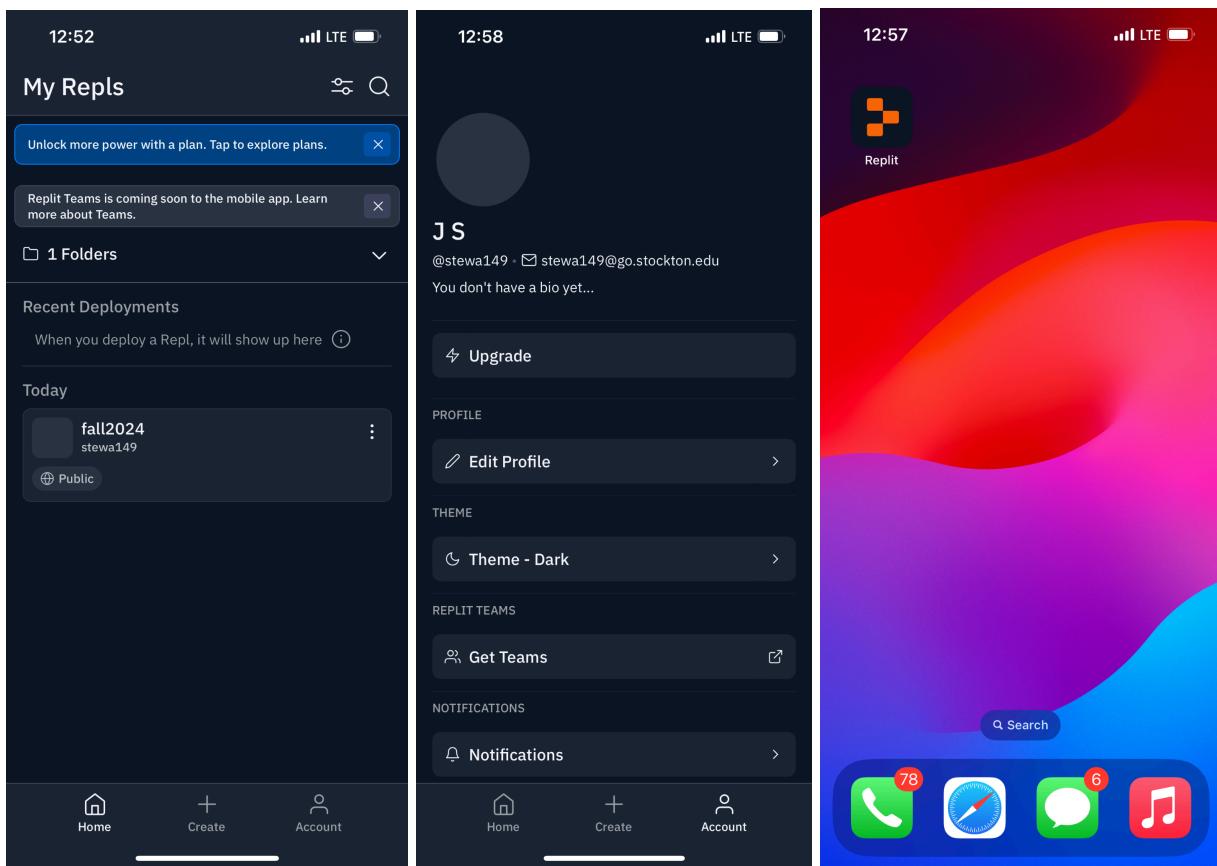
Jack Stewart

12.10.2024
Probability and Applied Statistics

Part 1: Finding an IDE

Replit

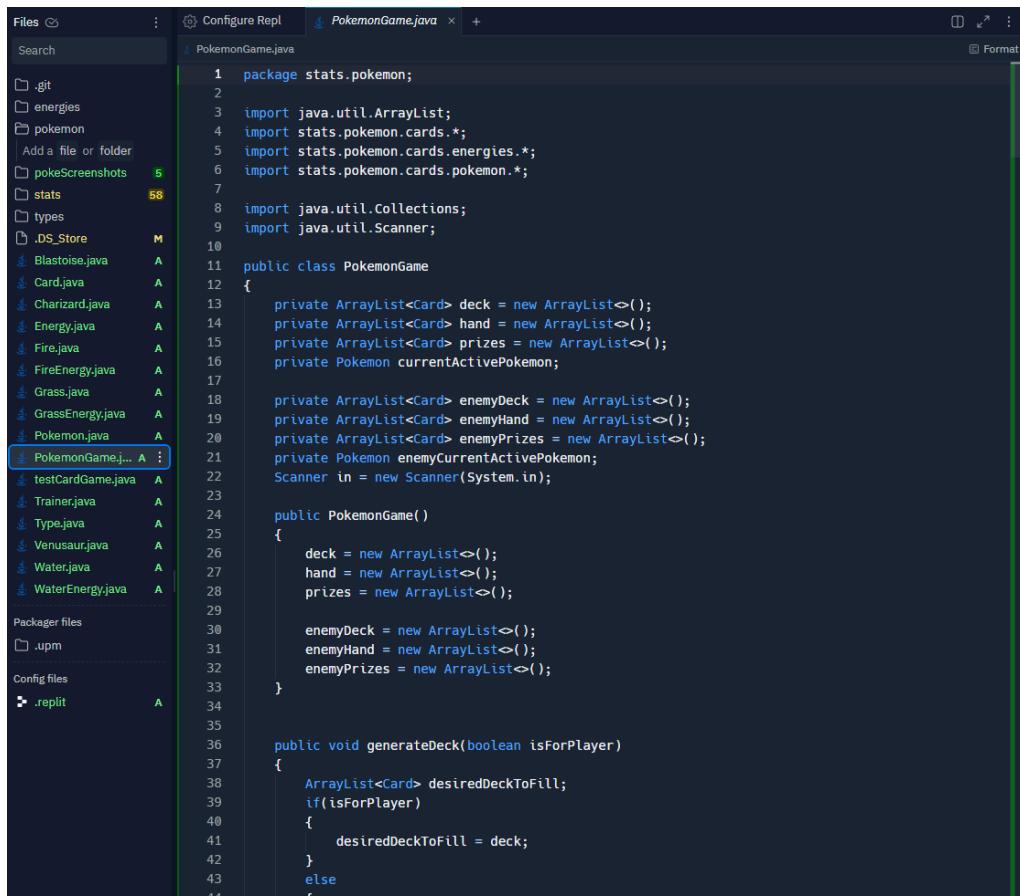
The first application I tried to use to make the application run on my phone was Replit. It was the only application I could find that claimed to be able to run java programs on an Iphone. I installed it on my phone and tried to install my files through github, which worked fine. The image below is from the Replit mobile app which I was trying to use to run the code.



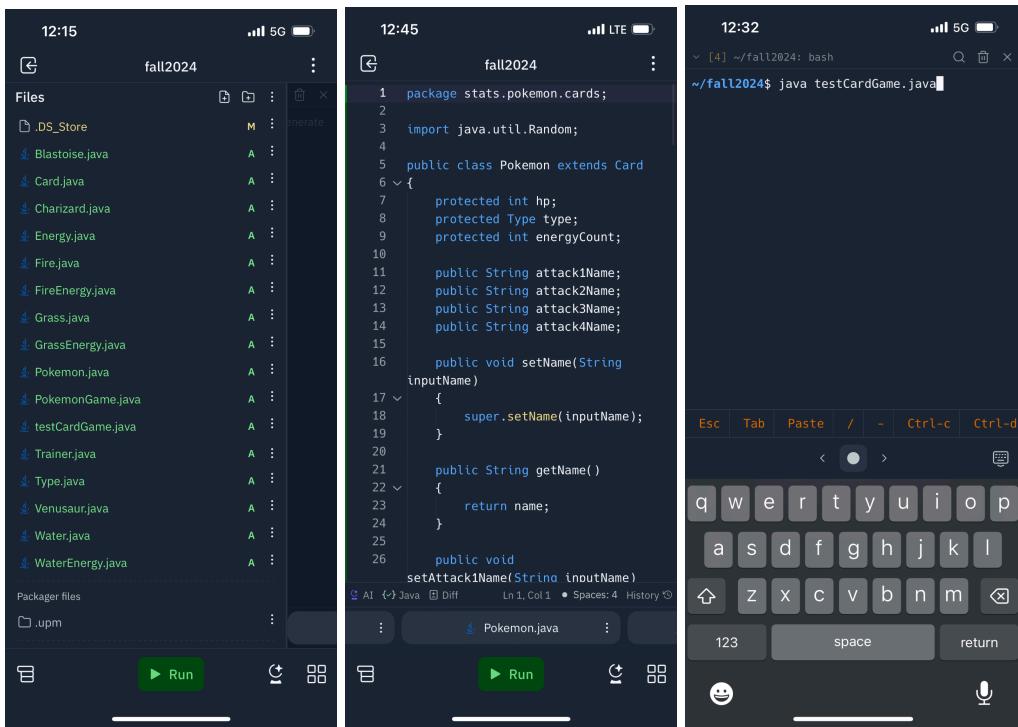
I initially strongly disliked Replit as it felt extremely hard to navigate. When I tried to import only my pokemon files it refused, and decided that instead I needed to install the entire repository which contained a ton of excess files. Navigating through it on my phone was especially frustrating.

Part 2: Running the Program

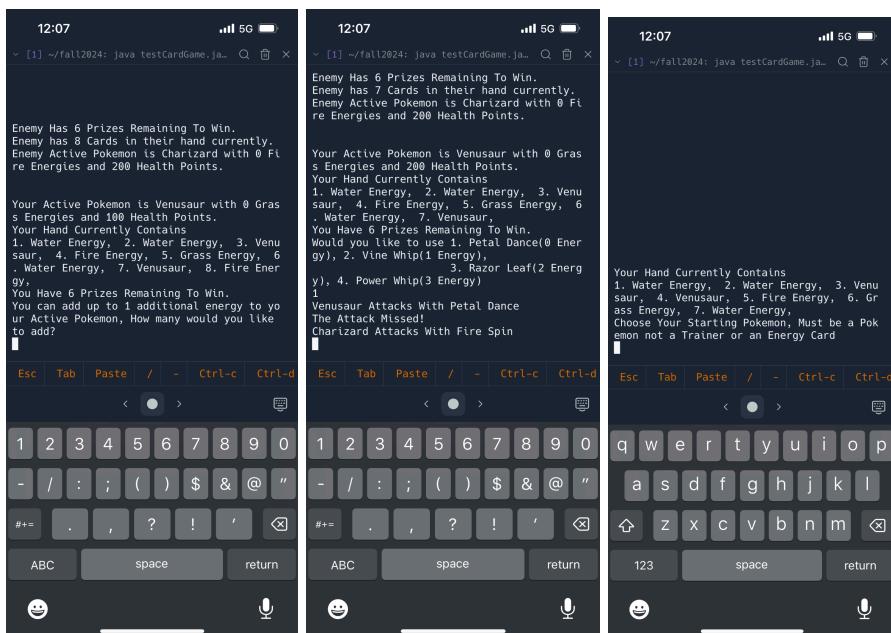
While I was able to import my repository from project 1, I had a lot of problems with Replit not running it. Replit for some reason wanted to run a different file, and despite going through tons of settings it was extremely difficult to get it to run the right file by default. I went through the setting of the app, changed the .replit file, and it still completely refused to run the right program instead opting to run the airplane program from Data Structures the entire time. I ended up trying to run the file directly from the shell, and this did not work either as it refused to locate any of my other subclasses that were needed to run the program such as pokemon, energy, charizard, ect. I worked on this for over an hour with no luck, so I began looking for other options. I searched back over the web for other programs that could run the programs, with no luck. I tried for a short period to run the program in an android virtual machine but that was presenting its own problems. I then went back to Replit and chose to try something different, by moving every single file to the main directory and deleting every other folder from the main directory. This surprisingly fixed the problems I was having with running the program and running it directly from the shell worked fine. You can see the way the repository ended up below on both the desktop and mobile views.



```
1 package stats.pokemon;
2
3 import java.util.ArrayList;
4 import stats.pokemon.cards.*;
5 import stats.pokemon.cards.energies.*;
6 import stats.pokemon.cards.pokemon.*;
7
8 import java.util.Collections;
9 import java.util.Scanner;
10
11 public class PokemonGame
12 {
13     private ArrayList<Card> deck = new ArrayList<>();
14     private ArrayList<Card> hand = new ArrayList<>();
15     private ArrayList<Card> prizes = new ArrayList<>();
16     private Pokemon currentActivePokemon;
17
18     private ArrayList<Card> enemyDeck = new ArrayList<>();
19     private ArrayList<Card> enemyHand = new ArrayList<>();
20     private ArrayList<Card> enemyPrizes = new ArrayList<>();
21     private Pokemon enemyCurrentActivePokemon;
22     Scanner in = new Scanner(System.in);
23
24     public PokemonGame()
25     {
26         deck = new ArrayList<>();
27         hand = new ArrayList<>();
28         prizes = new ArrayList<>();
29
30         enemyDeck = new ArrayList<>();
31         enemyHand = new ArrayList<>();
32         enemyPrizes = new ArrayList<>();
33     }
34
35
36     public void generateDeck(boolean isForPlayer)
37     {
38         ArrayList<Card> desiredDeckToFill;
39         if(isForPlayer)
40         {
41             desiredDeckToFill = deck;
42         }
43         else
44         {
45             desiredDeckToFill = hand;
46         }
47         Collections.shuffle(desiredDeckToFill);
48         desiredDeckToFill.addAll(prizes);
49         desiredDeckToFill.addAll(enemyPrizes);
50
51         if(isForPlayer)
52         {
53             deck = desiredDeckToFill;
54         }
55         else
56         {
57             hand = desiredDeckToFill;
58         }
59     }
60 }
```



A big part of how I was able to make this happen was the decision to stop trying to work on my phone and use the Replit desktop app on my computer, which made navigating the program pretty easy. The main run functionality of Replit still didn't work, but running directly through the shell with the command `java testCardGame` worked completely fine without requiring any actual debugging within the files, which surprised me. I played through a game of it on my phone, with no problems whatsoever from this point onwards, and there are screenshots below of what the interface looked like.



It's been great working with you inside and outside of the classroom this year professor! I'm really glad I got the opportunity to work with you specifically to improve my programming skills because I feel a lot more confident with them now. I am sorry my project 2 is not quite as finished as it could be but I tried to make it as good and polished as possible, I simply ended up not having enough time to make it through the entire project, particularly part 2 I just didn't have time to do in full to the quality of standard I was hoping for. I hope you enjoyed reading through it nonetheless, and could see the story in Parts 1 and 3 that I didn't have the time to build into Part 2. This won't be the last you will see of me, I'll be coming in to find you at some point next semester for help with my resume and preparing for the hunt for internships and jobs. Once again thank you, especially for the opportunity to sit in on your Data structures class, I cannot overstate how much better my programming became through just that one class. I found a copy of the clean coding book online that you mentioned last class and I plan to start reading it this Winter. I'll let you know what I think of it next semester. Have a great winter break and happy holidays!