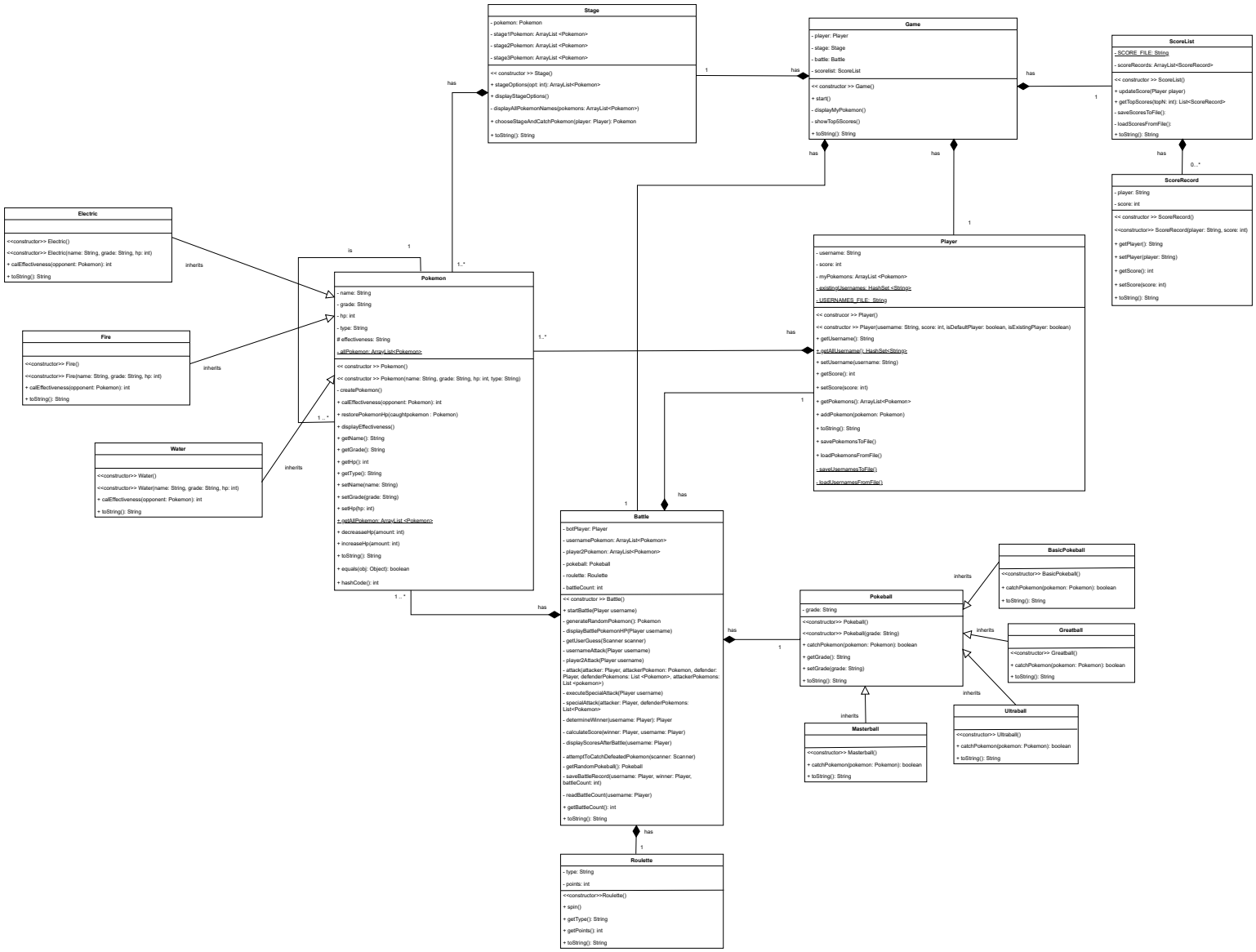
Stage
- pokemon: Pokemon
- stage1Pokemon: ArrayList <Pokemon>
- stage2Pokemon: ArrayList <Pokemon>
- stage3Pokemon: ArrayList <Pokemon>
<< constructor >> Stage()
+ stageOptions(opt: int): ArrayList<Pokemon>
+ displayStageOptions()
- displayAllPokemonNames(pokemons: ArrayList<Pokemon>)
+ chooseStageAndCatchPokemon(player: Player): Pokemon
+ toString(): String
Game
- player: Player
- stage: Stage
- battle: Battle
- scorelist: ScoreList
<< constructor >> Game()
+ start()
- displayMyPokemon()
- showTopSScores()
+ toString(): String
ScoreList
- SCORE_FILE: String
- scoreRecords: ArrayList<ScoreRecord>
<< constructor >> ScoreList()
+ updateScore(Player player)
+ getTopScores(topN: int): List<ScoreRecord>
- saveScoresToFile():
- loadScoresFromFile():
+ toString(): String
ScoreRecord
- player: String
- score: int
<< constructor >> ScoreRecord()
<<constructor>> ScoreRecord(player: String, score: int)
+ getPlayer(): String
+ setPlayer(player: String)
+ getScore(): int
+ setScore(score: int)
+ toString(): String
Electric
<<constructor>> Electric()
<<constructor>> Electric(name: String, grade: String, hp: int)
+ calEffectiveness(opponent: Pokemon): int
+ toString(): String
Fire
<<constructor>> Fire()
<<constructor>> Fire(name: String, grade: String, hp: int)
+ calEffectiveness(opponent: Pokemon): int
+ toString(): String
Water
<<constructor>> Water()
<<constructor>> Water(name: String, grade: String, hp: int)
+ calEffectiveness(opponent: Pokemon): int
+ toString(): String
Pokemon
- name: String
- grade: String
- hp: int
- type: String
effectiveness: String
- allPokemon: ArrayList<Pokemon>
<< constructor >> Pokemon()
<< constructor >> Pokemon(name: String, grade: String, hp: int, type: String)
- createPokemon()
+ calEffectiveness(opponent: Pokemon): int
+ restorePokemonHp(caughtpokemon : Pokemon)
+ displayEffectiveness()
+ getName(): String
+ getGrade(): String
+ getHp(): int
+ getType(): String
+ setName(name: String)
+ setGrade(grade: String)
+ setHp(hp: int)
+ getAllPokemon: ArrayList <Pokemon>
+ decreaseHp(amount: int)
+ increaseHp(amount: int)
+ toString(): String
+ equals(obj: Object): boolean
+ hashCode(): int
Player
- username: String
- score: int
- myPokemons: ArrayList <Pokemon>
- existingUsernames: HashSet <String>
- USERNAMES_FILE: String
<< construcor >> Player()
<< constructor >> Player(username: String, score: int, isDefaultPlayer: boolean, isExistingPlayer: boolean)
+ getUsername(): String
+ getAllUsernames(): HashSet<String>
+ setUsername(username: String)
+ getScore(): int
+ setScore(score: int)
+ getPokemons(): ArrayList<Pokemon>
+ addPokemon(pokemon: Pokemon)
+ toString(): String
+ savePokemonsToFile()
+ loadPokemonsFromFile()
- saveUsernamesToFile()
- loadUsernamesFromFile()
Battle
- botPlayer: Player
- usernamePokemon: ArrayList<Pokemon>
- player2Pokemon: ArrayList<Pokemon>
- pokeball: Pokeball
- roulette: Roulette
- battleCount: int
<< constructor >> Battle()
+ startBattle(Player username)
- generateRandomPokemon(): Pokemon
- displayBattlePokemonHP(Player username)
- getUserGuess(Scanner scanner)
- usernameAttack(Player username)
- player2Attack(Player username)
- attack(attacker: Player, attackerPokemon: Pokemon, defender: Player, defenderPokemons: List <Pokemon>, attackerPokemons: List <pokemon>)
- executeSpecialAttack(Player username)
- specialAttack(attacker: Player, defenderPokemons: List<Pokemon>)
- determineWinner(username: Player): Player
- calculateScore(winner: Player, username: Player)
- displayScoresAfterBattle(username: Player)
- attemptToCatchDefeatedPokemon(scanner: Scanner)
- getRandomPokeball(): Pokeball
- saveBattleRecord(username: Player, winner: Player, battleCount: int)
- readBattleCount(username: Player)
+ getBattleCount(): int
+ toString(): String
Roulette
- type: String
- points: int
<<constructor>>Roulette()
+ spin()
+ getType(): String
+ getPoints(): int
+ toString(): String
Pokeball
- grade: String
<<constructor>> Pokeball()
<<constructor>> Pokeball(grade: String)
+ catchPokemon(pokemon: Pokemon): boolean
+ getGrade(): String
+ setGrade(grade: String)
+ toString(): String
BasicPokeball
<<constructor>> BasicPokeball()
+ catchPokemon(pokemon: Pokemon): boolean
+ toString(): String
Greatball
<<constructor>> Greatball()
+ catchPokemon(pokemon: Pokemon): boolean
+ toString(): String
Ultraball
<<constructor>> Ultraball()
+ catchPokemon(pokemon: Pokemon): boolean
+ toString(): String
Masterball
<<constructor>> Masterball()
+ catchPokemon(pokemon: Pokemon): boolean
+ toString(): String
has
1
1..*
is
inherits
1...*