Pokemua

Created by

Chayakorn Kittikoon 6531310321

Pannawich Lohanimit 6532120621

2110215 Programming Methodology

Semester 2 Year 2023

Chulalongkorn University

Pokemua

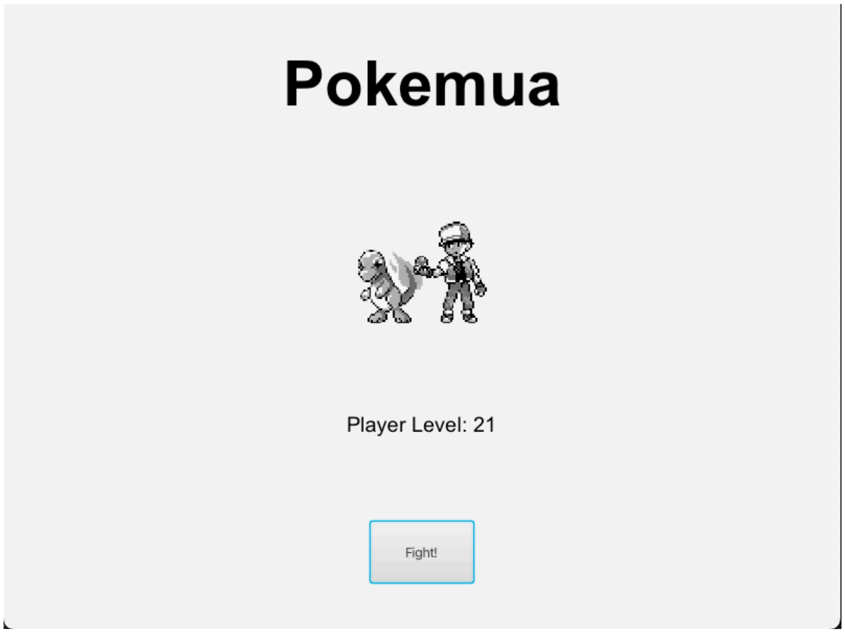
Introduction

Pokemua is inspired by famous video games, Pokemon Red, and Pokemon Blue. However, this game is battle royale, so player need to survive in this game.

Rules

When player start this game for the first time, player will need to choose their first Pokemua**.** In each game, player can choose up to 3 Pokemua to play in the game. Player needs to survive from another player(bot) and from wild Pokemua. Player will win when they can be the last player in the game and the Exp and their Pokemua will be level up too. When Pokemua have reached required level they can get new skill and new stat. For each Pokemua, they can have maximum skill at 4 skills so when they get new skill, they need to remove the old one.

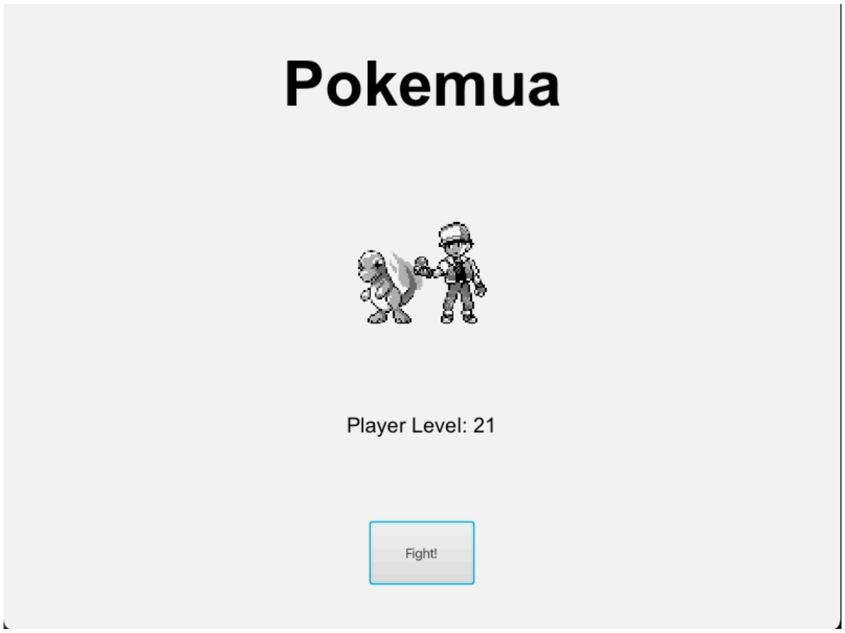
First Pokemua scene



Pokemua

Player Level: 21

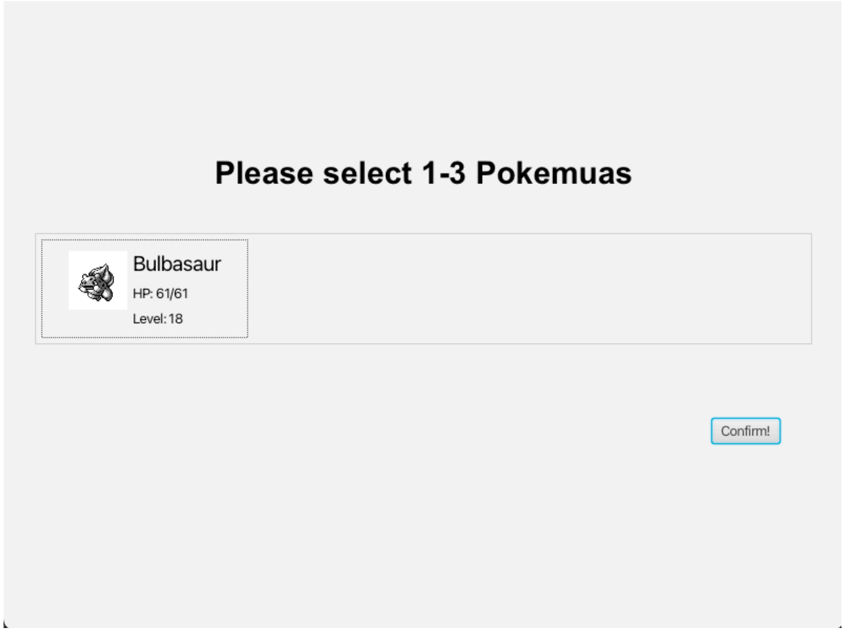
Home scene



Pokemua

Player Level: 21

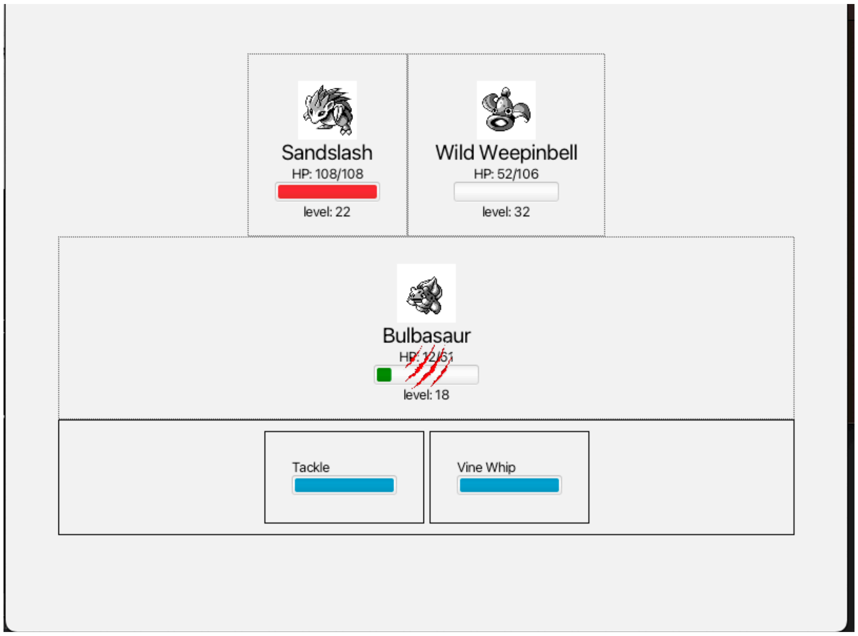
Select Pokemua scene



Please select 1-3 Pokemuas

Bulbasaur
HP: 61/61
Level: 18

Fight scene

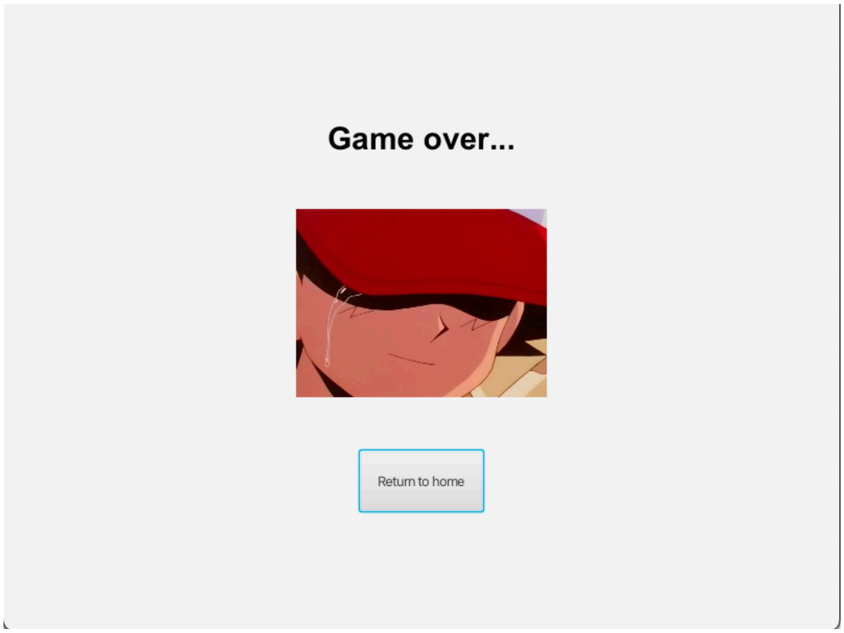


Sandslash
HP: 108/108
level: 22

Wild Weepinbell
HP: 52/106
level: 32

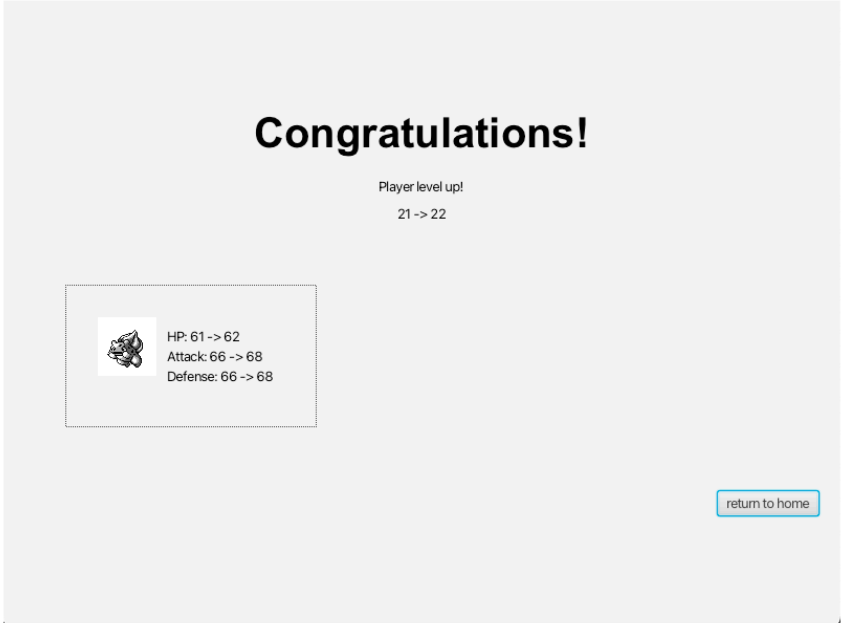
Bulbasaur
HP: 12/61
level: 18

Fight lose scene



Game over...

Fight win scene



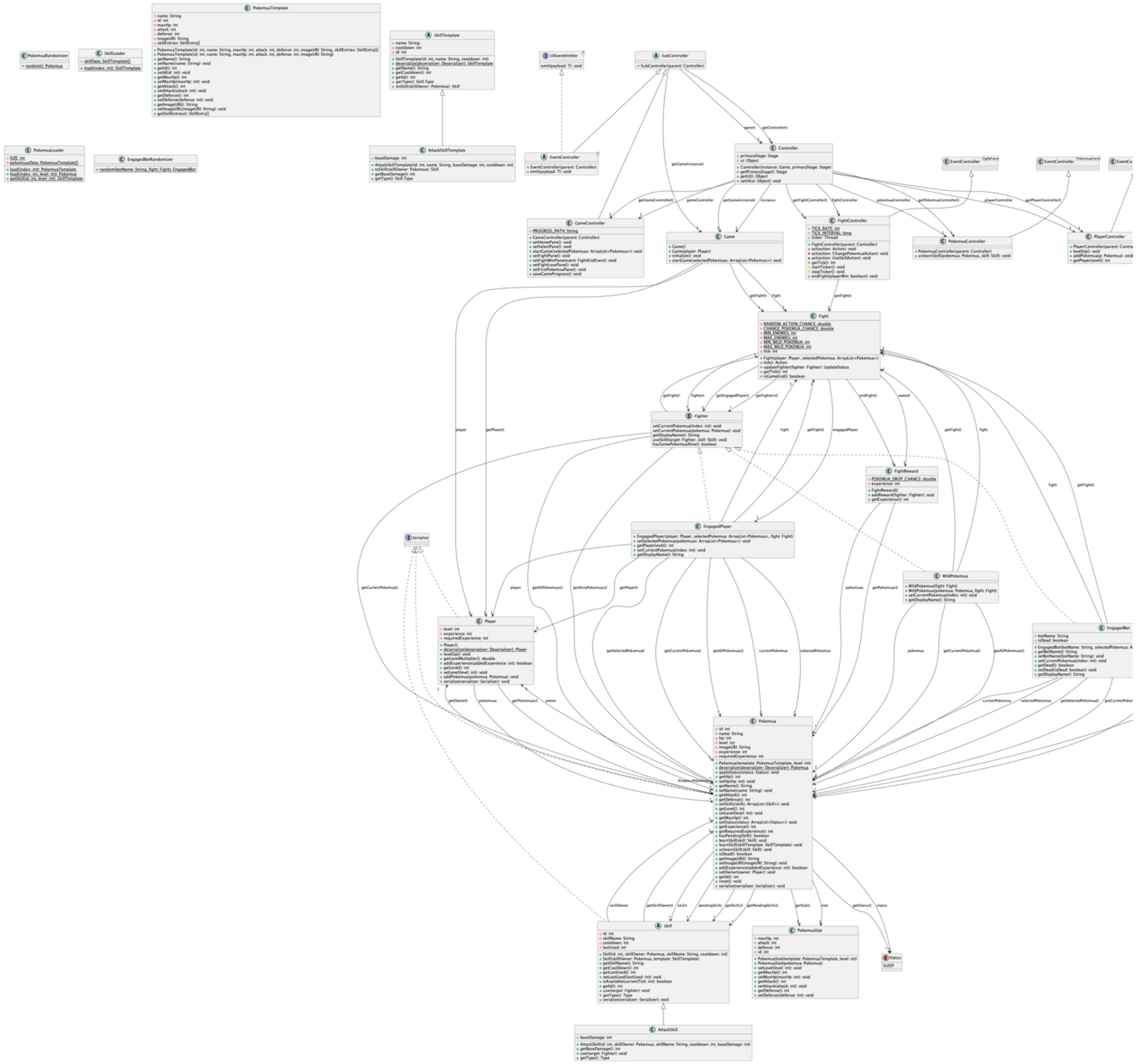
Congratulations!

Player level up!

21 -> 22

HP: 61 -> 62
Attack: 66 -> 68
Defense: 66 -> 68

Class diagram



1. Package action

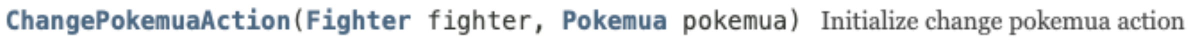
1.1 Class Action

1.1.1 Method

abstract Action.Type getType() Get type of action

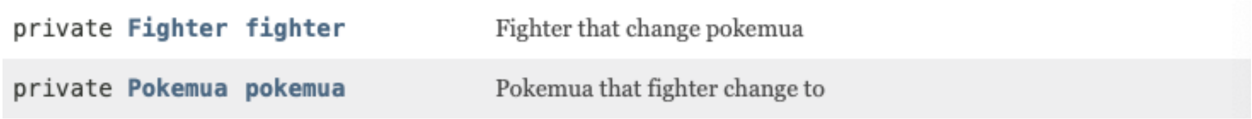
1.2 Class ChangePokemuaAction

1.2.1 Constructor



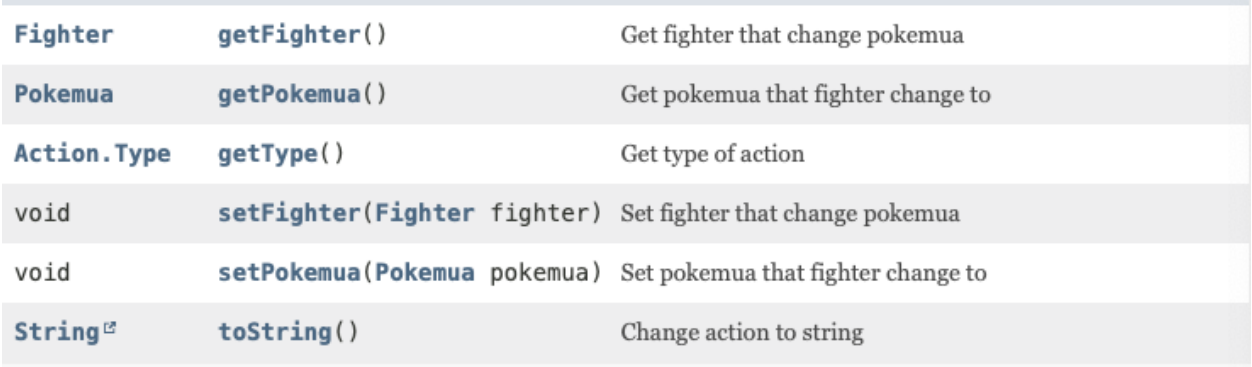
`ChangePokemuaAction(Fighter fighter, Pokemua pokemua)`	Initialize change pokemua action

1.2.2 Field



`private Fighter fighter`	Fighter that change pokemua
`private Pokemua pokemua`	Pokemua that fighter change to

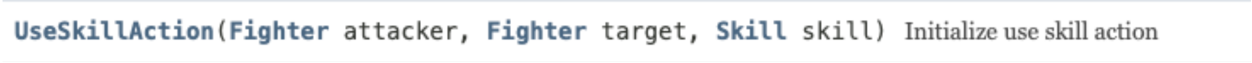
1.2.3 Method



`Fighter`	`getFighter()`	Get fighter that change pokemua
`Pokemua`	`getPokemua()`	Get pokemua that fighter change to
`Action.Type`	`getType()`	Get type of action
`void`	`setFighter(Fighter fighter)`	Set fighter that change pokemua
`void`	`setPokemua(Pokemua pokemua)`	Set pokemua that fighter change to
`String`	`toString()`	Change action to string

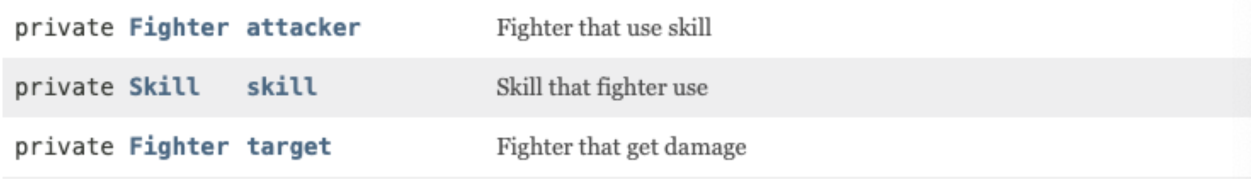
1.3 Class UseSkillAction

1.3.1 Constructor



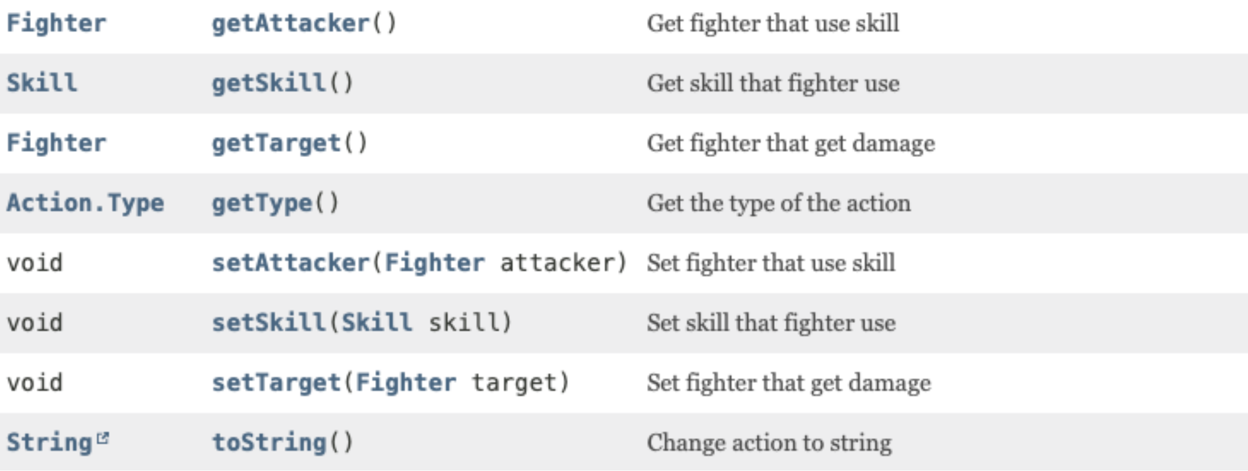
`UseSkillAction(Fighter attacker, Fighter target, Skill skill)`	Initialize use skill action

1.3.2 Field



`private Fighter attacker`	Fighter that use skill
`private Skill skill`	Skill that fighter use
`private Fighter target`	Fighter that get damage

1.3.3 Method

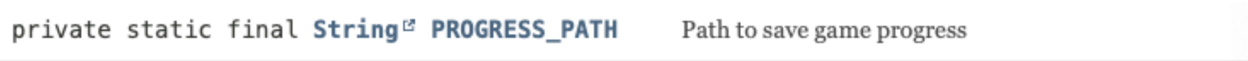


Fighter	getAttacker()	Get fighter that use skill
Skill	getSkill()	Get skill that fighter use
Fighter	getTarget()	Get fighter that get damage
Action.Type	getType()	Get the type of the action
void	setAttacker(Fighter attacker)	Set fighter that use skill
void	setSkill(Skill skill)	Set skill that fighter use
void	setTarget(Fighter target)	Set fighter that get damage
String	toString()	Change action to string

2. Package application

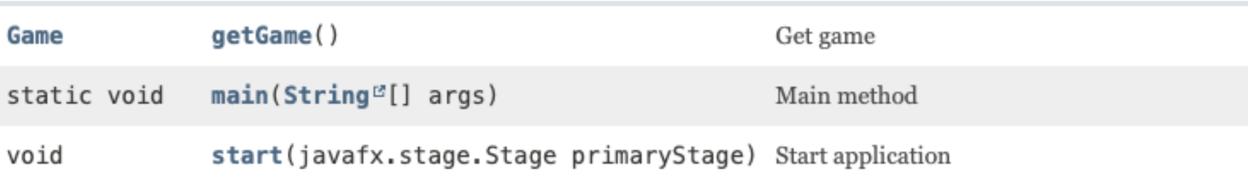
2.1 Class Main

2.1.1 Field



private static final String PROGRESS_PATH	Path to save game progress

2.1.2 Method

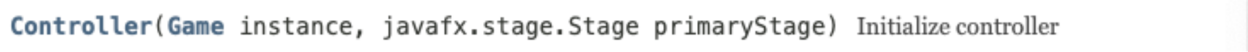


Game	getGame()	Get game
static void	main(String[] args)	Main method
void	start(javafx.stage.Stage primaryStage)	Start application

3. Package controllers

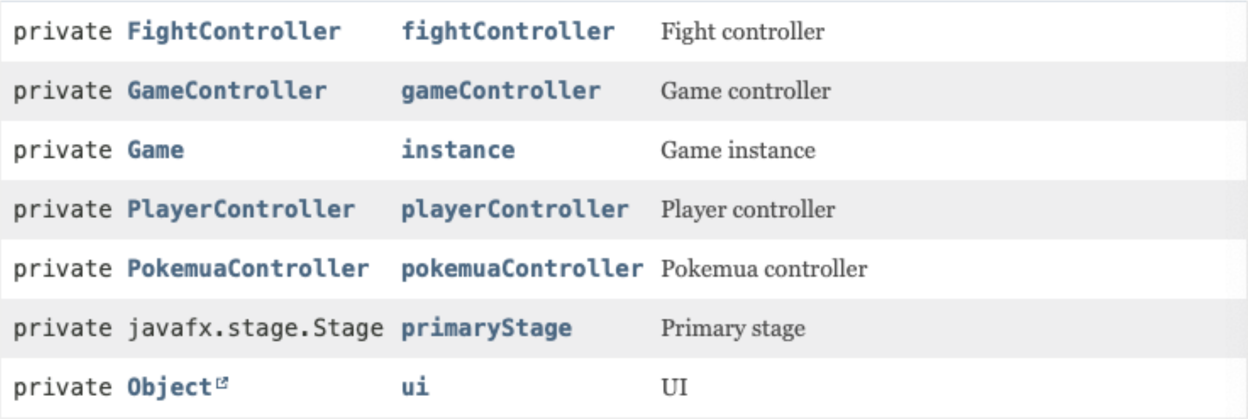
3.1 Class Controller

3.1.1 Constructor



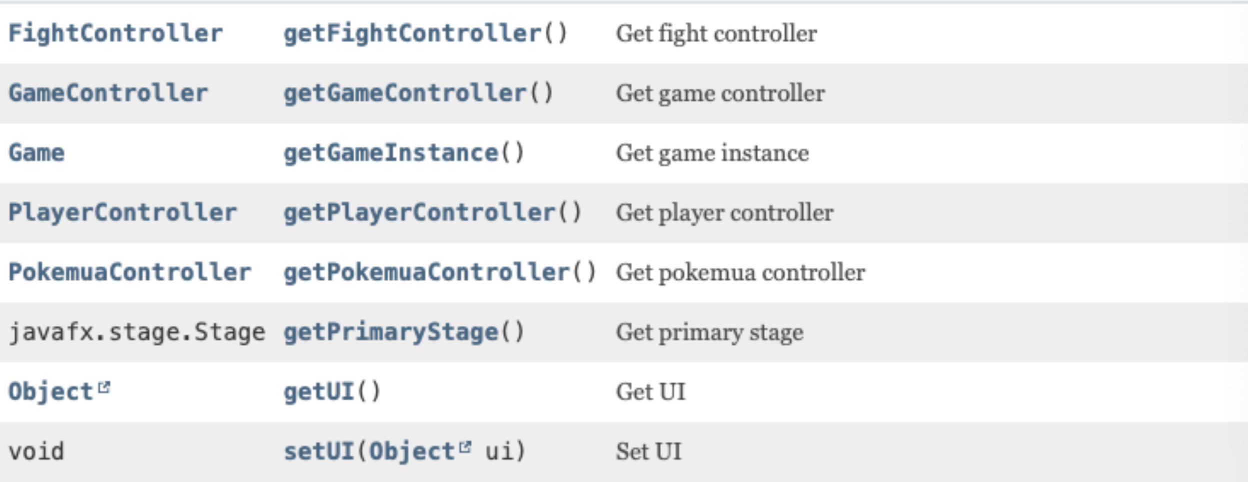
Controller(Game instance, javafx.stage.Stage primaryStage)	Initialize controller

3.1.2 Field



private FightController	fightController	Fight controller
private GameController	gameController	Game controller
private Game	instance	Game instance
private PlayerController	playerController	Player controller
private PokemuaController	pokemuaController	Pokemua controller
private javafx.stage.Stage	primaryStage	Primary stage
private Object	ui	UI

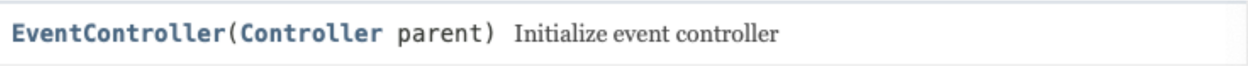
3.1.3 Method



FightController	getFightController()	Get fight controller
GameController	getGameController()	Get game controller
Game	getGameInstance()	Get game instance
PlayerController	getPlayerController()	Get player controller
PokemuaController	getPokemuaController()	Get pokemua controller
javafx.stage.Stage	getPrimaryStage()	Get primary stage
Object	getUI()	Get UI
void	setUI(Object ui)	Set UI

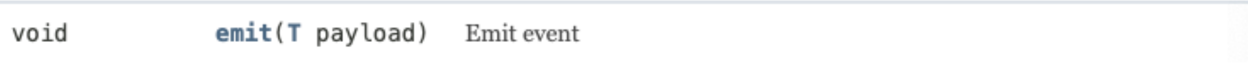
3.2 Class EventController<T extends Event>

3.2.1 Constructor



EventController(Controller parent)	Initialize event controller

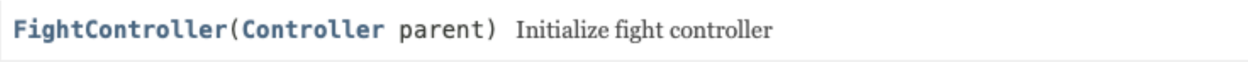
3.2.2 Method



void	emit(T payload)	Emit event

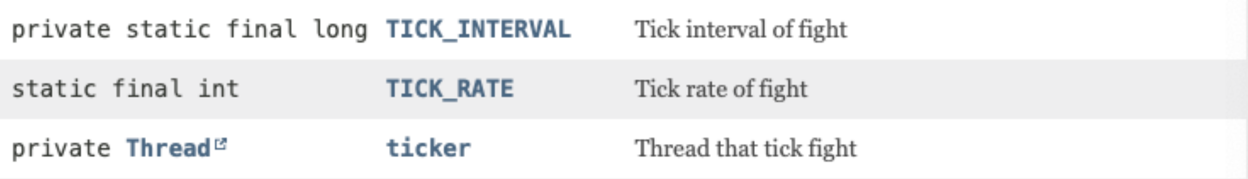
3.3 Class FightController

3.3.1 Constructor



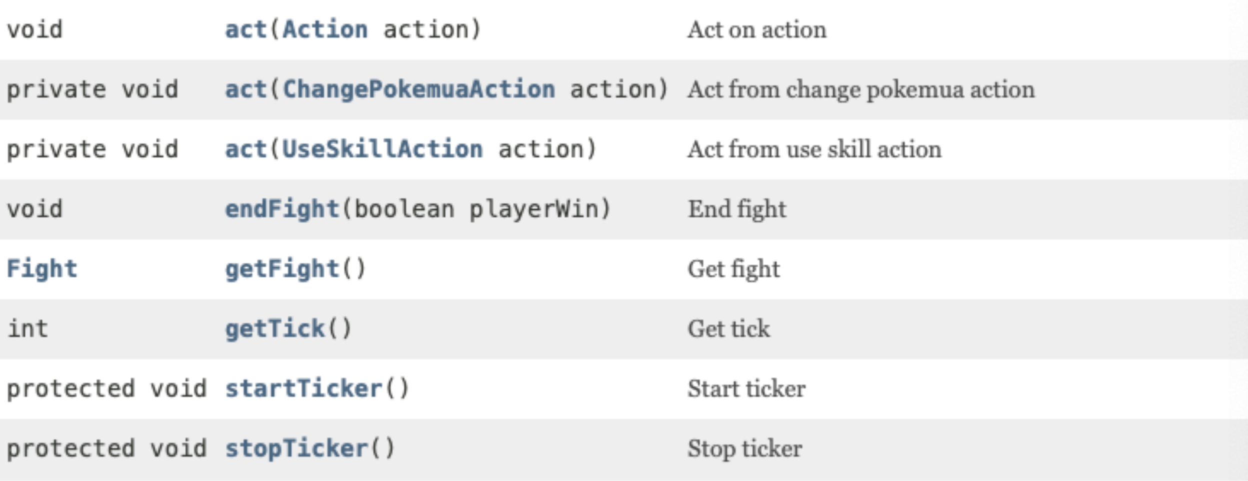
FightController(Controller parent)	Initialize fight controller

3.3.2 Field



private static final long	TICK_INTERVAL	Tick interval of fight
static final int	TICK_RATE	Tick rate of fight
private Thread	ticker	Thread that tick fight

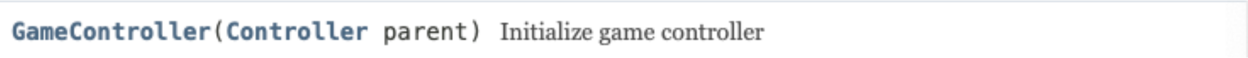
3.3.3 Method



void	act(Action action)	Act on action
private void	act(ChangePokemuaAction action)	Act from change pokemua action
private void	act(UseSkillAction action)	Act from use skill action
void	endFight(boolean playerWin)	End fight
Fight	getFight()	Get fight
int	getTick()	Get tick
protected void	startTicker()	Start ticker
protected void	stopTicker()	Stop ticker

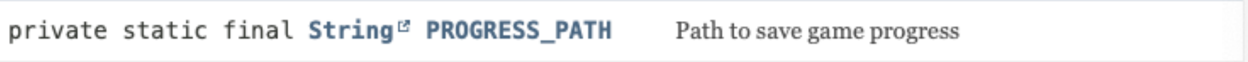
3.4 Class GameController

3.4.1 Constructor



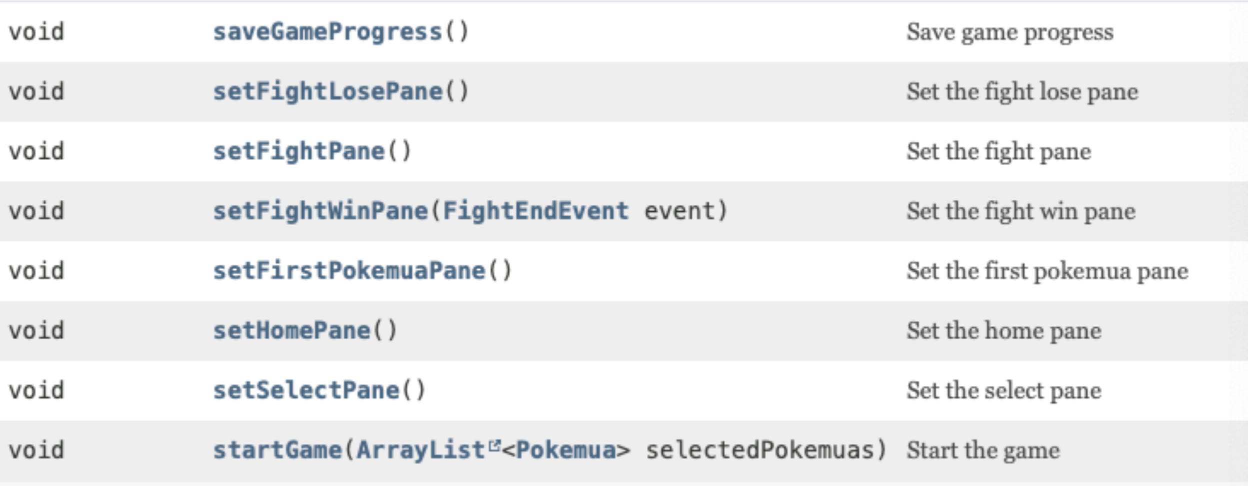
GameController(Controller parent)	Initialize game controller

3.4.2 Field



private static final String	PROGRESS_PATH	Path to save game progress

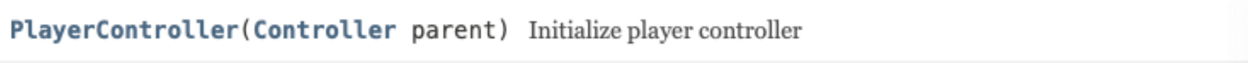
3.4.3 Method



void	saveGameProgress()	Save game progress
void	setFightLosePane()	Set the fight lose pane
void	setFightPane()	Set the fight pane
void	setFightWinPane(FightEndEvent event)	Set the fight win pane
void	setFirstPokemuaPane()	Set the first pokemua pane
void	setHomePane()	Set the home pane
void	setSelectPane()	Set the select pane
void	startGame(ArrayList<Pokemua> selectedPokemuas)	Start the game

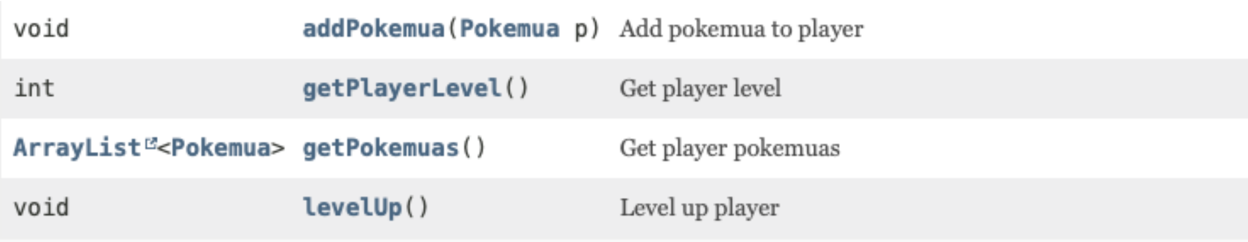
3.5 Class PlayerController

3.5.1 Constructor



PlayerController(Controller parent)	Initialize player controller

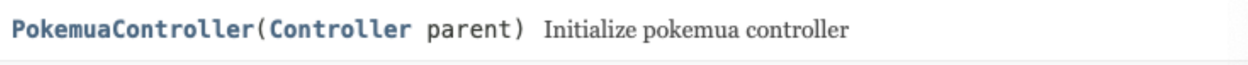
3.5.2 Method



void	addPokemua(Pokemua p)	Add pokemua to player
int	getPlayerLevel()	Get player level
ArrayList<Pokemua>	getPokemuas()	Get player pokemuas
void	levelUp()	Level up player

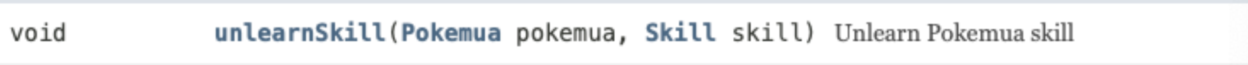
3.6 Class PokemuaController

3.6.1 Constructor



PokemuaController(Controller parent)	Initialize pokemua controller

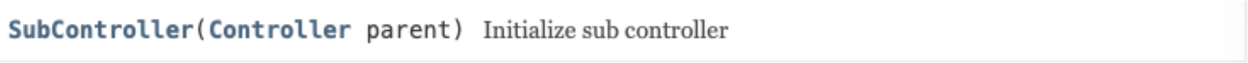
3.6.2 Method



void	unlearnSkill(Pokemua pokemua, Skill skill)	Unlearn Pokemua skill

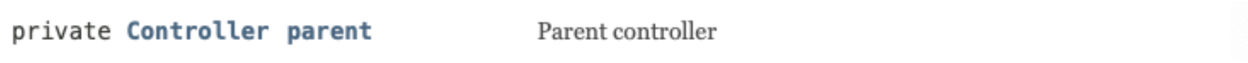
3.7 Class SubController

3.7.1 Constructor



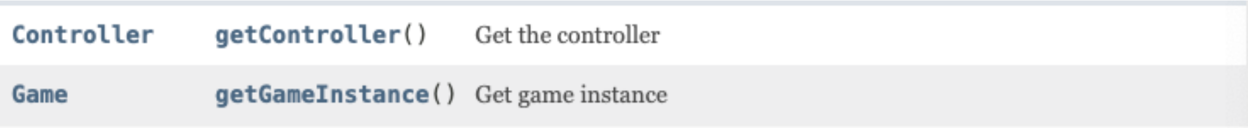
SubController(Controller parent)	Initialize sub controller

3.7.2 Field



private Controller parent	Parent controller

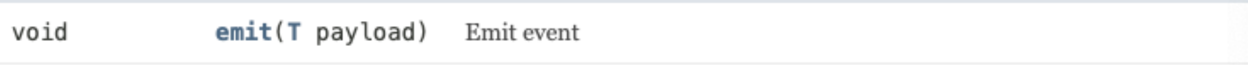
3.7.3 Method



Controller	getController()	Get the controller
Game	getGameInstance()	Get game instance

3.8 Class UIEventEmitter<T extends Event>

3.8.1 Method



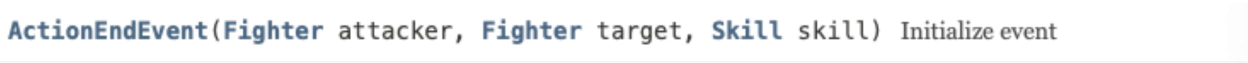
void	emit(T payload)	Emit event

4. Package event

4.1 Package event.fight

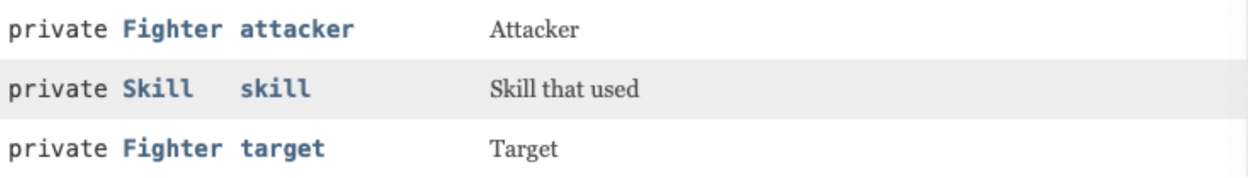
4.1.1 Class ActionEndEvent

4.1.1.1 Constructor



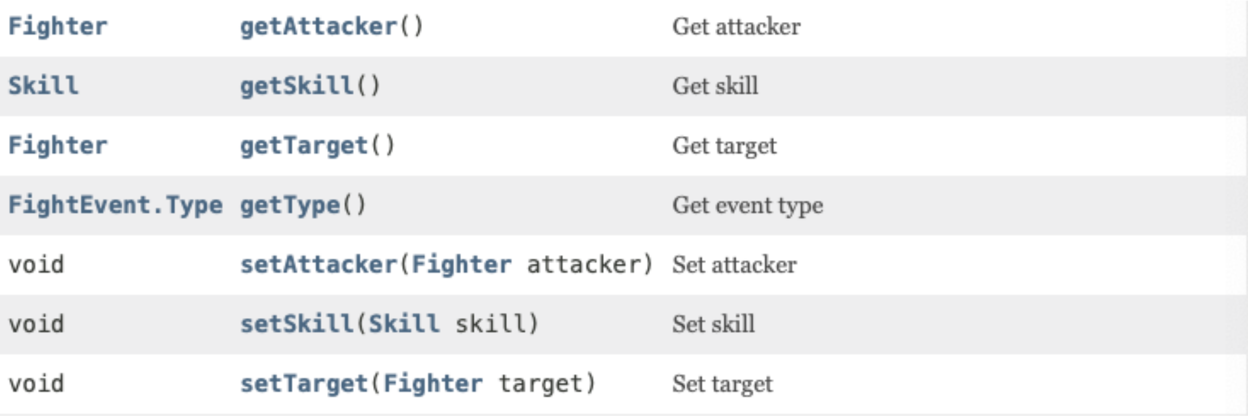
ActionEndEvent(Fighter attacker, Fighter target, Skill skill)	Initialize event

4.1.1.2 Field



private Fighter attacker	Attacker
private Skill skill	Skill that used
private Fighter target	Target

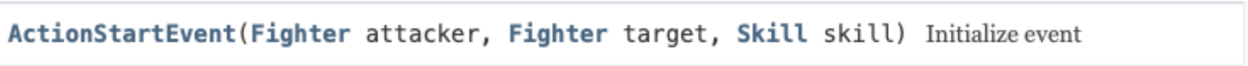
4.1.1.3 Method



Fighter	getAttacker()	Get attacker
Skill	getSkill()	Get skill
Fighter	getTarget()	Get target
FightEvent.Type	getType()	Get event type
void	setAttacker(Fighter attacker)	Set attacker
void	setSkill(Skill skill)	Set skill
void	setTarget(Fighter target)	Set target

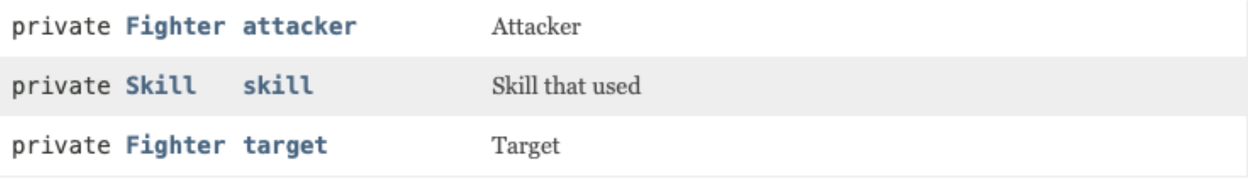
4.1.2 Class ActionStartEvent

4.1.2.1 Constructor



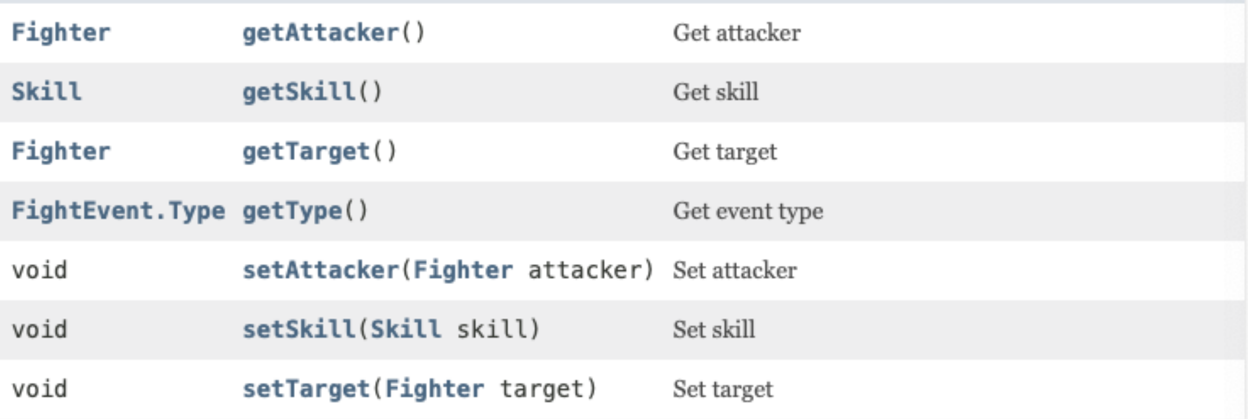
ActionStartEvent(Fighter attacker, Fighter target, Skill skill)	Initialize event

4.1.2.2 Field



private Fighter	attacker	Attacker
private Skill	skill	Skill that used
private Fighter	target	Target

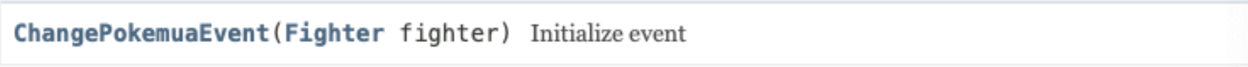
4.1.2.3 Method



Fighter	getAttacker()	Get attacker
Skill	getSkill()	Get skill
Fighter	getTarget()	Get target
FightEvent.Type	getType()	Get event type
void	setAttacker(Fighter attacker)	Set attacker
void	setSkill(Skill skill)	Set skill
void	setTarget(Fighter target)	Set target

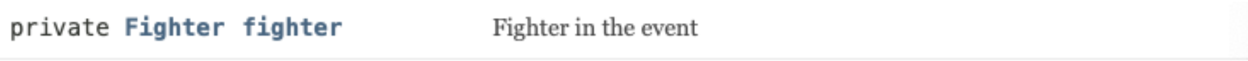
4.1.3 Class ChangePokemuaEvent

4.1.3.1 Constructor



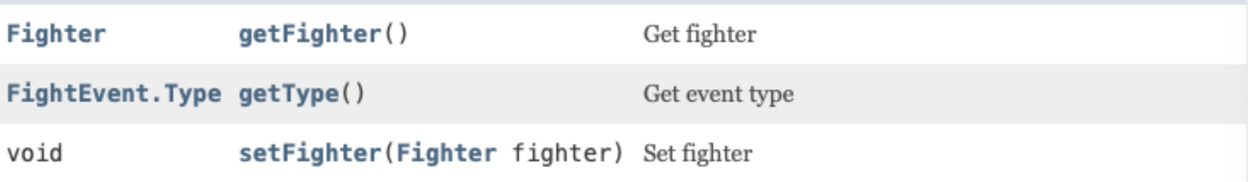
ChangePokemuaEvent(Fighter fighter)	Initialize event

4.1.3.2 Field



private Fighter fighter	Fighter in the event

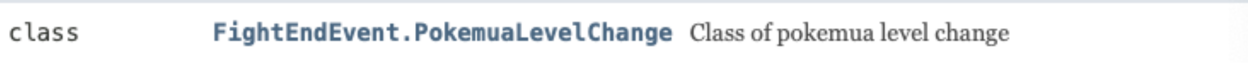
4.1.3.3 Method



Fighter	getFighter()	Get fighter
FightEvent.Type	getType()	Get event type
void	setFighter(Fighter fighter)	Set fighter

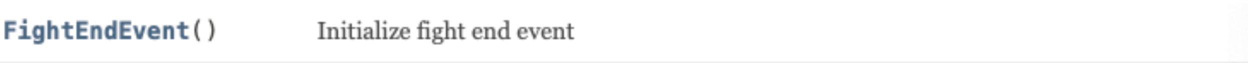
4.1.4 Class FightEndEvent

4.1.4.1 Nested Class



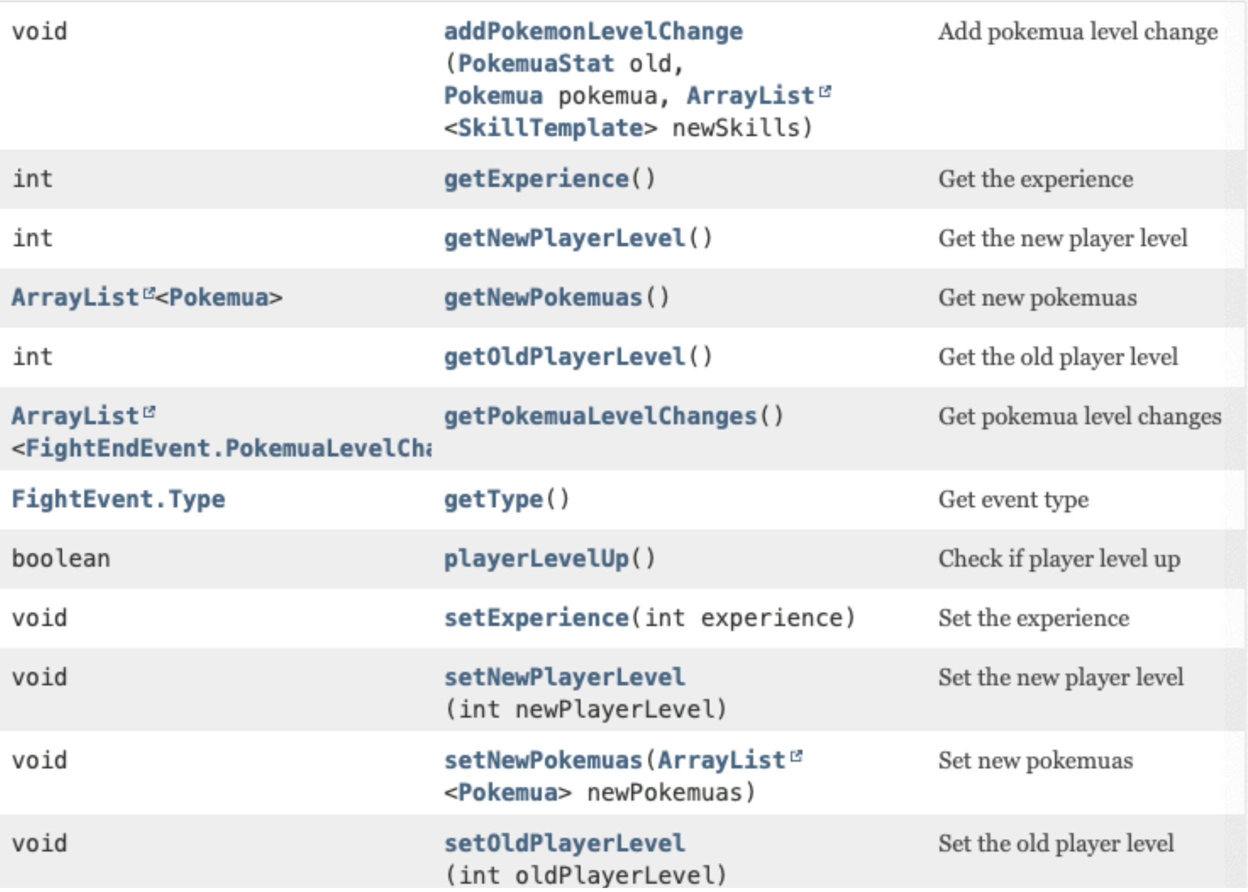
class	FightEndEvent.PokemuaLevelChange	Class of pokemua level change

4.1.4.2 Constructor



FightEndEvent()	Initialize fight end event

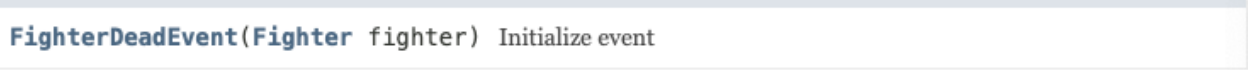
4.1.4.3 Method



void	addPokemonLevelChange (PokemuaStat old, Pokemua pokemua, ArrayList <SkillTemplate> newSkills)	Add pokemua level change
int	getExperience()	Get the experience
int	getNewPlayerLevel()	Get the new player level
ArrayList<Pokemua>	getNewPokemuas()	Get new pokemuas
int	getOldPlayerLevel()	Get the old player level
ArrayList <FightEndEvent.PokemuaLevelCha	getPokemuaLevelChanges()	Get pokemua level changes
FightEvent.Type	getType()	Get event type
boolean	playerLevelUp()	Check if player level up
void	setExperience(int experience)	Set the experience
void	setNewPlayerLevel (int newPlayerLevel)	Set the new player level
void	setNewPokemuas(ArrayList <Pokemua> newPokemuas)	Set new pokemuas
void	setOldPlayerLevel (int oldPlayerLevel)	Set the old player level

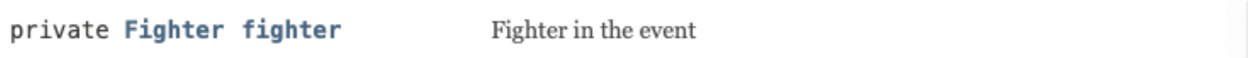
4.1.5 Class FightDeadEvent

4.1.5.1 Constructor



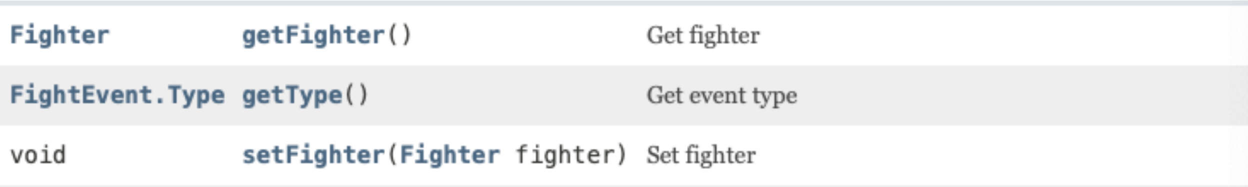
FighterDeadEvent(Fighter fighter)	Initialize event

4.1.5.2 Field



private Fighter fighter	Fighter in the event

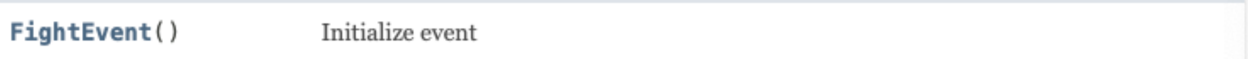
4.1.5.3 Method



Fighter	getFighter()	Get fighter
FightEvent.Type	getType()	Get event type
void	setFighter(Fighter fighter)	Set fighter

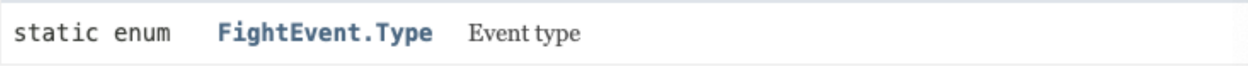
4.1.6 Class FightEvent

4.1.6.1 Constructor



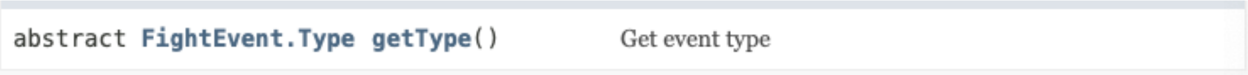
FightEvent()	Initialize event

4.1.6.2 Nested Class



static enum	FightEvent.Type	Event type

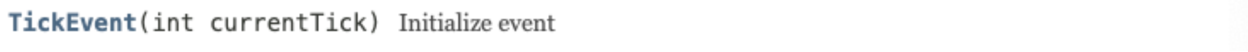
4.1.6.3 Method



abstract FightEvent.Type getType()	Get event type

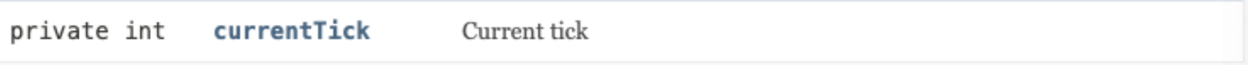
4.1.7 Class TickEvent

4.1.7.1 Constructor



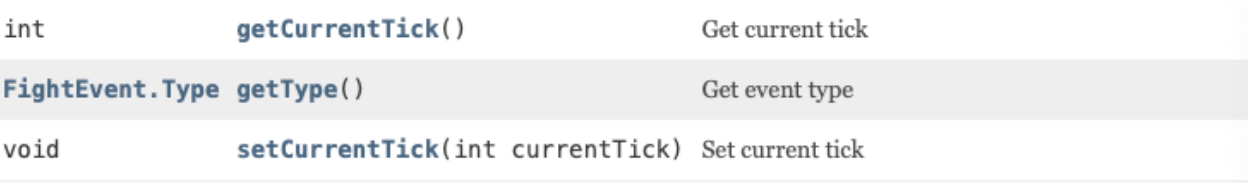
TickEvent(int currentTick)	Initialize event

4.1.7.2 Field



private int	currentTick	Current tick

4.1.7.3 Method

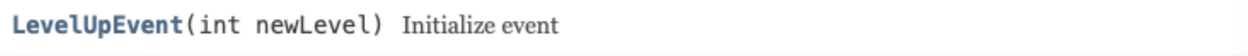


int	getCurrentTick()	Get current tick
FightEvent.Type	getType()	Get event type
void	setCurrentTick(int currentTick)	Set current tick

4.2 Package event.player

4.2.1 Class LevelUpEvent

4.2.1.1 Constructor

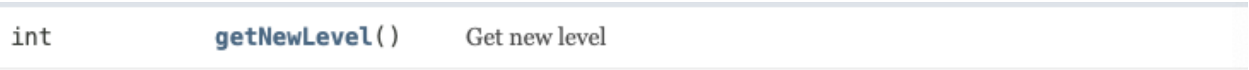


LevelUpEvent(int newLevel)	Initialize event

4.2.1.2 Field

private int	newLevel	New level

4.2.1.3 Method



int	getNewLevel()	Get new level

4.2.2 Class PlayerEvent

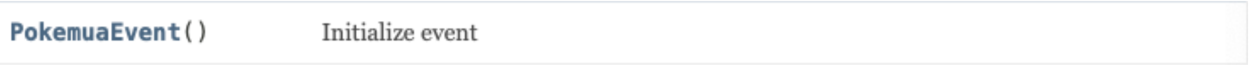
4.3 Package event.pokemua

4.3.1 Class Pokemua Event

4.3.1.1 Nested Class

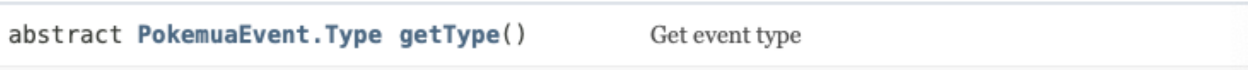
static enum	PokemuaEvent.Type	Event type

4.3.1.2 Constructor



PokemuaEvent()	Initialize event

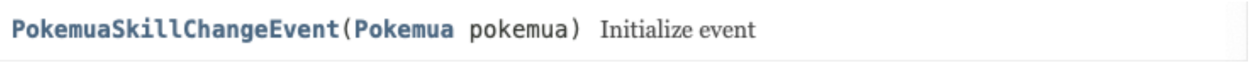
4.3.1.3 Method



abstract PokemuaEvent.Type	getType()	Get event type

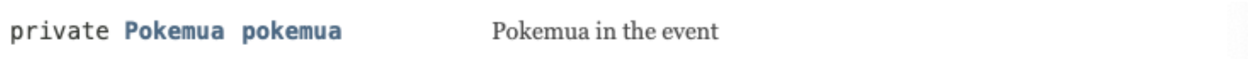
4.3.2 Class PokemuaSkillChangeEvent

4.3.2.1 Constructor



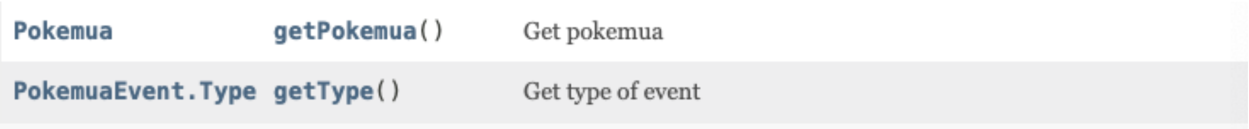
PokemuaSkillChangeEvent(Pokemua pokemua)	Initialize event

4.2.2.3 Field



private Pokemua	pokemua	Pokemua in the event

4.2.2.3 Method



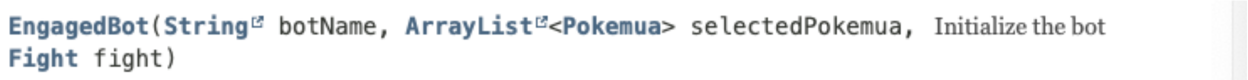
Pokemua	getPokemua()	Get pokemua
PokemuaEvent.Type	getType()	Get type of event

5. Package logic

5.1 Package logic.fighter

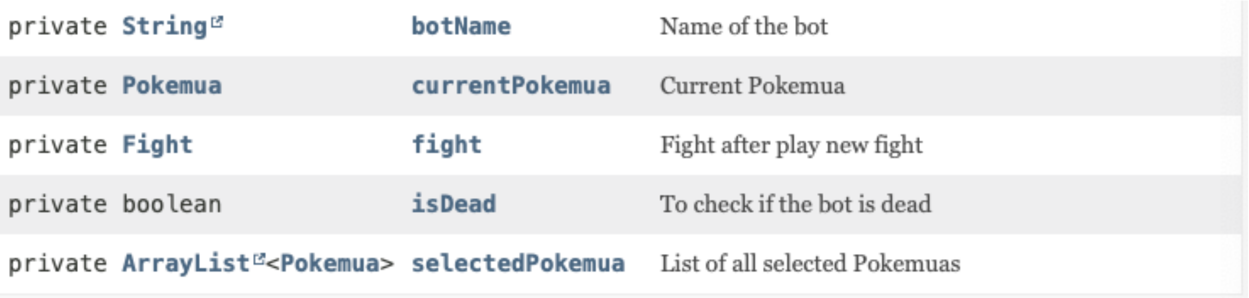
5.1.1 Class EngagedBot

5.1.1.1 Constructor



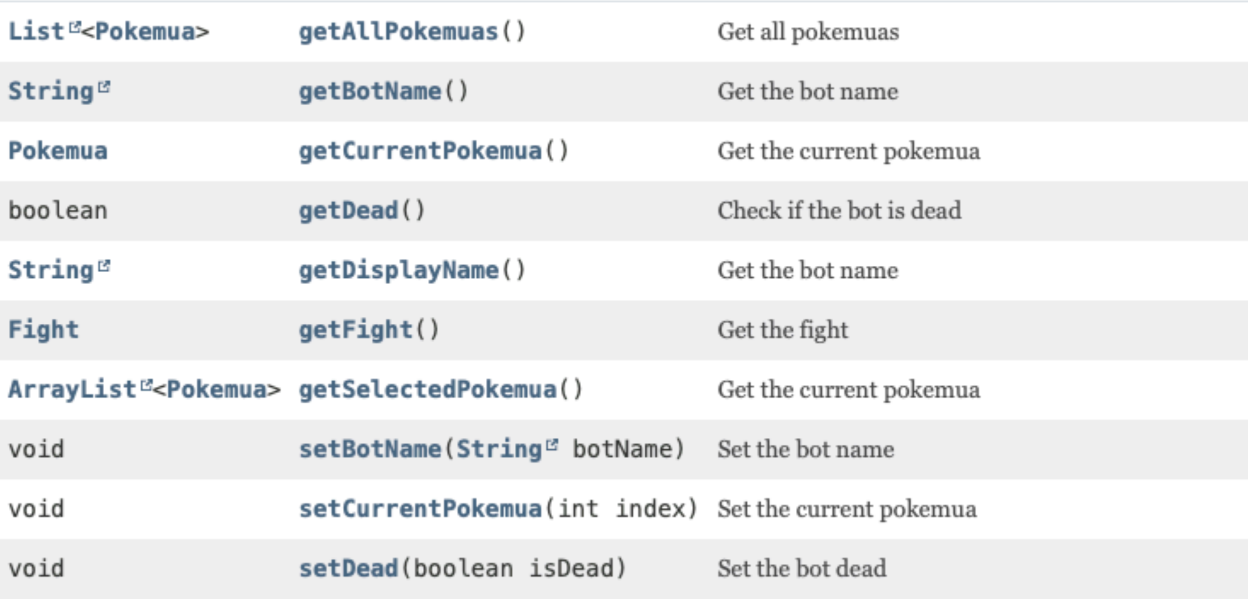
`EngagedBot(String botName, ArrayList<Pokemua> selectedPokemua, Fight fight)`	Initialize the bot

5.1.1.2 Field



`private String`	`botName`	Name of the bot
`private Pokemua`	`currentPokemua`	Current Pokemua
`private Fight`	`fight`	Fight after play new fight
`private boolean`	`isDead`	To check if the bot is dead
`private ArrayList<Pokemua>`	`selectedPokemua`	List of all selected Pokemuas

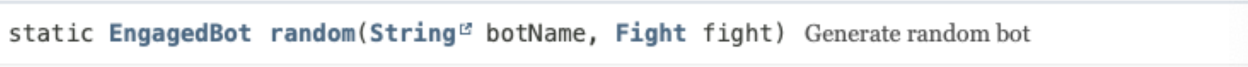
5.1.1.3 Method



`List<Pokemua>`	`getAllPokemuas()`	Get all pokemuas
`String`	`getBotName()`	Get the bot name
`Pokemua`	`getCurrentPokemua()`	Get the current pokemua
`boolean`	`getDead()`	Check if the bot is dead
`String`	`getDisplayName()`	Get the bot name
`Fight`	`getFight()`	Get the fight
`ArrayList<Pokemua>`	`getSelectedPokemua()`	Get the current pokemua
`void`	`setBotName(String botName)`	Set the bot name
`void`	`setCurrentPokemua(int index)`	Set the current pokemua
`void`	`setDead(boolean isDead)`	Set the bot dead

5.1.2 Class EngagedBotRandomizer

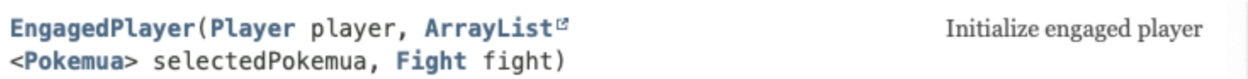
5.1.2.1 Method



static EngagedBot	random(String botName, Fight fight)	Generate random bot

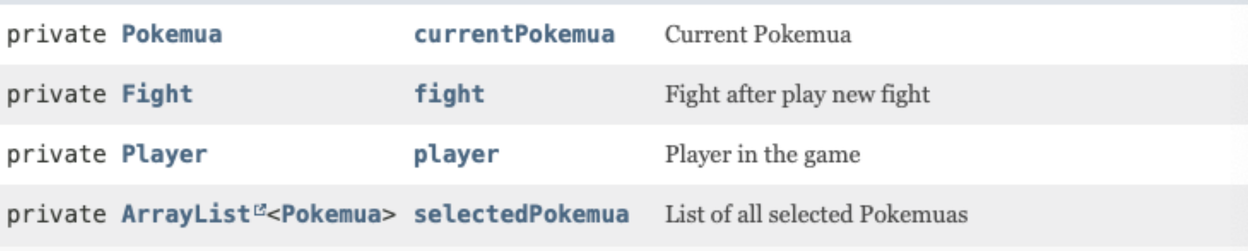
5.1.3 Class Engaged Player

5.1.3.1 Constructor



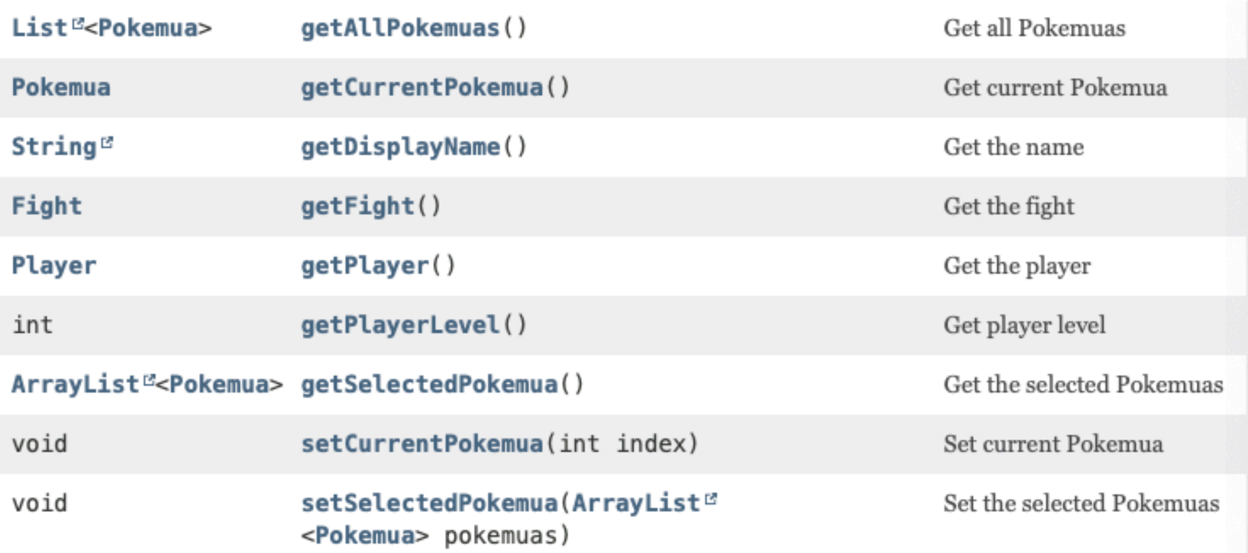
EngagedPlayer(Player player, ArrayList<Pokemua> selectedPokemua, Fight fight)	Initialize engaged player

5.1.3.2 Field



private Pokemua	currentPokemua	Current Pokemua
private Fight	fight	Fight after play new fight
private Player	player	Player in the game
private ArrayList<Pokemua>	selectedPokemua	List of all selected Pokemuas

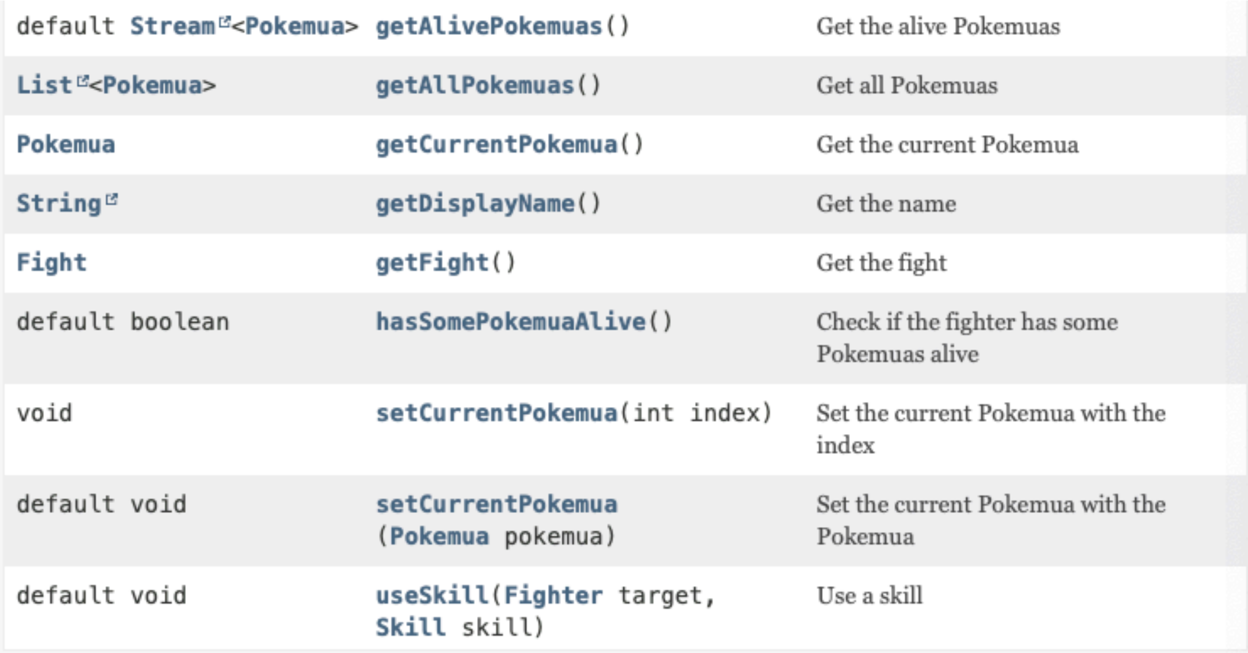
5.1.3.3 Method



List<Pokemua>	getAllPokemuas()	Get all Pokemuas
Pokemua	getCurrentPokemua()	Get current Pokemua
String	getDisplayName()	Get the name
Fight	getFight()	Get the fight
Player	getPlayer()	Get the player
int	getPlayerLevel()	Get player level
ArrayList<Pokemua>	getSelectedPokemua()	Get the selected Pokemuas
void	setCurrentPokemua(int index)	Set current Pokemua
void	setSelectedPokemua(ArrayList<Pokemua> pokemuas)	Set the selected Pokemuas

5.1.4 Interface Fighter

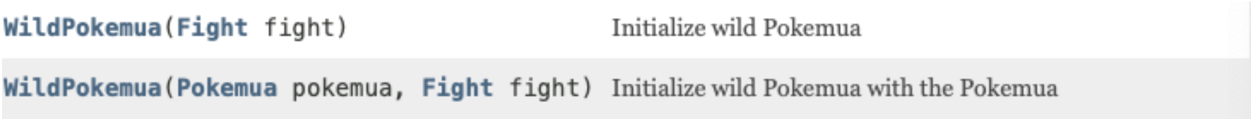
5.1.4.1 Method



default Stream<Pokemua>	getAlivePokemuas()	Get the alive Pokemuas
List<Pokemua>	getAllPokemuas()	Get all Pokemuas
Pokemua	getCurrentPokemua()	Get the current Pokemua
String	getDisplayName()	Get the name
Fight	getFight()	Get the fight
default boolean	hasSomePokemuaAlive()	Check if the fighter has some Pokemuas alive
void	setCurrentPokemua(int index)	Set the current Pokemua with the index
default void	setCurrentPokemua (Pokemua pokemua)	Set the current Pokemua with the Pokemua
default void	useSkill(Fighter target, Skill skill)	Use a skill

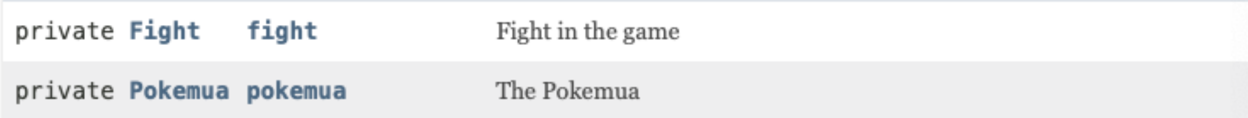
5.1.5 Class WildPokemua

5.1.5.1 Constructor



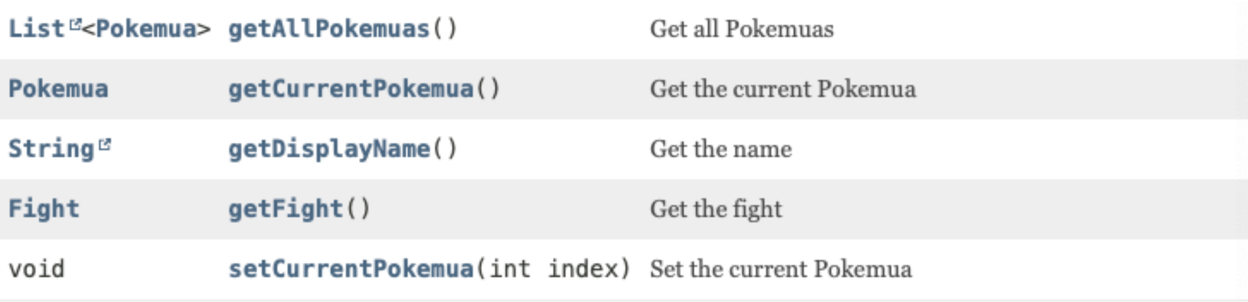
WildPokemua(Fight fight)	Initialize wild Pokemua
WildPokemua(Pokemua pokemua, Fight fight)	Initialize wild Pokemua with the Pokemua

5.1.5.2 Field



private Fight	fight	Fight in the game
private Pokemua	pokemua	The Pokemua

5.1.5.3 Method

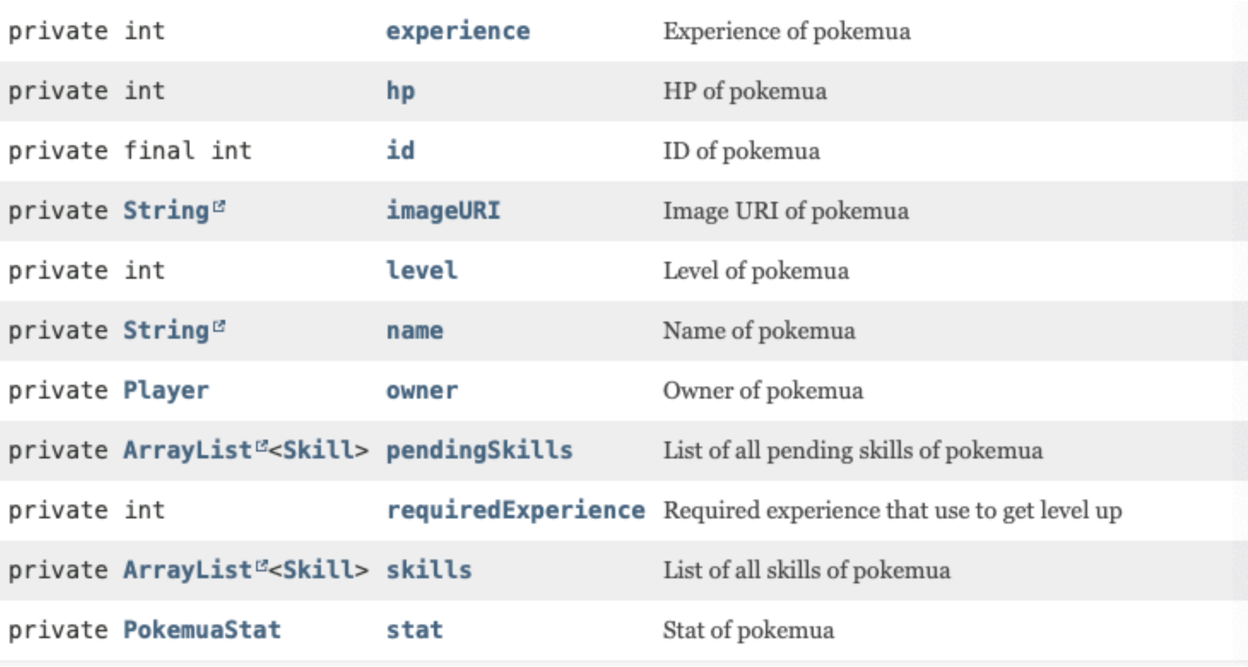


List<Pokemua>	getAllPokemuas()	Get all Pokemuas
Pokemua	getCurrentPokemua()	Get the current Pokemua
String	getDisplayName()	Get the name
Fight	getFight()	Get the fight
void	setCurrentPokemua(int index)	Set the current Pokemua

5.2 Package logic.pokemua

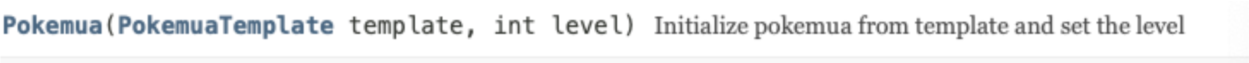
5.2.1 Class Pokemua

5.2.1.1 Field



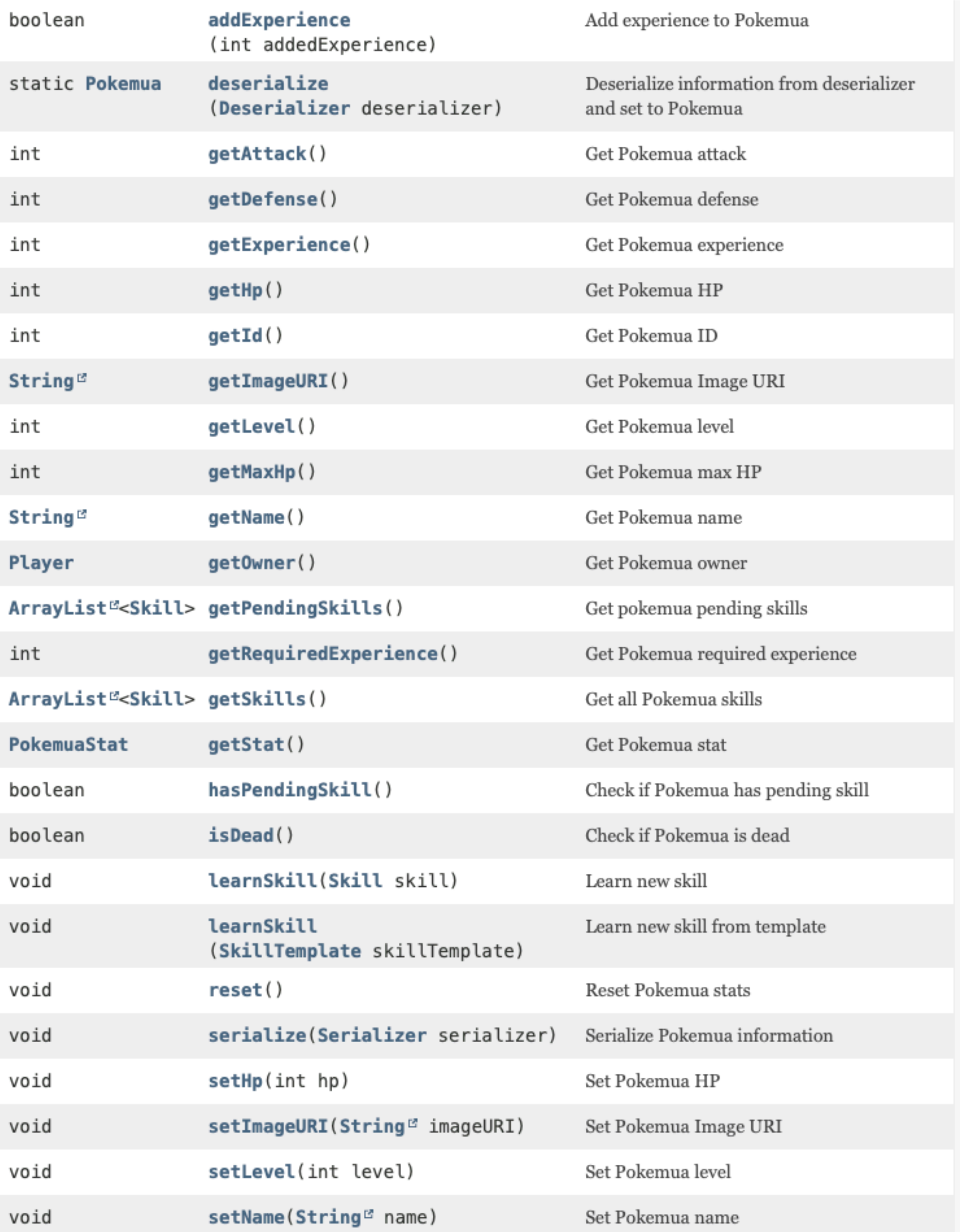
private int	experience	Experience of pokemua
private int	hp	HP of pokemua
private final int	id	ID of pokemua
private String	imageURI	Image URI of pokemua
private int	level	Level of pokemua
private String	name	Name of pokemua
private Player	owner	Owner of pokemua
private ArrayList<Skill>	pendingSkills	List of all pending skills of pokemua
private int	requiredExperience	Required experience that use to get level up
private ArrayList<Skill>	skills	List of all skills of pokemua
private PokemuaStat	stat	Stat of pokemua

5.2.1.2 Constructor

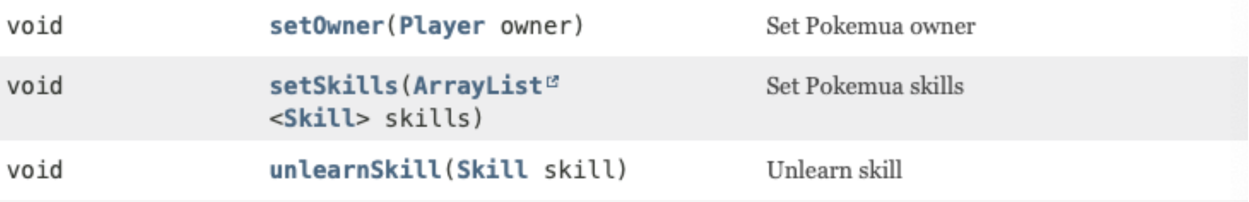


Pokemua(PokemuaTemplate template, int level)	Initialize pokemua from template and set the level

5.2.1.3 Method



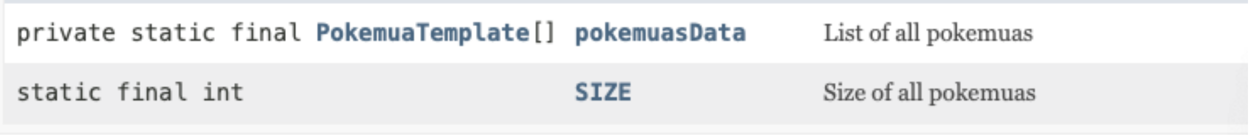
boolean	addExperience (int addedExperience)	Add experience to Pokemua
static Pokemua	deserialize (Deserializer deserializer)	Deserialize information from deserializer and set to Pokemua
int	getAttack()	Get Pokemua attack
int	getDefense()	Get Pokemua defense
int	getExperience()	Get Pokemua experience
int	getHp()	Get Pokemua HP
int	getId()	Get Pokemua ID
String	getImageURI()	Get Pokemua Image URI
int	getLevel()	Get Pokemua level
int	getMaxHp()	Get Pokemua max HP
String	getName()	Get Pokemua name
Player	getOwner()	Get Pokemua owner
ArrayList<Skill>	getPendingSkills()	Get pokemua pending skills
int	getRequiredExperience()	Get Pokemua required experience
ArrayList<Skill>	getSkills()	Get all Pokemua skills
PokemuaStat	getStat()	Get Pokemua stat
boolean	hasPendingSkill()	Check if Pokemua has pending skill
boolean	isDead()	Check if Pokemua is dead
void	learnSkill(Skill skill)	Learn new skill
void	learnSkill (SkillTemplate skillTemplate)	Learn new skill from template
void	reset()	Reset Pokemua stats
void	serialize(Serializer serializer)	Serialize Pokemua information
void	setHp(int hp)	Set Pokemua HP
void	setImageURI(String imageURI)	Set Pokemua Image URI
void	setLevel(int level)	Set Pokemua level
void	setName(String name)	Set Pokemua name



void	setOwner(Player owner)	Set Pokemua owner
void	setSkills(ArrayList<Skill> skills)	Set Pokemua skills
void	unlearnSkill(Skill skill)	Unlearn skill

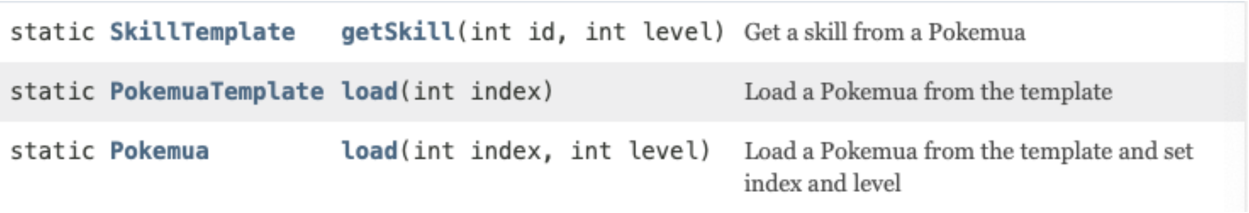
5.2.2 Class PokemuaLoader

5.2.2.1 Field



private static final PokemuaTemplate[]	pokemuasData	List of all pokemuas
static final int	SIZE	Size of all pokemuas

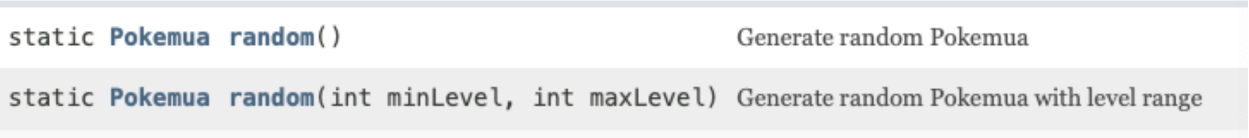
5.2.2.2 Method



static SkillTemplate	getSkill(int id, int level)	Get a skill from a Pokemua
static PokemuaTemplate	load(int index)	Load a Pokemua from the template
static Pokemua	load(int index, int level)	Load a Pokemua from the template and set index and level

5.2.3 Class PokemuaRandomizer

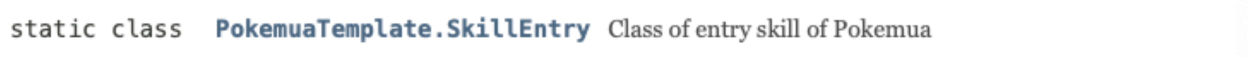
5.2.3.1 Method



static Pokemua	random()	Generate random Pokemua
static Pokemua	random(int minLevel, int maxLevel)	Generate random Pokemua with level range

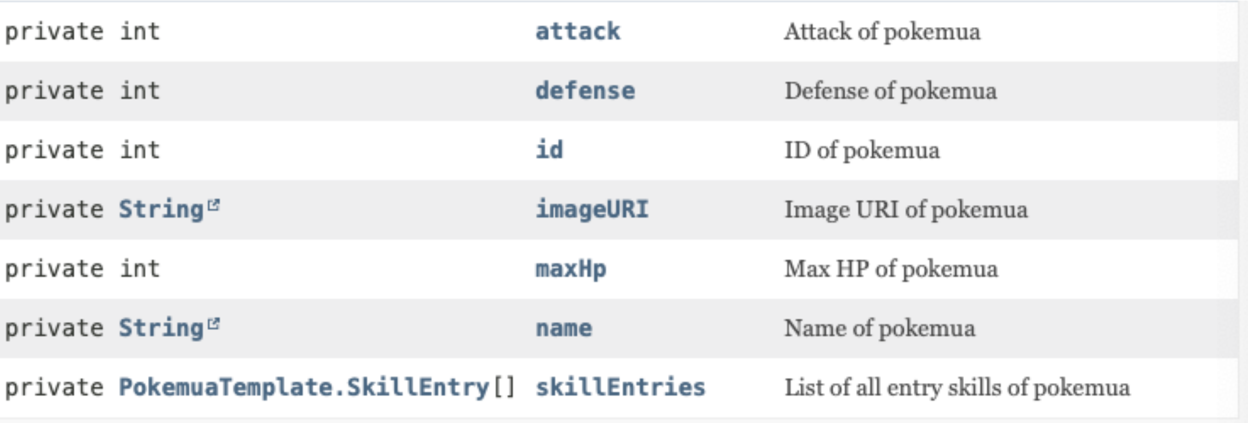
5.2.4 Class PokemuaTemplate

5.2.4.1 Nested Class



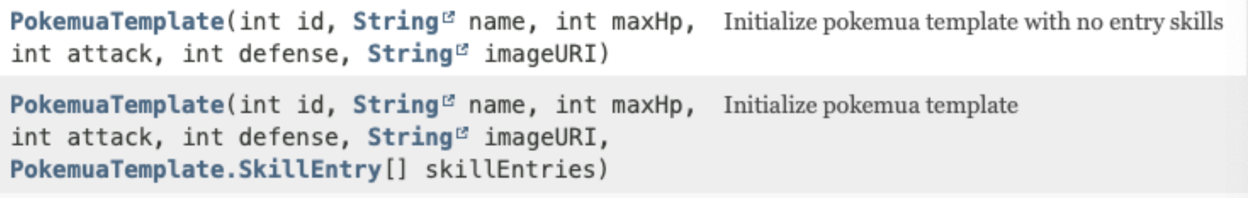
static class	PokemuaTemplate.SkillEntry	Class of entry skill of Pokemua

5.2.4.2 Field



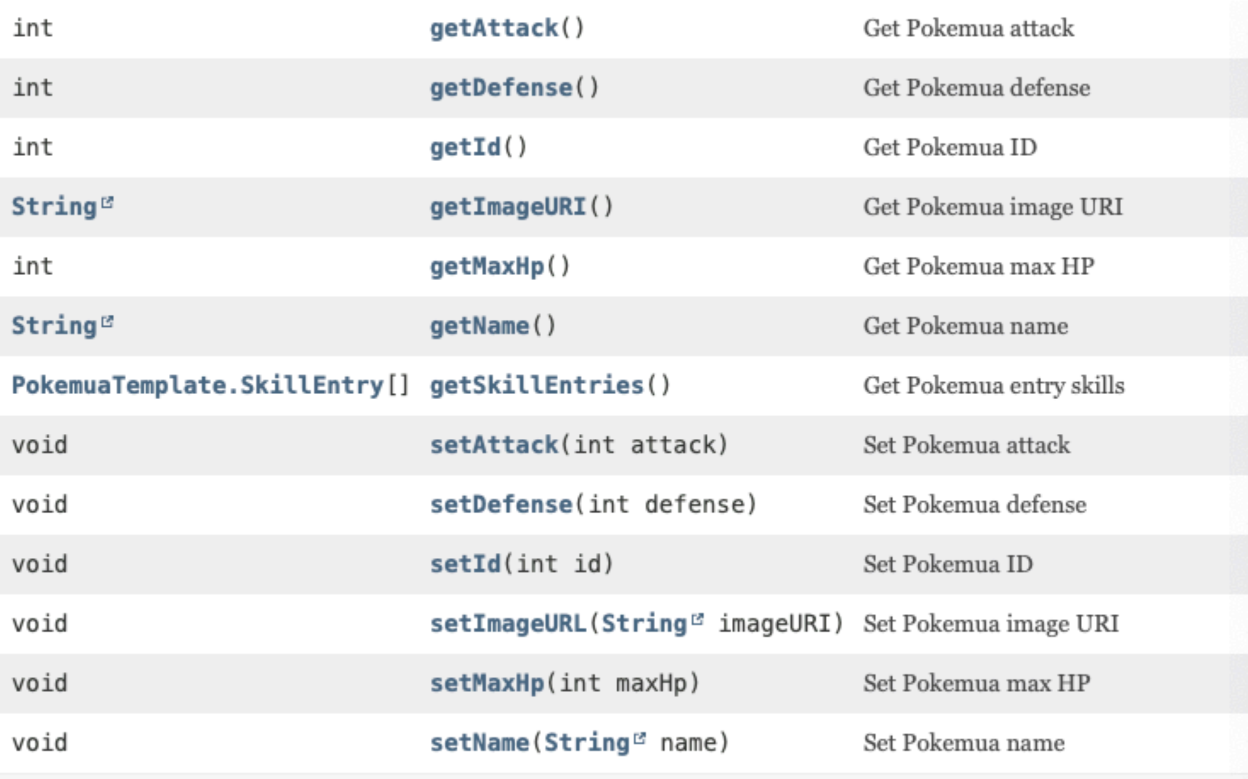
private int	attack	Attack of pokemua
private int	defense	Defense of pokemua
private int	id	ID of pokemua
private String	imageURI	Image URI of pokemua
private int	maxHp	Max HP of pokemua
private String	name	Name of pokemua
private PokemuaTemplate.SkillEntry[]	skillEntries	List of all entry skills of pokemua

5.2.4.3 Constructor



PokemuaTemplate(int id, String name, int maxHp, int attack, int defense, String imageURI)	Initialize pokemua template with no entry skills
PokemuaTemplate(int id, String name, int maxHp, int attack, int defense, String imageURI, PokemuaTemplate.SkillEntry[] skillEntries)	Initialize pokemua template

5.2.4.4 Method

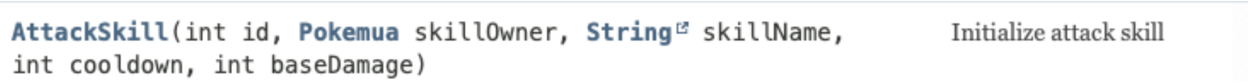


int	getAttack()	Get Pokemua attack
int	getDefense()	Get Pokemua defense
int	getId()	Get Pokemua ID
String	getImageURI()	Get Pokemua image URI
int	getMaxHp()	Get Pokemua max HP
String	getName()	Get Pokemua name
PokemuaTemplate.SkillEntry[]	getSkillEntries()	Get Pokemua entry skills
void	setAttack(int attack)	Set Pokemua attack
void	setDefense(int defense)	Set Pokemua defense
void	setId(int id)	Set Pokemua ID
void	setImageURL(String imageURI)	Set Pokemua image URI
void	setMaxHp(int maxHp)	Set Pokemua max HP
void	setName(String name)	Set Pokemua name

5.3 Package logic.skill

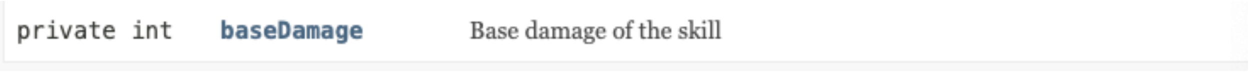
5.3.1 Class AttackSkill

5.3.1.1 Constructor



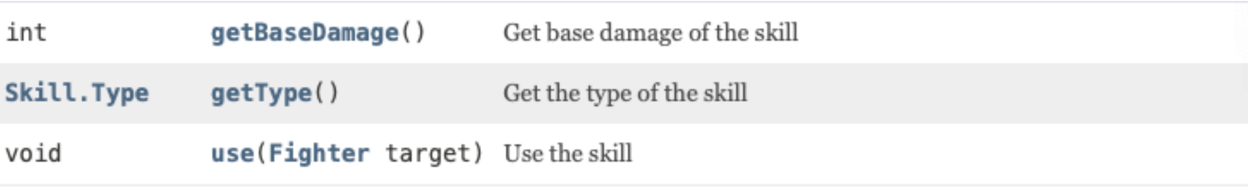
AttackSkill(int id, Pokemua skillOwner, String skillName, int cooldown, int baseDamage)	Initialize attack skill

5.3.1.2 Field



private int	baseDamage	Base damage of the skill

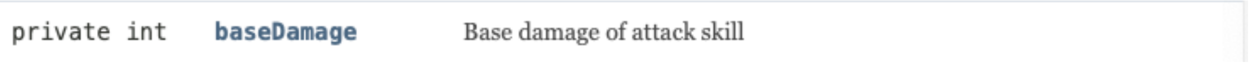
5.3.1.3 Method



int	getBaseDamage()	Get base damage of the skill
Skill.Type	getType()	Get the type of the skill
void	use(Fighter target)	Use the skill

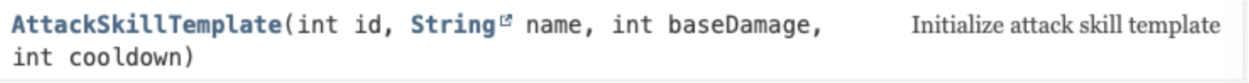
5.3.2 Class AttackSkillTemplate

5.3.2.1 Field



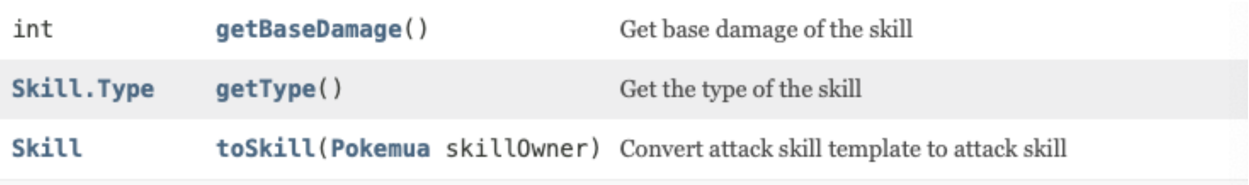
private int	baseDamage	Base damage of attack skill

5.3.2.1 Constructor



AttackSkillTemplate(int id, String name, int baseDamage, int cooldown)	Initialize attack skill template

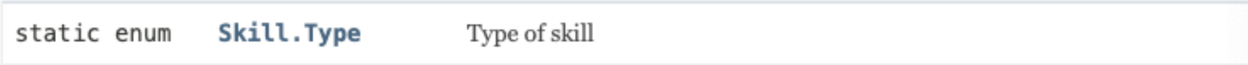
5.3.2.1 Method



int	getBaseDamage()	Get base damage of the skill
Skill.Type	getType()	Get the type of the skill
Skill	toSkill(Pokemua skillOwner)	Convert attack skill template to attack skill

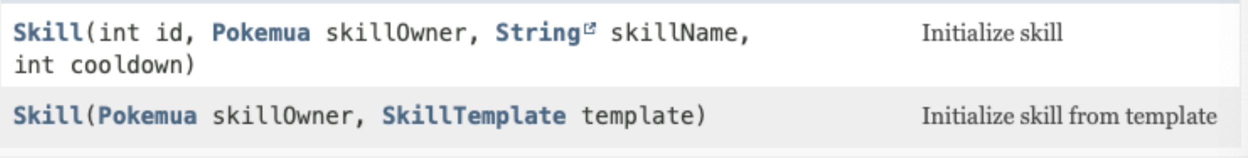
5.3.3 Class Skill

5.3.3.1 Nested Class



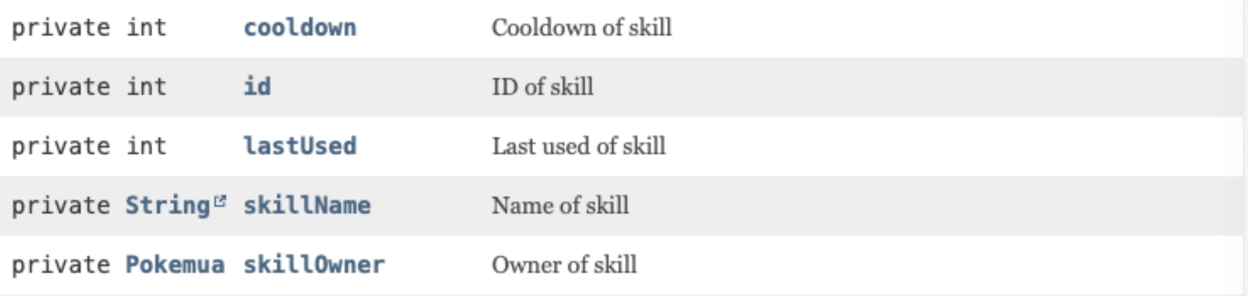
static enum	Skill.Type	Type of skill

5.3.3.2 Constructor



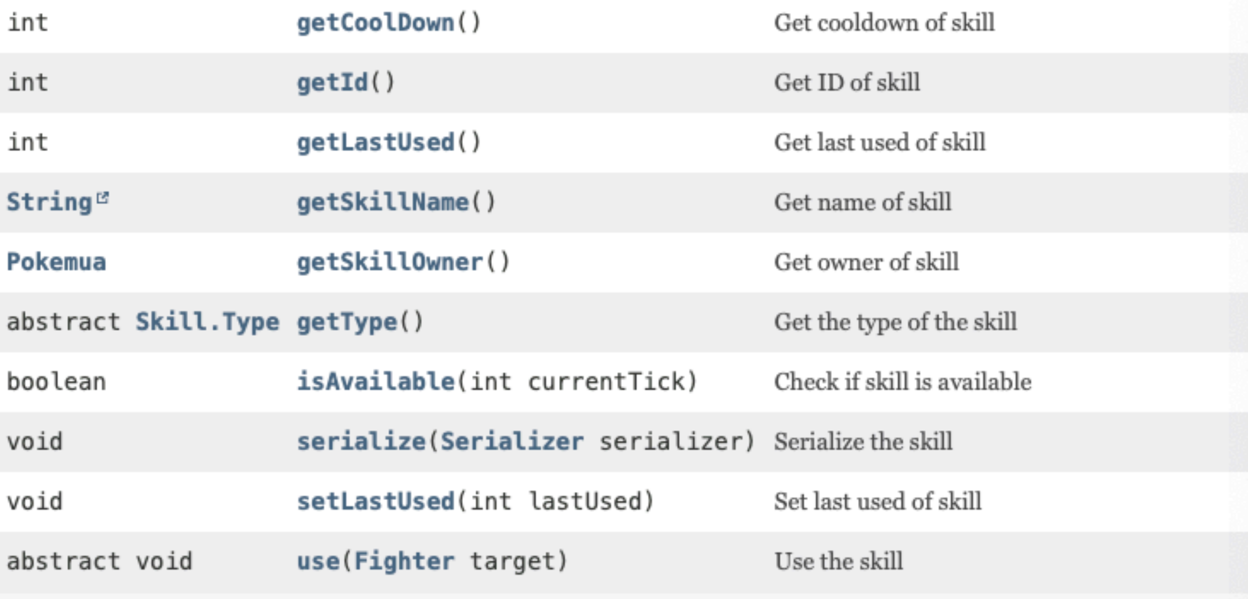
Skill(int id, Pokemua skillOwner, String skillName, int cooldown)	Initialize skill
Skill(Pokemua skillOwner, SkillTemplate template)	Initialize skill from template

5.3.3.3 Field



private int	cooldown	Cooldown of skill
private int	id	ID of skill
private int	lastUsed	Last used of skill
private String	skillName	Name of skill
private Pokemua	skillOwner	Owner of skill

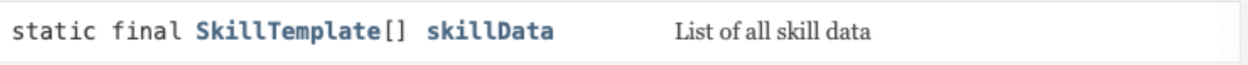
5.3.3.4 Method



int	getCoolDown()	Get cooldown of skill
int	getId()	Get ID of skill
int	getLastUsed()	Get last used of skill
String	getSkillName()	Get name of skill
Pokemua	getSkillOwner()	Get owner of skill
abstract Skill.Type	getType()	Get the type of the skill
boolean	isAvailable(int currentTick)	Check if skill is available
void	serialize(Serializer serializer)	Serialize the skill
void	setLastUsed(int lastUsed)	Set last used of skill
abstract void	use(Fighter target)	Use the skill

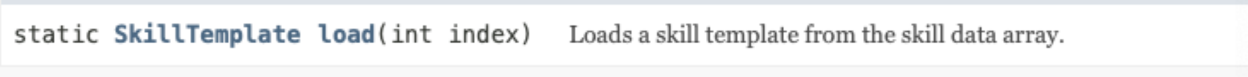
5.3.4 Class SkillLoader

5.3.4.1 Field



static final SkillTemplate[]	skillData	List of all skill data

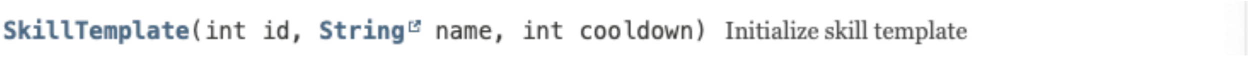
5.3.4.2 Method



static SkillTemplate load(int index)	Loads a skill template from the skill data array.

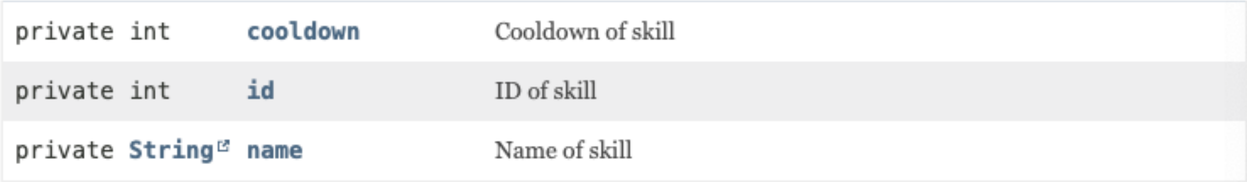
5.3.5 Class SkillTemplate

5.3.5.1 Constructor



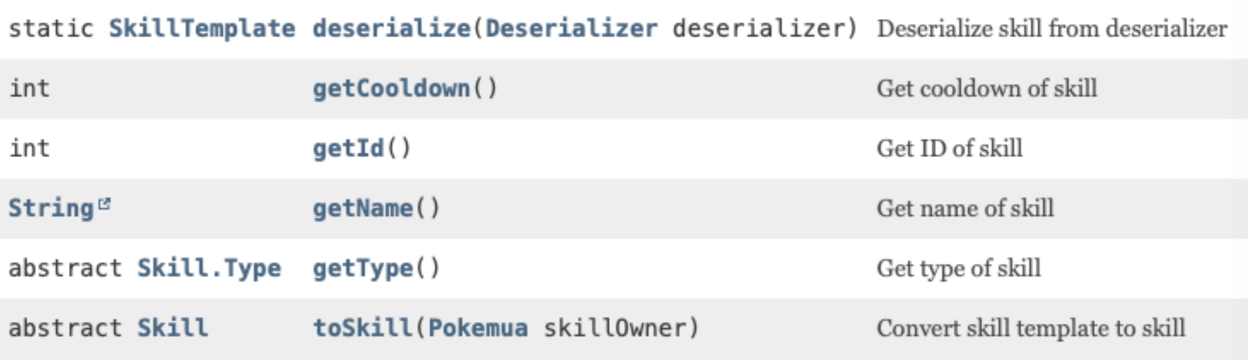
SkillTemplate(int id, String name, int cooldown)	Initialize skill template

5.3.5.2 Field



private int	cooldown	Cooldown of skill
private int	id	ID of skill
private String	name	Name of skill

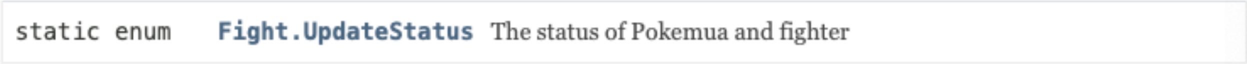
5.3.5.3 Method



static SkillTemplate	deserialize(Deserializer deserializer)	Deserialize skill from deserializer
int	getCooldown()	Get cooldown of skill
int	getId()	Get ID of skill
String	getName()	Get name of skill
abstract Skill.Type	getType()	Get type of skill
abstract Skill	toSkill(Pokemua skillOwner)	Convert skill template to skill

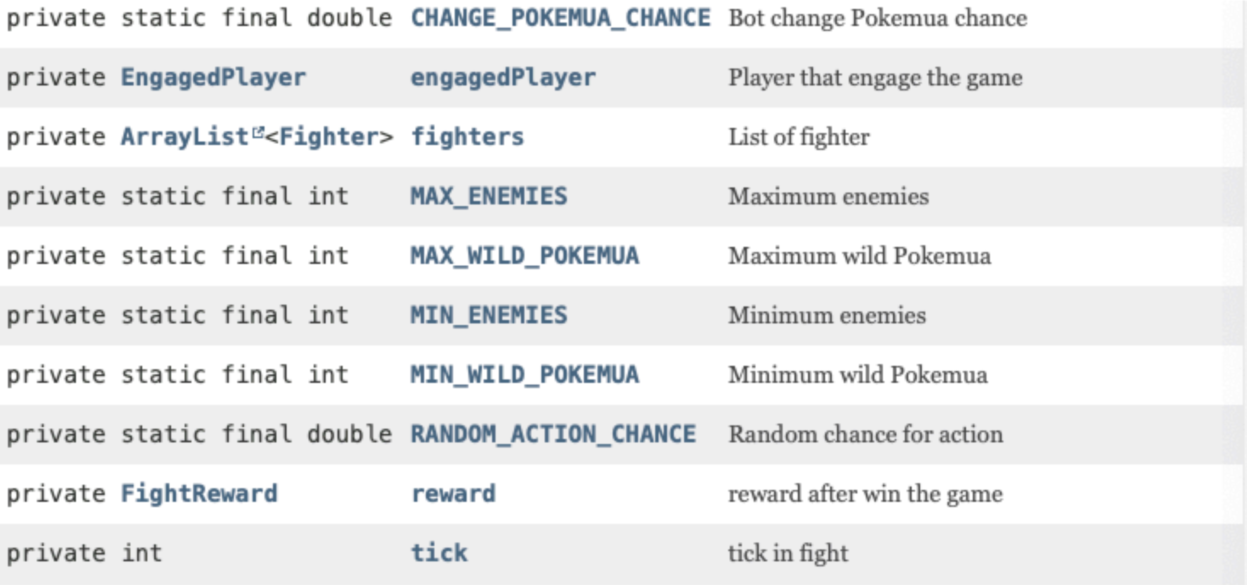
5.4 Class Fight

5.4.1 Nested Class



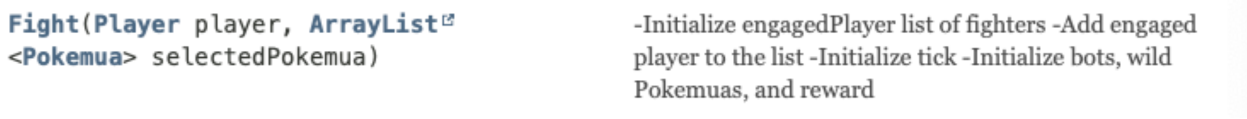
static enum	Fight.UpdateStatus	The status of Pokemua and fighter

5.4.2 Field



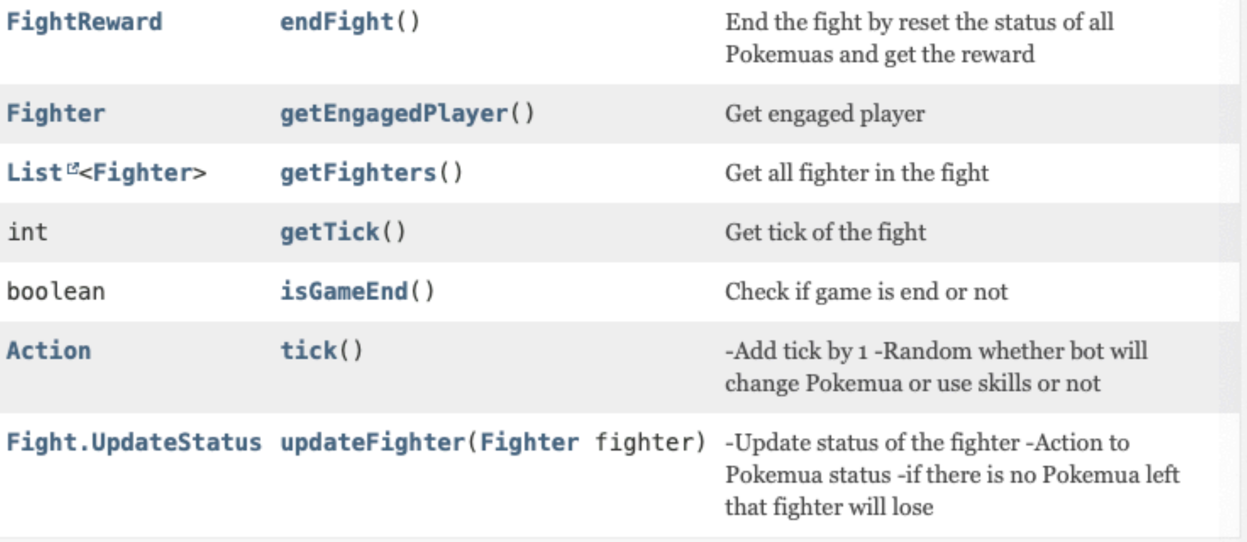
private static final double	CHANGE_POKEMUA_CHANCE	Bot change Pokemua chance
private EngagedPlayer	engagedPlayer	Player that engage the game
private ArrayList<Fighter>	fighters	List of fighter
private static final int	MAX_ENEMIES	Maximum enemies
private static final int	MAX_WILD_POKEMUA	Maximum wild Pokemua
private static final int	MIN_ENEMIES	Minimum enemies
private static final int	MIN_WILD_POKEMUA	Minimum wild Pokemua
private static final double	RANDOM_ACTION_CHANCE	Random chance for action
private FightReward	reward	reward after win the game
private int	tick	tick in fight

5.4.3 Constructor



Fight(Player player, ArrayList<Pokemua> selectedPokemua)	-Initialize engagedPlayer list of fighters -Add engaged player to the list -Initialize tick -Initialize bots, wild Pokemuas, and reward

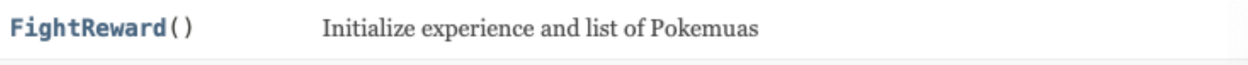
5.4.4 Method



FightReward	endFight()	End the fight by reset the status of all Pokemuas and get the reward
Fighter	getEngagedPlayer()	Get engaged player
List<Fighter>	getFighters()	Get all fighter in the fight
int	getTick()	Get tick of the fight
boolean	isGameEnd()	Check if game is end or not
Action	tick()	-Add tick by 1 -Random whether bot will change Pokemua or use skills or not
Fight.UpdateStatus	updateFighter(Fighter fighter)	-Update status of the fighter -Action to Pokemua status -if there is no Pokemua left that fighter will lose

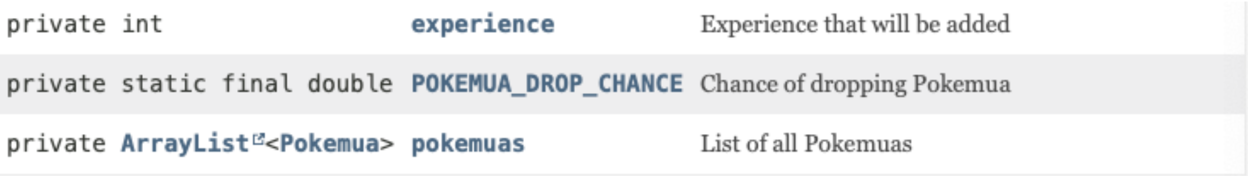
5.5 Class FightReward

5.5.1 Constructor



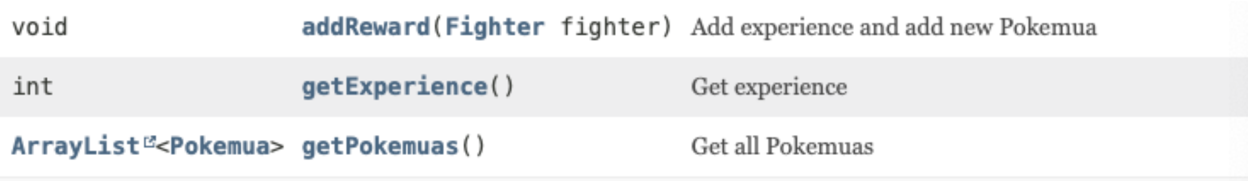
FightReward()	Initialize experience and list of Pokemuas

5.5.2 Field



private int	experience	Experience that will be added
private static final double	POKEMUA_DROP_CHANCE	Chance of dropping Pokemua
private ArrayList<Pokemua>	pokemuas	List of all Pokemuas

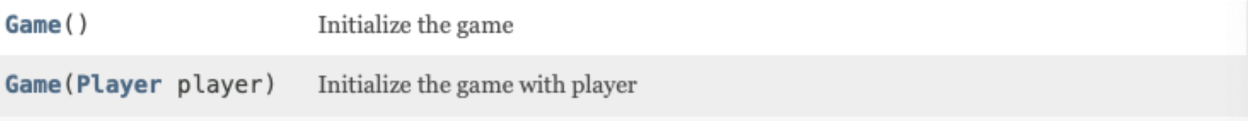
5.5.3 Method



void	addReward(Fighter fighter)	Add experience and add new Pokemua
int	getExperience()	Get experience
ArrayList<Pokemua>	getPokemuas()	Get all Pokemuas

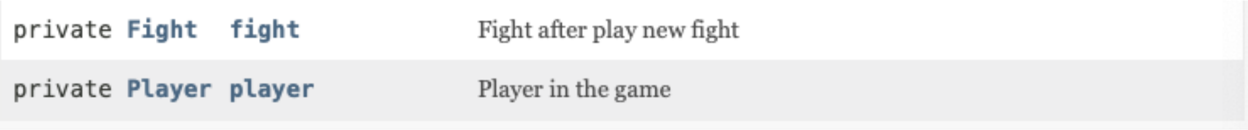
5.6 Class Game

5.6.1 Constructor



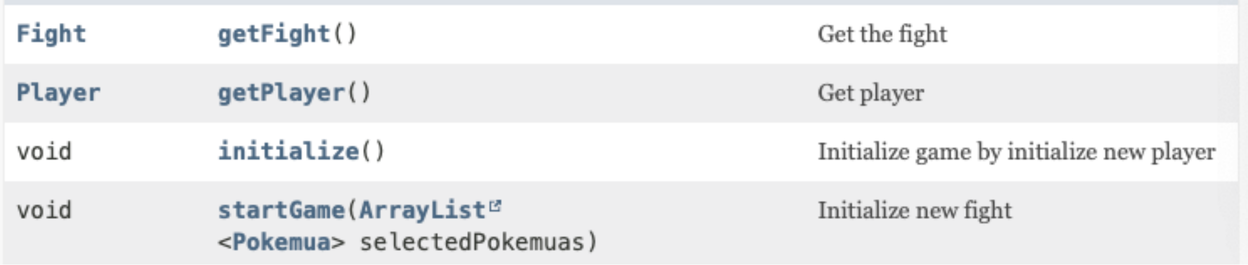
Game()	Initialize the game
Game(Player player)	Initialize the game with player

5.6.2 Field



private Fight	fight	Fight after play new fight
private Player	player	Player in the game

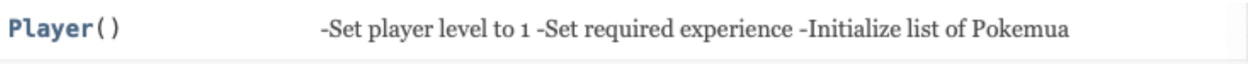
5.6.3 Method



Fight	getFight()	Get the fight
Player	getPlayer()	Get player
void	initialize()	Initialize game by initialize new player
void	startGame(ArrayList<Pokemua> selectedPokemuas)	Initialize new fight

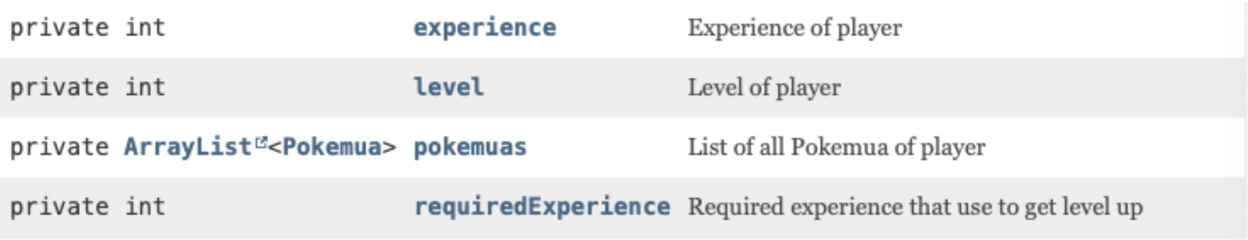
5.7 Class Player

5.7.1 Constructor



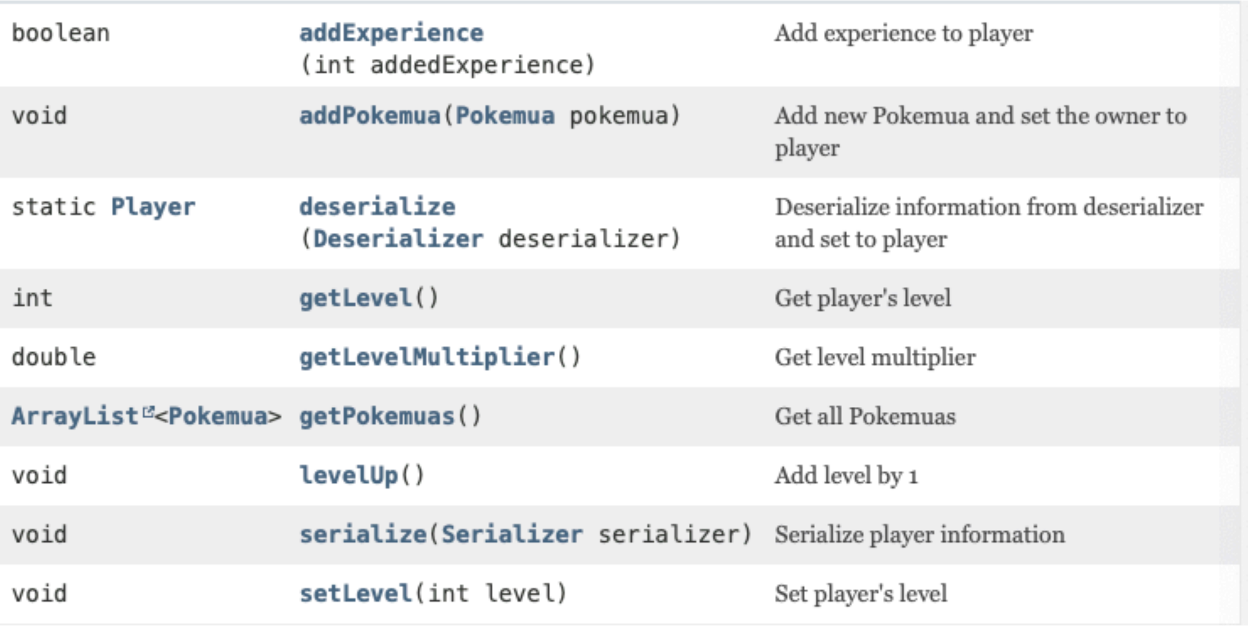
Player()	-Set player level to 1 -Set required experience -Initialize list of Pokemua

5.7.2 Field



private int	experience	Experience of player
private int	level	Level of player
private ArrayList<Pokemua>	pokemuas	List of all Pokemua of player
private int	requiredExperience	Required experience that use to get level up

5.7.3 Method

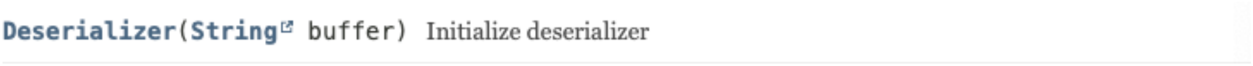


boolean	addExperience (int addedExperience)	Add experience to player
void	addPokemua(Pokemua pokemua)	Add new Pokemua and set the owner to player
static Player	deserialize (Deserializer deserializer)	Deserialize information from deserializer and set to player
int	getLevel()	Get player's level
double	getLevelMultiplier()	Get level multiplier
ArrayList<Pokemua>	getPokemuas()	Get all Pokemuas
void	levelUp()	Add level by 1
void	serialize(Serializer serializer)	Serialize player information
void	setLevel(int level)	Set player's level

6. Package serial

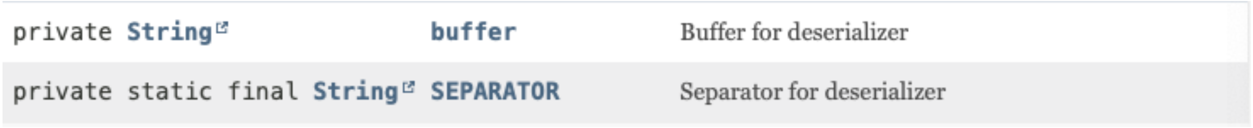
6.1 Class Deserializer

6.1.1 Constructor



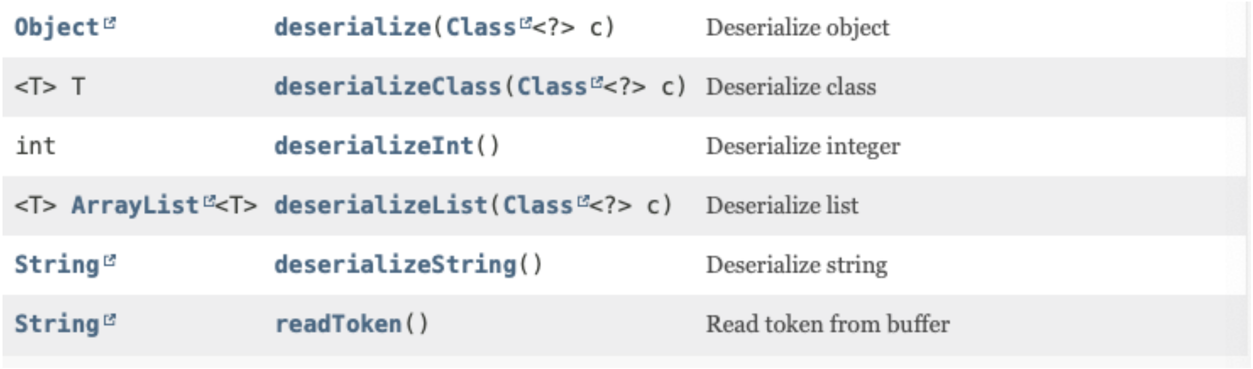
Deserializer(String buffer)	Initialize deserializer

6.1.2 Field



private String	buffer	Buffer for deserializer
private static final String	SEPARATOR	Separator for deserializer

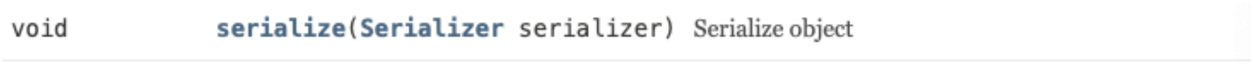
6.1.3 Method



Object	deserialize(Class<?> c)	Deserialize object
<T> T	deserializeClass(Class<?> c)	Deserialize class
int	deserializeInt()	Deserialize integer
<T> ArrayList<T>	deserializeList(Class<?> c)	Deserialize list
String	deserializeString()	Deserialize string
String	readToken()	Read token from buffer

6.2 Class Serialize

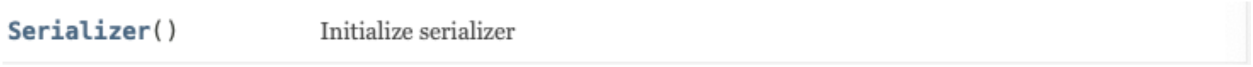
6.2.1 Method



void	serialize(Serializer serializer)	Serialize object

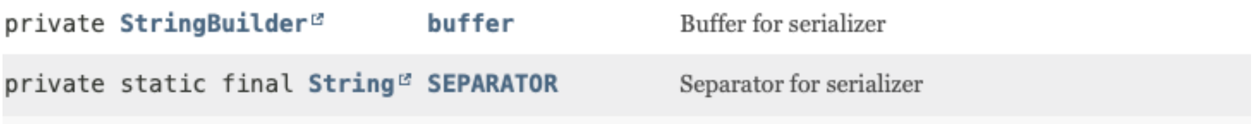
6.3 Class Serializer

6.3.1 Constructor



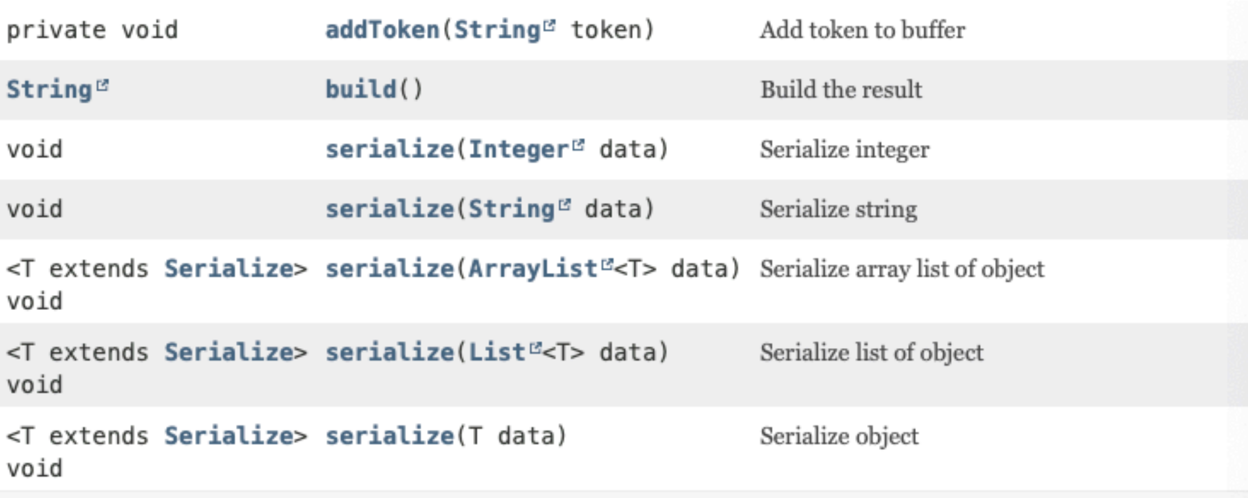
Serializer()	Initialize serializer

6.3.2 Field



private StringBuilder	buffer	Buffer for serializer
private static final String	SEPARATOR	Separator for serializer

6.3.3 Method



private void	addToken(String token)	Add token to buffer
String	build()	Build the result
void	serialize(Integer data)	Serialize integer
void	serialize(String data)	Serialize string
<T extends Serialize> void	serialize(ArrayList<T> data)	Serialize array list of object
<T extends Serialize> void	serialize(List<T> data)	Serialize list of object
<T extends Serialize> void	serialize(T data)	Serialize object