ICharacter *(interface)*

Method	Return
addStatus(IStatus)	void
dropStatus(IStatus)	void
applyStatuses()	void
waitTurn()	void
attack(ICharacter)	void
receiveDamage(int)	void
bePhysicallyAttacked(int)	void
beMagicallyAttacked(int)	void

Property	Type
name	String
alive	boolean
hp	int
maxHp	int
attack_enabled	boolean

AbstractCharacter *(class)*

Field	Type
scheduledExecutor	ScheduledExecutorService
turnsQueue	BlockingQueue<ICharacter>
defense	int
resistance	int
alive	boolean
equippedWeapon	GenericWeapon
statuses	ArrayList<IStatus>
statusIterator	Iterator<IStatus>
inIteration	boolean

Method	Return
AbstractCharacter(BlockingQueue<ICharacter>, String, int, int, int)	
addToQueue()	void
addStatus(IStatus)	void
dropStatus(IStatus)	void
applyStatuses()	void
waitTurn()	void
attack(ICharacter)	void
receiveDamage(int)	void
bePhysicallyAttacked(int)	void
beMagicallyAttacked(int)	void

Property	Type
name	String
alive	boolean
hp	int
maxHp	int
weight	double
attack_enabled	boolean

AbstractCharacterTest *(class)*

Field	Type
turns	BlockingQueue<ICharacter>
testCharacters	List<ICharacter>
testWeapon	GenericWeapon

Method	Return
preSetUp()	void
waitTurnTest()	void
addToQueue()	void

Enemy *(class)*

Field	Type
attack_damage	int

Method	Return
Enemy(BlockingQueue<ICharacter>, String, int, int, int, int, double)	
waitTurn()	void
attack(ICharacter)	void
equals(Object)	boolean
hashCode()	int

Property	Type
weight	double

EnemyTest *(class)*

Method	Return
setUp()	void
checkConstructor()	void
waitTurnTest()	void
damageTest()	void
attack()	void
testEquals()	void
testHashCode()	void

Package player