





**<<abstract>>
DungeonCharacter**

name: String
hitPoints: int
initiative: int
AC: int
attackBehavior: AttackBehavior

+ DungeonCharacter(name:String, hitPoints:int, initiative: int, AC:int, attackBehavior: AttackBehavior)
+ getAC():int
+ setAC(AC:int):void
+ getName():String
+ getAttackBehavior(): AttackBehavior
+ getHitPoints(): int
+ getInitiative(): int
+ setInitiative(initiative: int): void
+ addHitPoints(hitPoints: int):void
+ subtractHitPoints(hitPoints: int):void
+ isAlive(): boolean
+ attack(): void

Monster

+ Monster(name: String, hitPoints: int, initiative: int, ac: int, ab: AttackBehavior)
+ setNewMonsterAttack(attack: AttackBehavior): void
+ attack(enemy: DungeonCharacter): int

Extends

Skeleton

+ Skeleton()
+ toString: String

Bandit

+ Bandit()
+ toString: String

Orc

+ Orc()
+ toString: String

KoBold

+ KoBold()
+ toString: String

Extends

Goblin

+ Goblin()
+ toString: String

Gnoll

+ Gnoll()
+ toString: String

MonsterFactory

+ newMonster(n: int): Monster

Hero
- charRace: String - charClass: String - specialAttack: SpecialAttack
+ Hero(name: String, hitPoints: int, initiative: int, ac: int, ab: AttackBehavior) # setCharClass(charClass: String): void + getCharClass(): String # setCharRace(charRace: String): void + getCharRace(): String + getSpecialAttack(): SpecialAttack + specialAttack(enemy: DungeonCharacter): int + attack(enemy: DungeonCharacter): int

AssignSpecialAttack	
+ getSpecialAttack(checkClass: String): SpecialAttack	

HeroFactory	
+ newHero(charRace: String, charClass: String, name: String): Hero	

Aarakocra
+ Aarakocra(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

BugBear
+ BugBear(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + field: type

Elf
+ Elf(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

HalfElf
+ HalfElf(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

Dragonborn
+ Dragonborn(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

Dwarf
+ Dwarf(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

Gnome
+ Gnome(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior) + attack(opponent: DungeonCharacter): int

extends

Goliath

+ Goliath(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior)
+ attack(opponent: DungeonCharacter): int

Tabaxi

+ Tabaxi(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior)
+ attack(opponent: DungeonCharacter): int

Human

+ Human(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior)
+ attack(opponent: DungeonCharacter): int

Triton

+ Triton(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior)
+ attack(opponent: DungeonCharacter): int

Lizardfolk

+ Lizardfolk(name:String, hitPoints: int, initiative: int, ac: int, charClass: String, ab: AttackBehavior)
+ attack(opponent: DungeonCharacter): int