	# spribs: Spribs # posición: Veci	ElementoDe.Auego or				
	+ vistarEnemic	plenemics: Enemical: void				
	visitarPower visitarPlatafo visitarPlatafo visitar lugado	polenemigo: Enemigo); void (oppur PowerUp); void masjolat: Plantstorma); void nj; Jugastin); void signotor: Jugastin); void di Endisad Evaluago); void di Endisad Evaluago); void	Enemigo	Powertip	Plataforma # maos: Maos	Juago
	+ zerAfectadoj + afectarjentido	igador: Jugable) void of: EntidedDeJuego): void onleiem: ElementoDeJuego): void	# vidax: int # dafo: int	+ get_puntaje(j:int + setNelocidad(vet_int); void + move(de: Vecion;) void + afectarJugadov(ug: Jugable); void	+ exPompible(): boolean	#m/Partids: Partids #m/Jugador: Jugador
	+ aceptarColor	onjeent stemenocusago); void	# puntos/Utint # puntos/Death int	+ afectarJugador(jug: Jugable): void	BioqueSolide	Manking Ranking Mosrebro: MasterMind Mnundos: Mundo
			afectarje: EntidadMovil): void + asrAfectadoie: EntidadMovil): void +	Estrella	serAfectado(jugador: Jugable: void + delectarCollaion(entidad: Collaionable); void	
	4	Entidad	getPuntosKSI() int getPuntosDeath() int	# tempoDuracion = 10	+ exPompble(): boolean + + aceptarColision(elem: ElementoDeJuego):	+Juego() +nuevaPartida(): void
	# welcoldad: rel # describit: Vector # movel@ir: Vector; void # movel@ir: Vector; void # movel@ir: Vector; void # movel@ir: Vector; void		Spiny	Estrella() afectar Jagador(jug: Jagable); void + aceptar Collision(elem: Elemento DeJuego);	Tuberia .	
			# velocidad: int # posicion: Vector # dirección: Vector	Champinion/Verde	# plants: PiranhaPlant	Nivel
geFosicion(): Vector gefvelocidad(): int detectarColizion(entid)		ad f: Collsionable, entidad2: Collsionable): boolean	# puntosKill : int = 60 # puntosDeath : int = -30	+ ChampifonVerde()	+ Tuberis() + setAfectado(jugador: Jugable); void + + detectarCollsion(entidad: Collsion); void + + esRompible(); boolsan +	-plateformas: Coleccion-Plateformas- -powerUps: Coleccion-PowerUps- -amerigos: Coleccion-Enemigos- firmaps: Mapa
			- Spiny()	Champiton/verde() stector/Jugador(jug: Jugable); void + sceptor/Collision(elem: Elements/DeJuego); void +	+ esRompible(): boolean + + aceptarCollaion(elent ElementoDeJuego): sold +	
			afectarje: EntidadMovil): void ++ serAfectadoje: EntidadMovil): void ++	SuperChampinion	Vacio	Niveljer: Mapa) addFlataformajp: Plataformaj; vold addFloweriopic: Enamigosi; vold addFloweriopip: Poweriopi; vold
			PiranhaPlant	+ SuperChampifon() + afectarJugador(jug: Jugable): void +	+ Vacio() + afectar(upador: Jupable); void	 addEnemigo(e: Enemigos): void addPowerUp(p: PowerUp): void removePlataforma(p: Plataforma): void
			# velocidad: int # posicion: Vector	+ aceptarCollsion(elem: ElementoDeJuego): vold +	+ senAfectado(jugador: Jugable); boolean + + detectarColsion(entidad: Colsion); void + + esRompible(); boolean +	add/braset/pip: Power/pip: void remove/Plastformap: Plastformap: void remove/Plastformap: Plastformap: void remove/Plastformap: Plastformap: void remove/Plastformap: Plosedtop! void chat/Victoria(): boolean get/Plastforman; Dicciocion-Plastforman get/Plastformap; Coleccion-Plastforman get/Plastformap; Coleccion-Plastforman get/Plastformap; Coleccion-Plastformap get/Plastformap; Coleccion-Plastformap get/Plastformap; Coleccion-Plastformap get/Plastformap; Coleccion-Plastformap get/Plastformap; Coleccion-Plastformap get/Plastformap get/P
Ju Fylder int	gable		# direction: Vector # millubaris: Tuberia # dentroTuberia: boolean # puntostiti = 20	FlorDeFuego	LadrilloGolido	gelFlataformas(): Coleccion=Plataforma= getEnemigos(): Coleccion=Enemigo=
			# puntosDeath = -30 # puntosDeath = -30	# velocidad: 0 # direccide: 0	LadrilloSolido # monedas: Moneda	* Ben-omerchit: Compositive-omerchi-
detererse(): void caen(): void		No.Jugable	PiranhaPlanti) afectar(e: EntidadMovil): void ++ serAfectado(e: EntidadMovil): void ++	FlorDeFuego() stectorJugador(jug: Jugable): void + sceptarCollation(elem: ElementoDeJuego):		
scavan-coar(; void recogeritem(powers ganar/vida(); void perder/vida(); void	up: PowerUp): void	+ detectarJugador(jugador: Jugable): void	+ aceptarCollaton(elem: ElementoDeJuego): void +	void +	LadriloSolido() setAfectado(): Jugaphie; boolean ++ petMonada(); Monada setPompble() boolean ++ aceptarColaton(elem: SiementoCeJuego(); loid ++	BloqueCePregunts
getVidas(): void getVidas(): int actualizerHUD(): vo	id	Para hacer saltar, caer o detener la entidad se usará el método move, pasandole el debido vector por parámetro	Lakitu	Moredas	+ aceptarCollaion(elem: ElementoDeJuego): sold +	# powerUp: PowerUp
		Protagonista	# velocidad: int # poeksion: Vector	Erantidad Coleccidorists	Meta	BloqueDePregunta() serAlschado(jugador: Jugabie): void ++ serPowertip(): Powertip sceptraCollision(elem: ElementoDeJuego):
		Reubecriptores: Subscriptor	# direction: Vector # spinys: Colection=Spinys # puntoskill = 60	# velocidad: 0 # dirección: 0		+ aceptarCollsion(elem: ElementoDeJuego): void ++
		+apilcavEstado(): 7	KoopaTroopa() afectanie: EntidadMovill: void ++	sfectarJugador(jug: Jugable); void + sceptarColision(elem: ElementoDeJuego); void +	afectard; vold sex-Michadojugador; Jugable); vold + detectar/Collacojugador; Jugable); vold + es/Brompble(); boolsan + aceptar/Collacojelen; Elemento/CeJuego);	
		MaricOefault #vidax: int #recoedax: int	* KoopaTrocpa() * afectar(s: EntidadMovil): wold ++ * sarAfectad(s: EntidadMovil): wold++ * acaptarCollabor(sient: ElementoDaJuego): vold +	100 +	+ aceptarCollaion(elem: ElementoDeJuego): sold +	Partida
		Amonedax: Int Aestado: Decorator			Bandera PrincesaPeach	Rigrafica: Sprite Riganitalia: HLD Receide: Sonido Receide: Sonido Receide: Jago:
		-Mario()	BuzzyGeetie # velocidad: int # resistant Varior		+ afectar(): void + afectar(): void	PROSSIDEJSego:
		*Mario() **matrivitati(; void	# posicion: Vector # directión: Vector # puntosXII : Int = 30 # puntosDeath : Int = -15		afectan() determina la finalización del nivel o del jue	go Partida () establecerModo(m: FactoryModo)
		+recogerMoneda)m: Moneda); void +afectar(e: Entidad); void + amounteientrillet dir/; void a	# puntosDeath : int = -15 + BuzzyGeede		Sonido	
		+recibirPowerUp(p: PowerUp): void + aceptarCollision(elem: ElementoDeJuego): void +	+ afectarje: EntidadMovil): void ++ serAfectado(e: EntidadMovil): void ++		#entidad: EntidadLogica • method/boe't hose	Ranking Blacker addresarchine
		BaseDarrorstor	+ aceptarCollaton(elem: ElementoDeJuego): void +		aScreetor)	Marchino: bd
		BaseDecorator #componenteDecorado: Componente	Goomba		+sonidoSalto() +sonidoGolpe() +musica()	+Ranking() +leenArchivo(): void
			# velocidad: int # posicion: Vector # dirección: Vector		+sonidoCollaion() +sonidoRecolectarPowerUp(p:PowerUp) +sonidoGameOver()	Controlativo/DaArchivo
		DecoratorSuper	# puntosi00 : int = 60 # puntosDeath : int = -30		+sonidoMuerte()	Marchino: bd
		=EstadoSuperMario() +aplicarEstado();	+ Goomba() + afectar(e: EntidadMovil): void ++ + senAlectado(e: EntidadMovil): void ++		Sprite	+ControladorArchivo(x:td) +modificarArchivo(x:td)
		*apicariumato(; evelocidad(v: int) :void *alectar(s:Enidad): void *disparar(): FineBall	+ ascarje: EntidadMovil); void ++ + senAfectado(e: EntidadMovil); void ++ + aceptarCollaion(elem: ElementoDeJuego); void +		AmEntidad: Entidad	+leanArchivo(): boolean
		DecoratorFireball	1 ,		*spriteSalto(): "Una imagen/ GIF" *spriteDerechs(): "Imagen/ GIF" *spriteIcquiends():" Imagen/GIF"	Mastermind #entidad: No.Jugable
		+EstadoMarioFlorDeFuego() +aplicarEstado()	Bowser # freBalls: Coleccion*Fireball*			#fabrica: FabricaEnemigos #gestorCollsiones: Collsion
		+aplicarEstado() +velocidad(v: lrt) void +afectar(e:Entidad): void +disparae(): FinsBall	a Snamer()			+MasterMindje: NoJugable; f: FabricaEnemigos; g:
		Decoratorinyulnerable	+ Bowser() + afectar(e: EntidatMovil): void ++ + senMectado(e: EntidatMovil): void ++ + aceptarCollaton(elem:			GestorCollsiones) +moverEntidad(ent Entidad,dir: Vector, velocidad int): Broilean
		aEstarbMariolmann/bla/)	ElementoDeJuego): void +			colsiorEntidad(ent Entidad plat Plateforma): boolean crearEnemigos(): NoJugable
		+aplicarEstado() +velocidad(v: int.) : void +afectarle Entidad(v-pid	**Interface** State			
		+afectar(e Entidad) void +disparan(): FineBall	afectarje: EntidadMovil; void ++ serAfectadoje: EntidadMovil; void ++			
		Disparar no es un comportamiento comó entre los 3 decoradores. ¿Que se puede hacer? RTA: Se dejan vacios.	getPuntoskill() int			
		RTA: Se dejan vacios.	KoopaTroopaDefault	ContestoKoopaTroopa		Controlador
			KoopaTroopa() afectar(e: EntidadMovil): void ++	- state:State # velocidad: Int # posición: Vector		Ppartida: Partida Pgrafico: Grafica
			* Koopat roopa() * afectar(e: Entiatelitorii); void ++ * are Mectar(e: Entiatelitorii); void ++ * getFurnoakii(); int * acquerCollatorijeter: EliementoCeLuago(: void +)	# posición: Vactor # dirección: Vector # purtica(NI = 90 # puntica(Death = -45		+ method(type): type
			ElementoDeJuego): void + KoopaCaparazonEstatico	# puntosDeath = -45		Jugador Browleiselou Controlador
				Context(KoopaDefault) cambiarEstado(State): void serAlectado(s: EntidadMoil): void serAlectado(s: EntidadMoil): void safetat; EntidadMoil): void		+Ausadoric Controlador)
			+ KoopaTroopa() + afectar(e: EntidadMovil): void ++ + seeAlectado(e: EntidadMovil): void ++ + exePhytholKill: int	afactarje: EntidadMonilj: void anAfactadoje: EntidadMonilj: void gePuntonKilj:int gePuntonDeath():int		+mover(dr: Vector): boolean
			+ sceptarCollaton(elem: ElementoDe.luego): void +	gePuntosDeath() int aceptarCollaion(elem: ElementoDe.luego): void +		
			KoopaCaparazonMovil			«interface» Subscriptor
			+ KoopaTroopa() + afectar(e: EntidadMovii): void ++			+actualizar()
			+ serAfectado(e: EntidadMov8): void ++			
			+ geo-umosus() riz + aceptarCollation(elem: ElementoDeJuego): void +			Subscriptor que observa el estado de mario(en considereacion), y el estado del koopa-troopa

