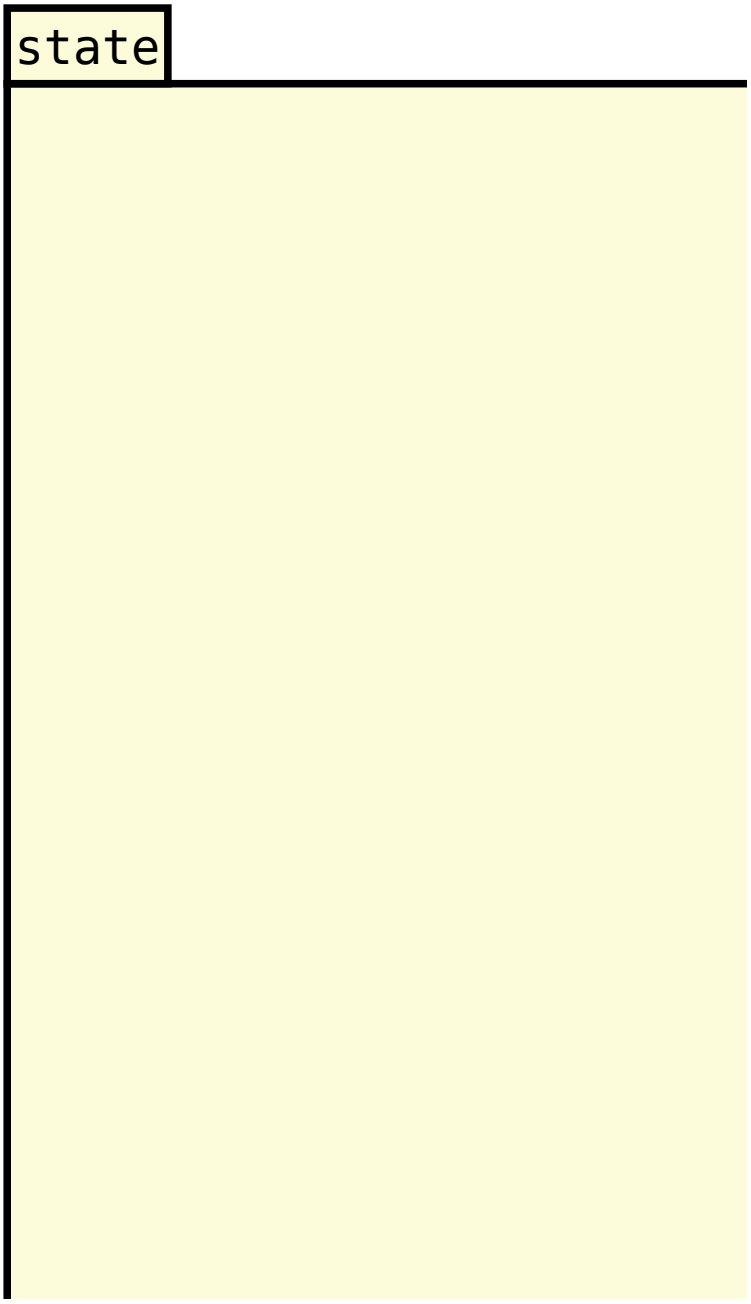
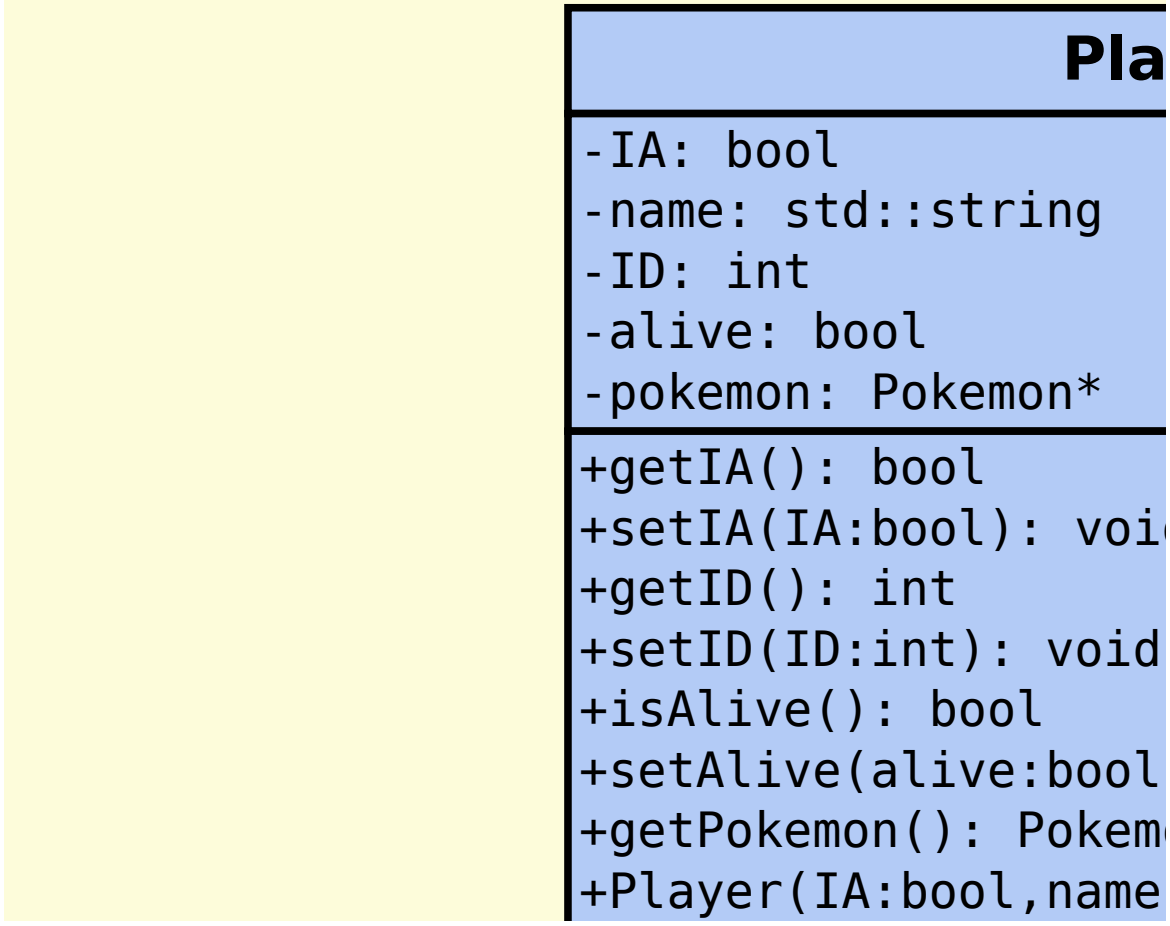
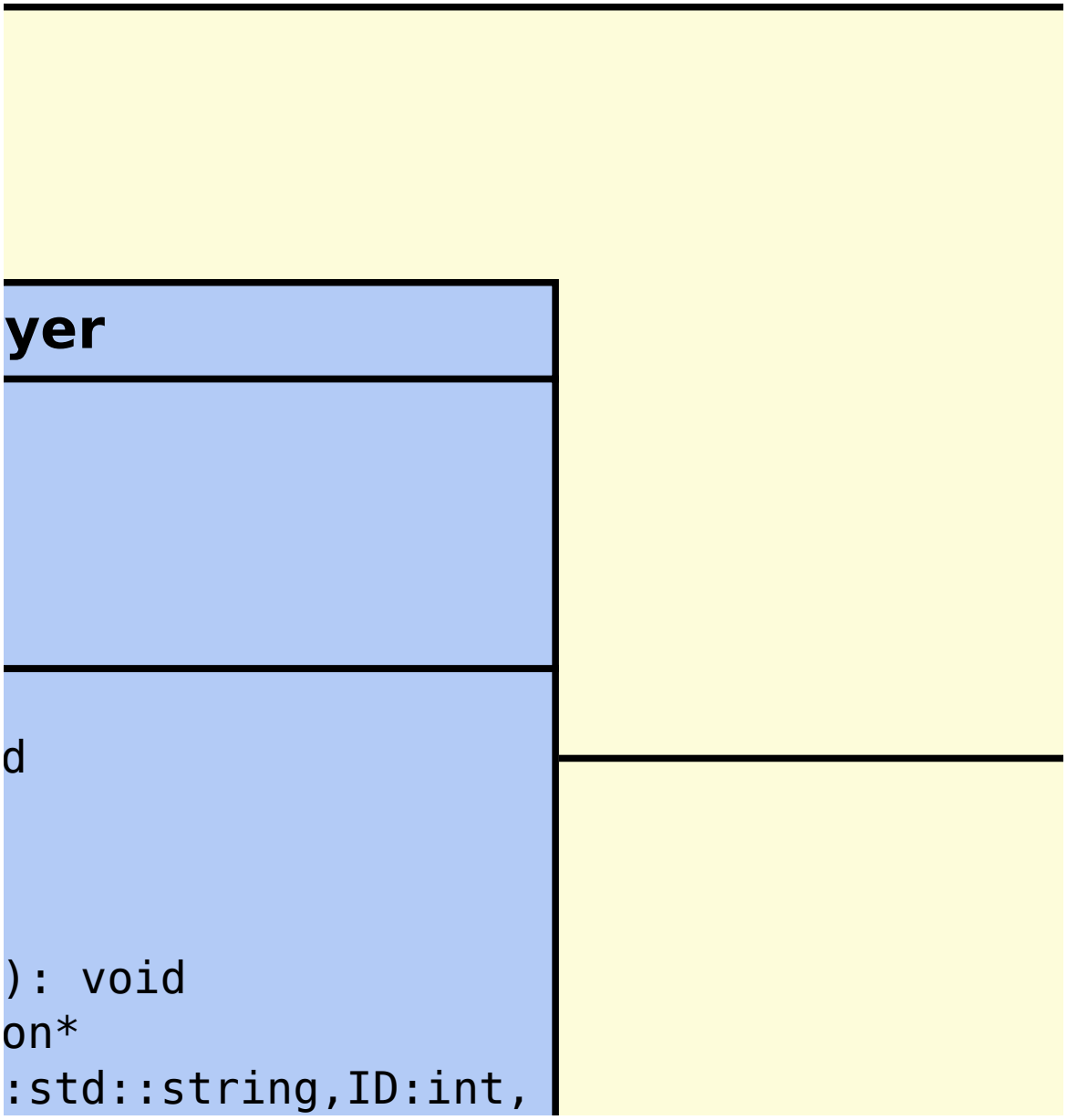
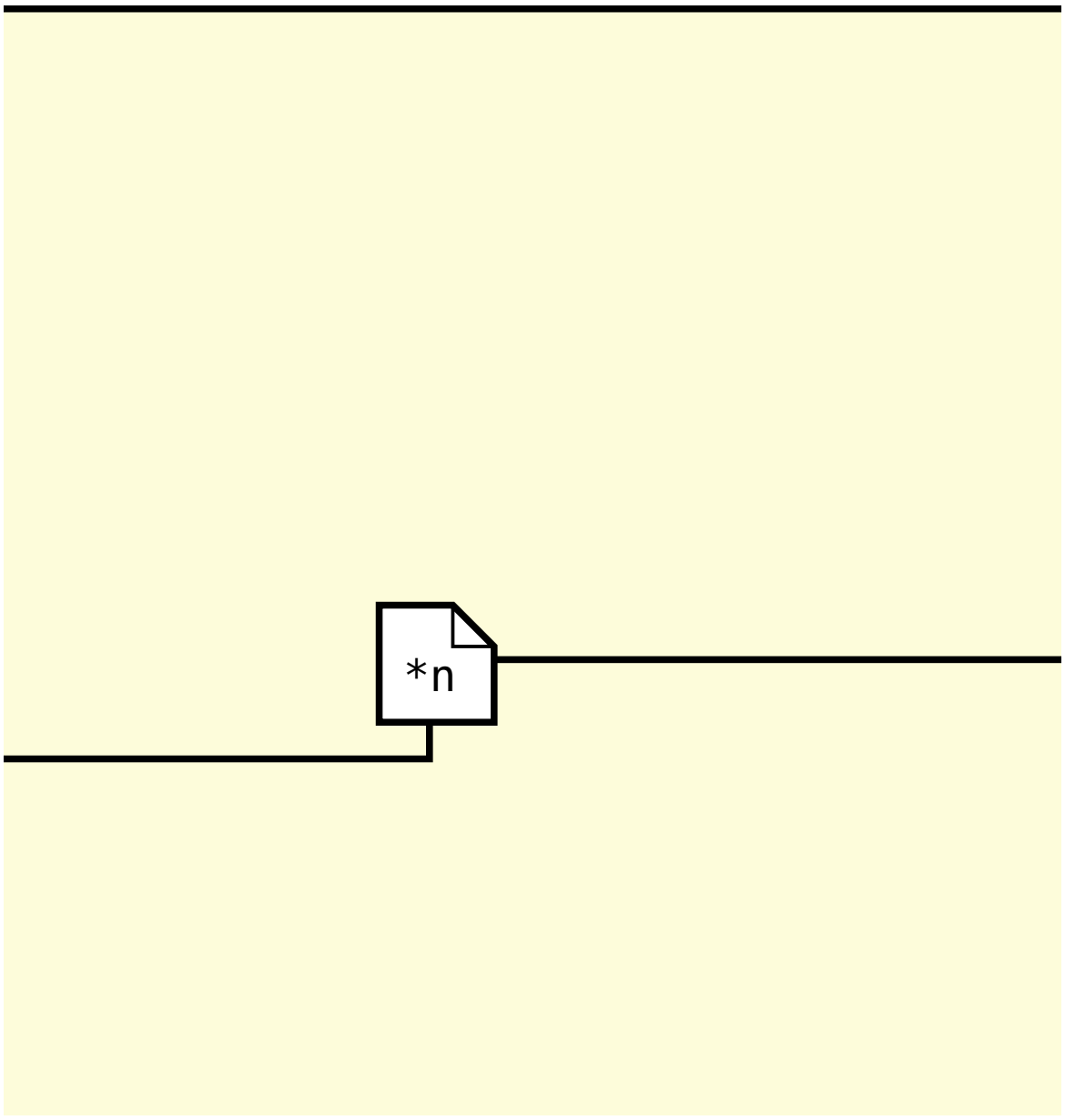
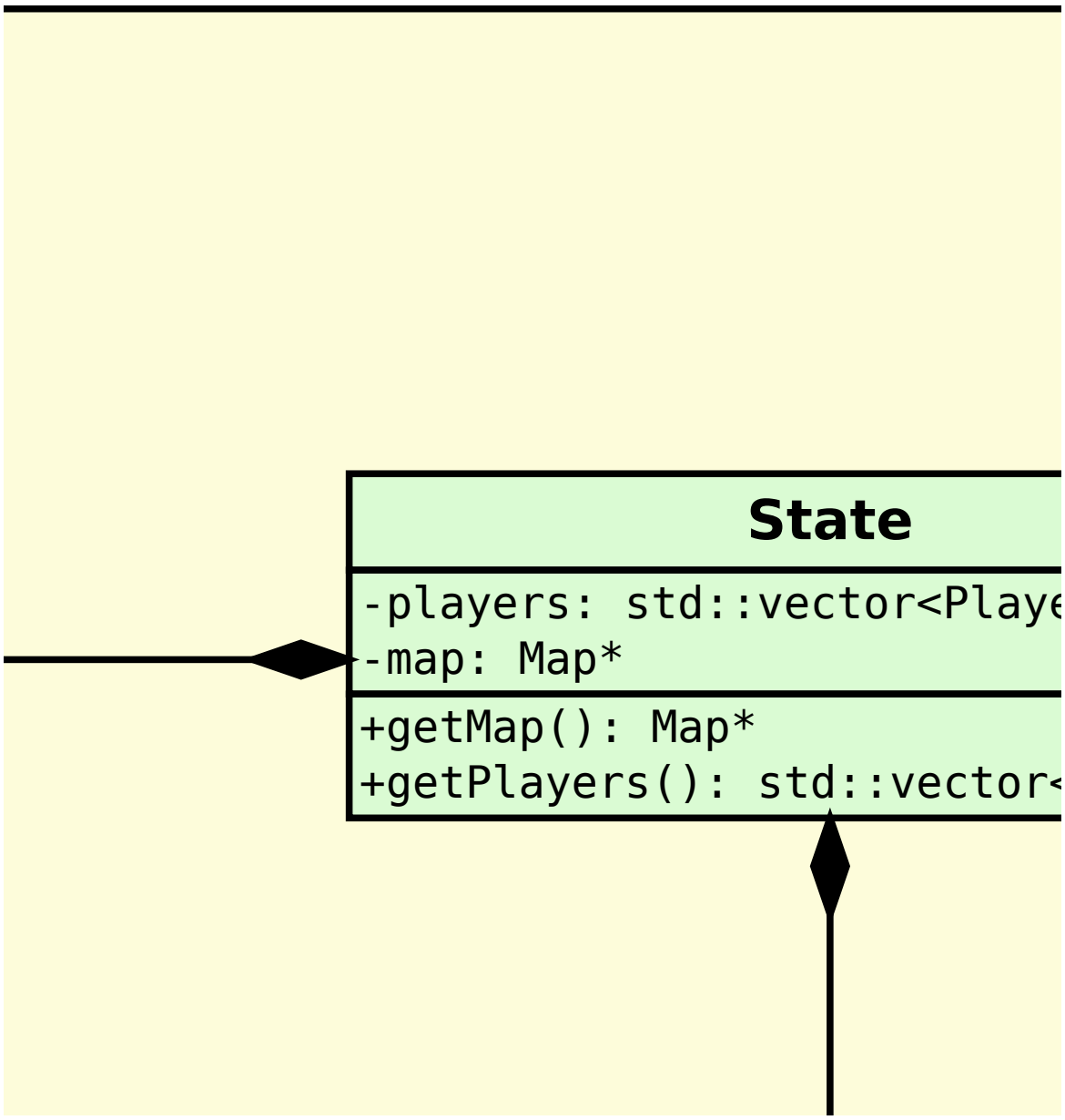
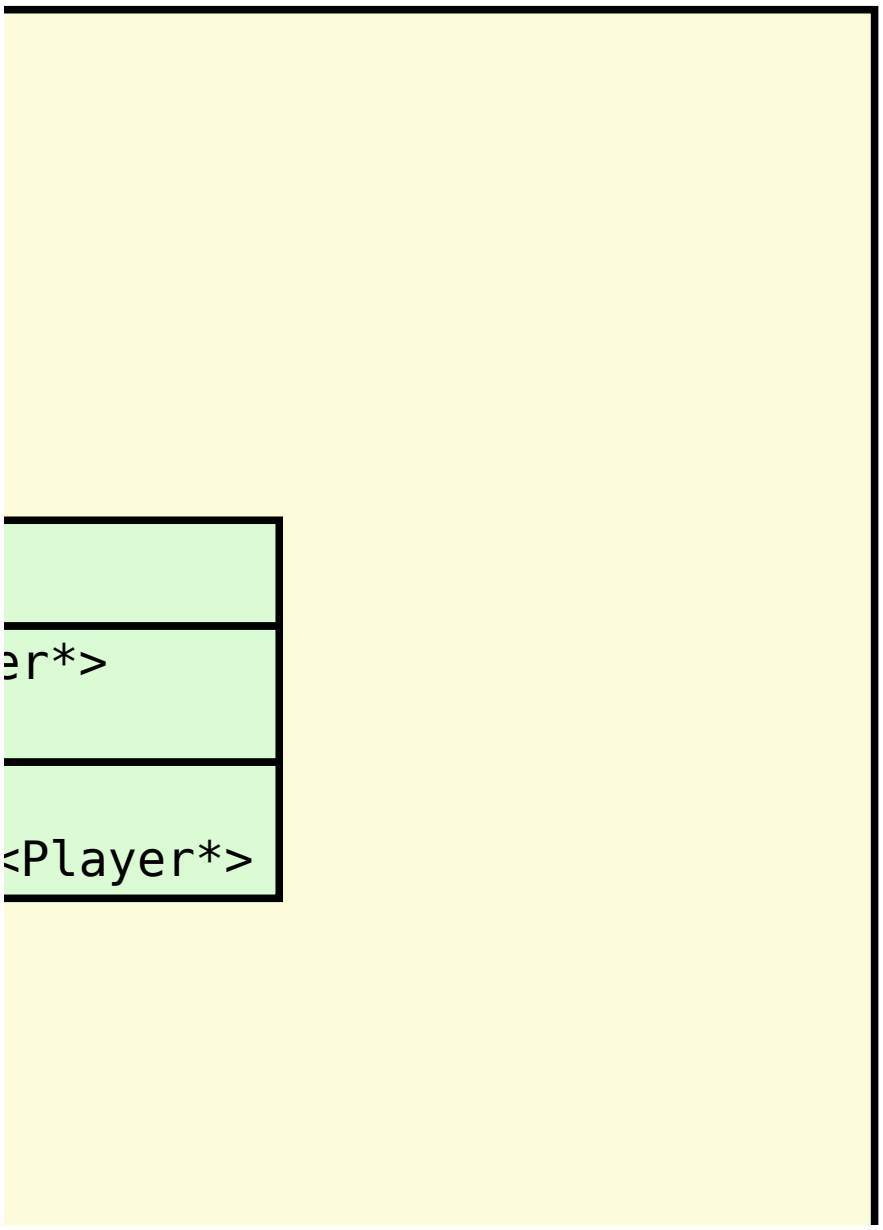

state

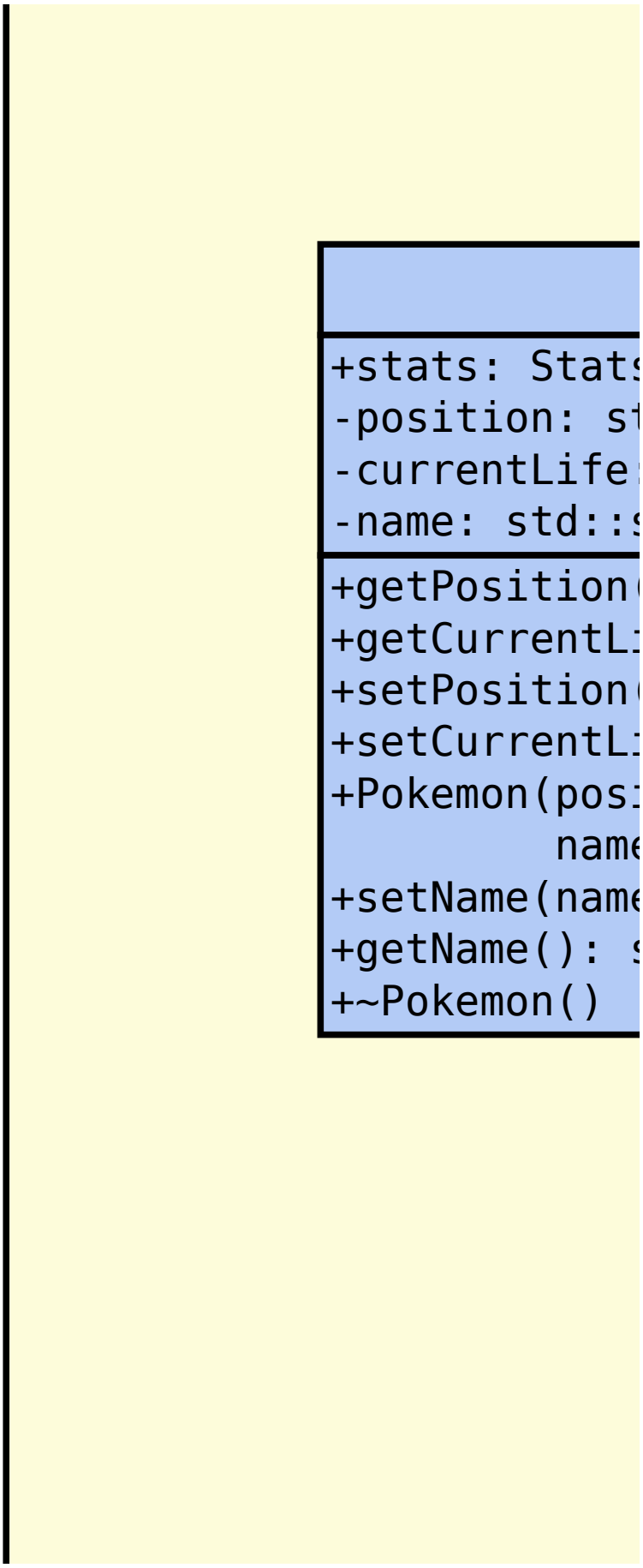

Pla
-IA: bool
-name: std::string
-ID: int
-alive: bool
-pokemon: Pokemon*
+getIA(): bool
+setIA(IA:bool): voi
+getID(): int
+setID(ID:int): void
+isAlive(): bool
+setAlive(alive:bool
+getPokemon(): Pokem
+Player(IA:bool,name

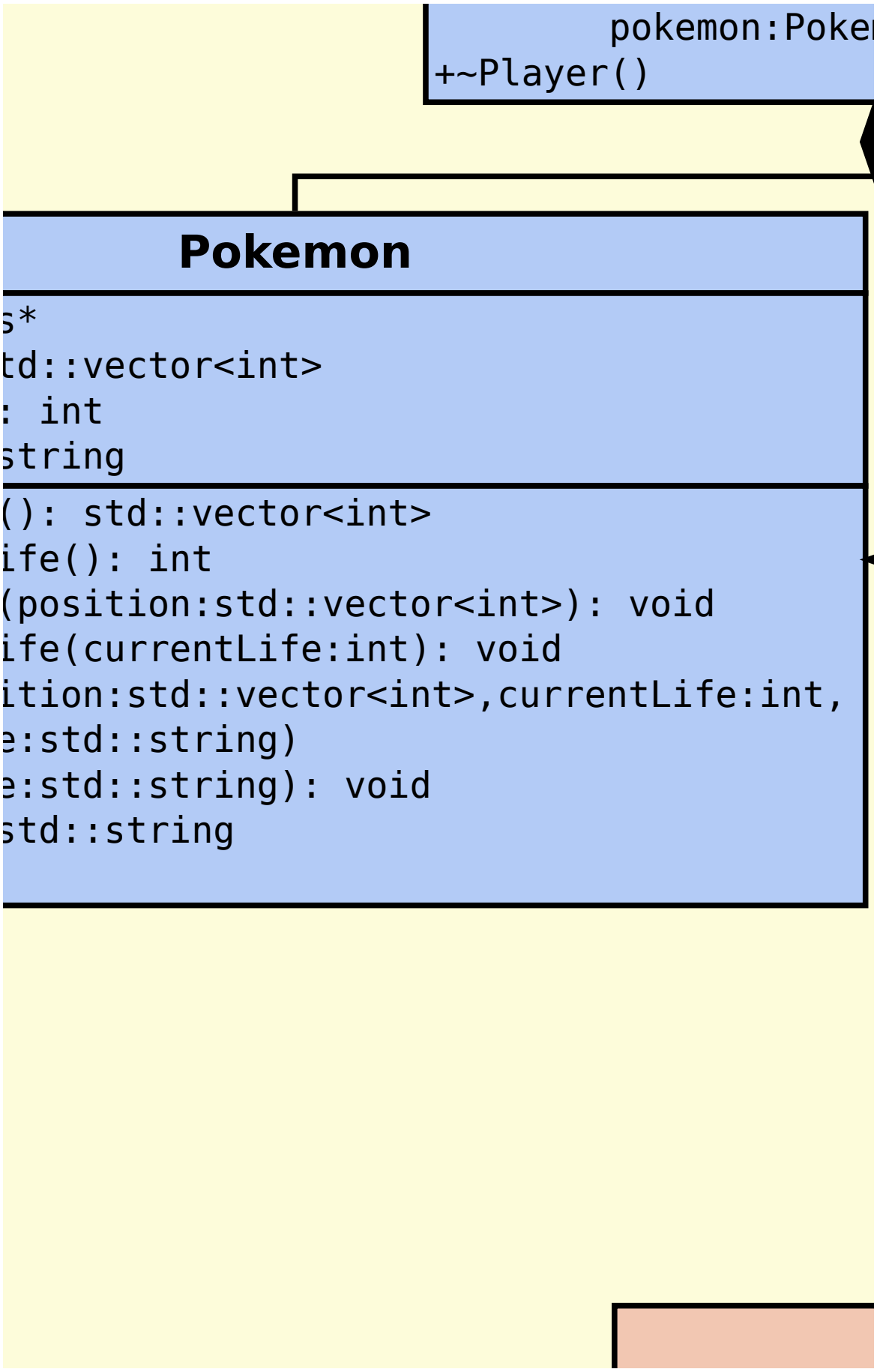

yer
d
): void
on*
:std::string,ID:int,

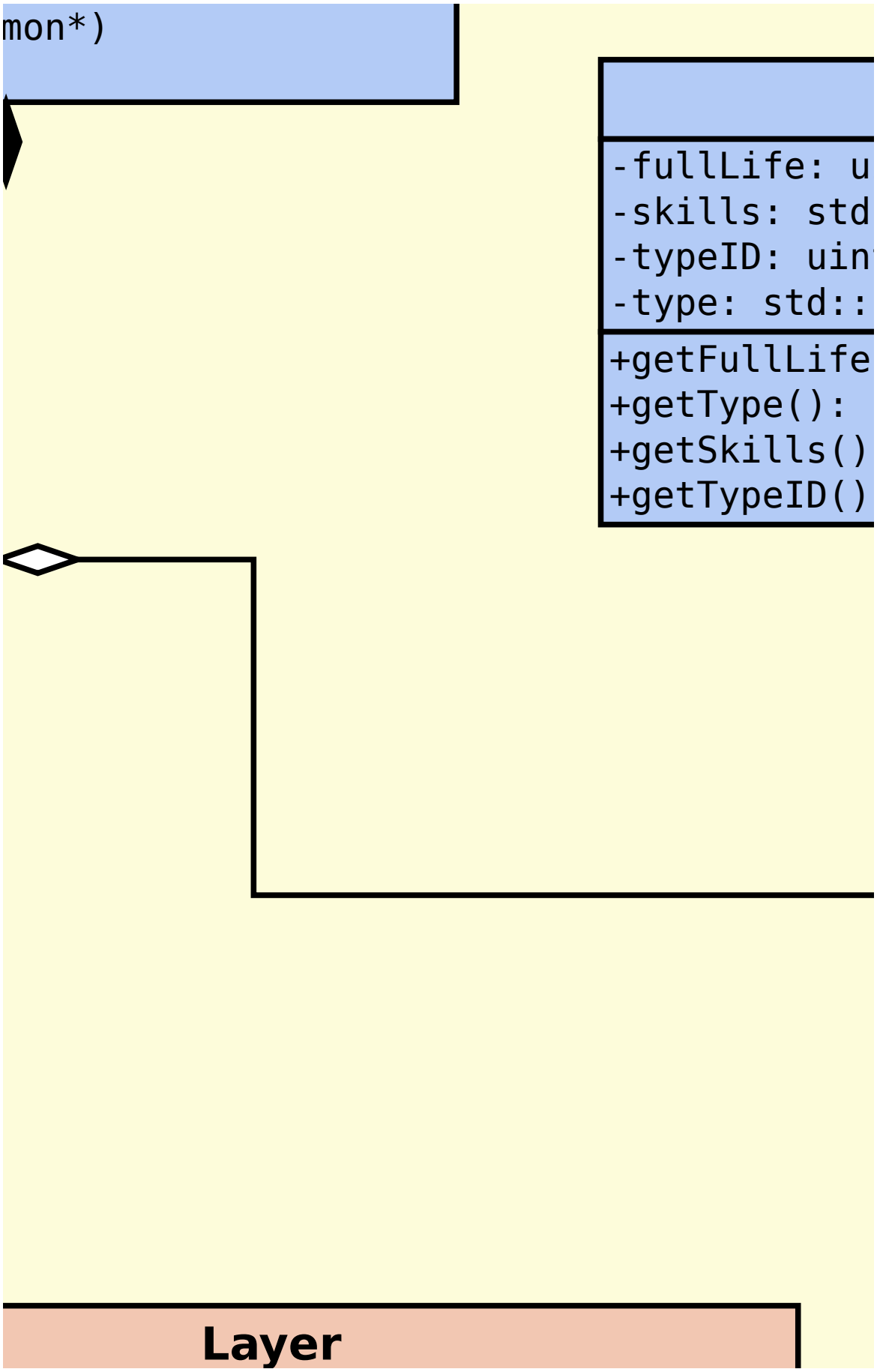

*n

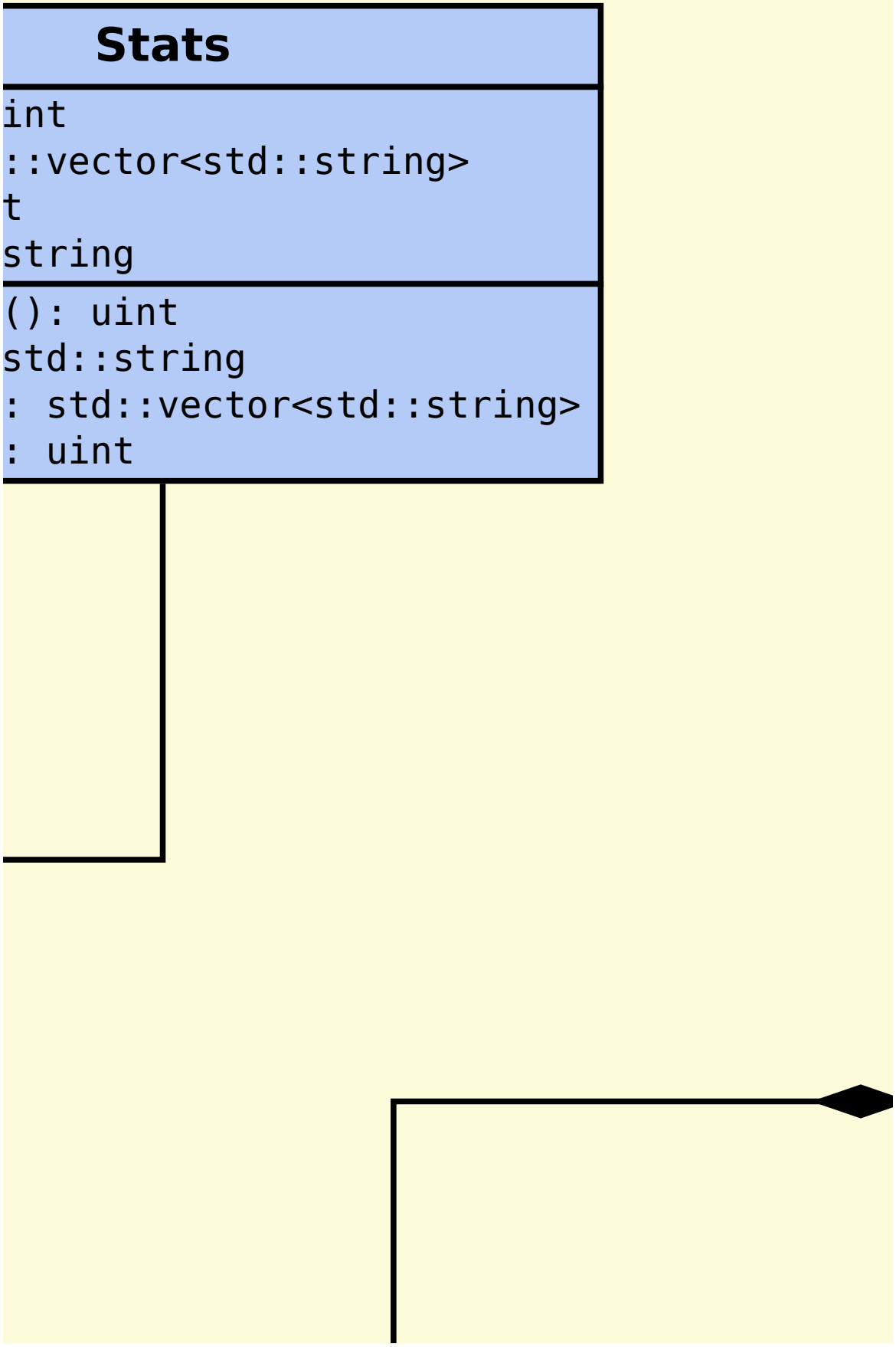

State
-players: std::vector<Playe
-map: Map*
+getMap(): Map*
+getPlayers(): std::vector<

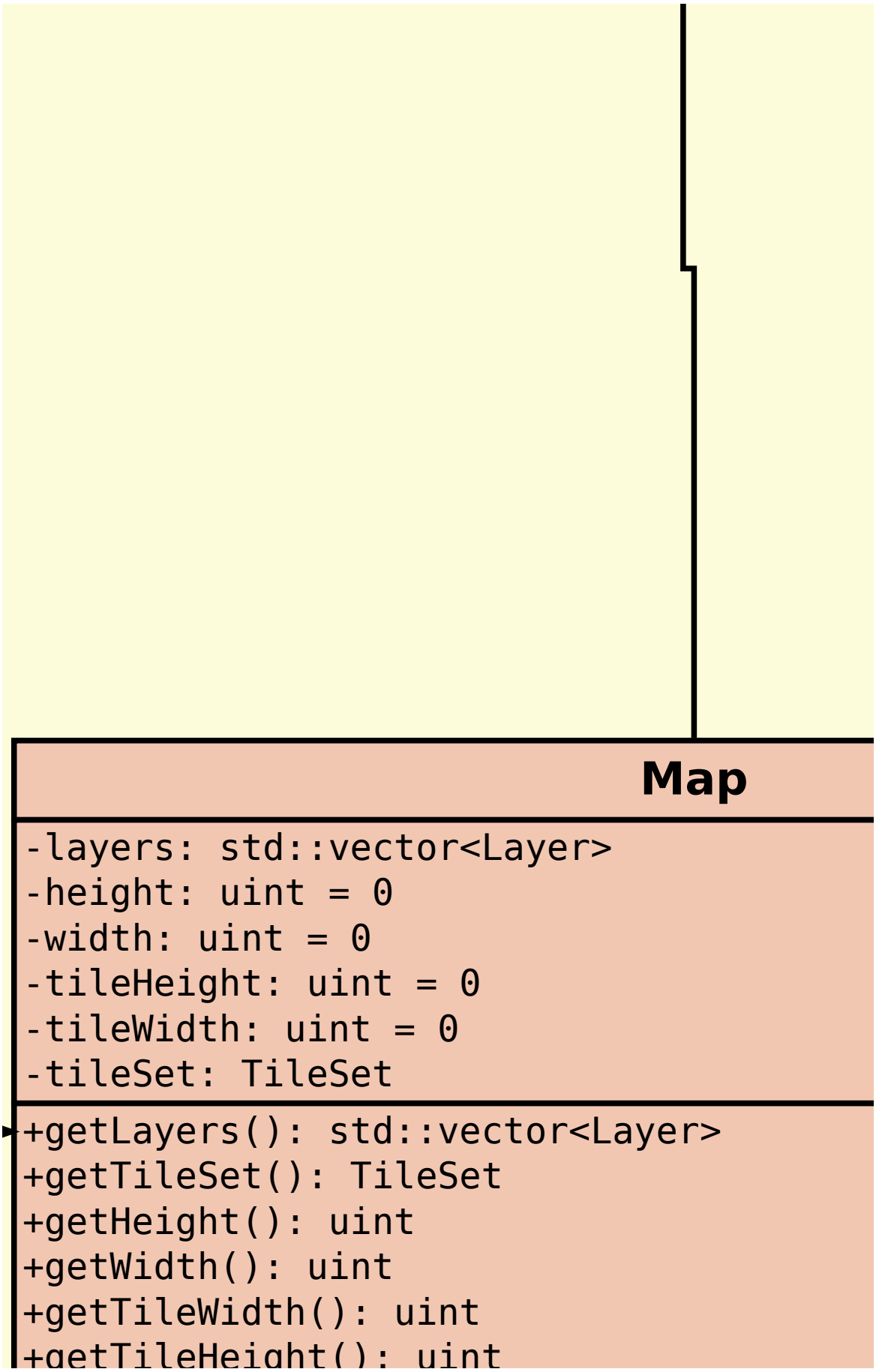

er*>
<Player*>

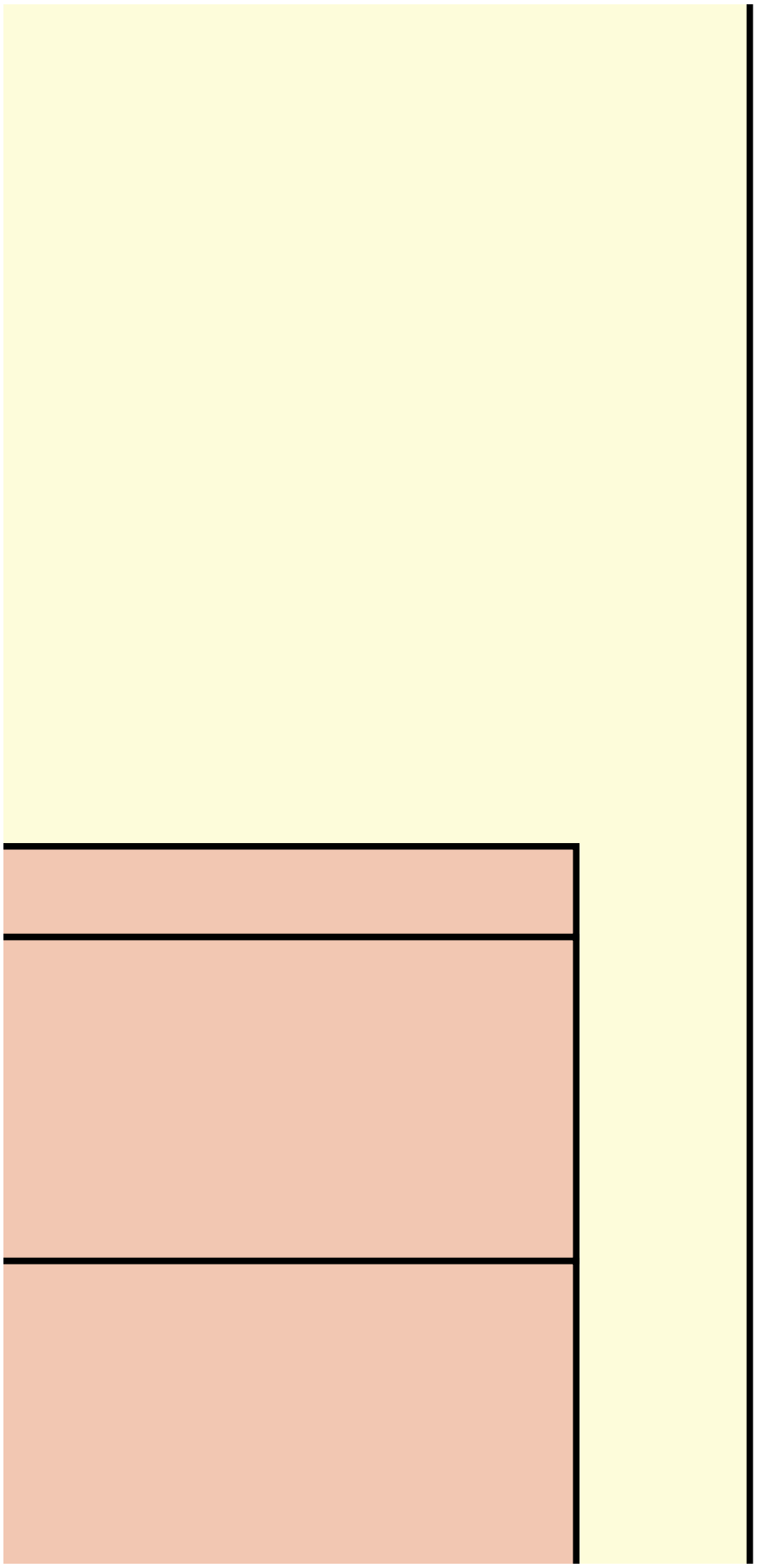

+stats: Stats
-position: st
-currentLife
-name: std::s
+getPosition(
+getCurrentLi
+setPosition(
+setCurrentLi
+Pokemon(posi
name
+setName(name
+getName(): s
+~Pokemon()

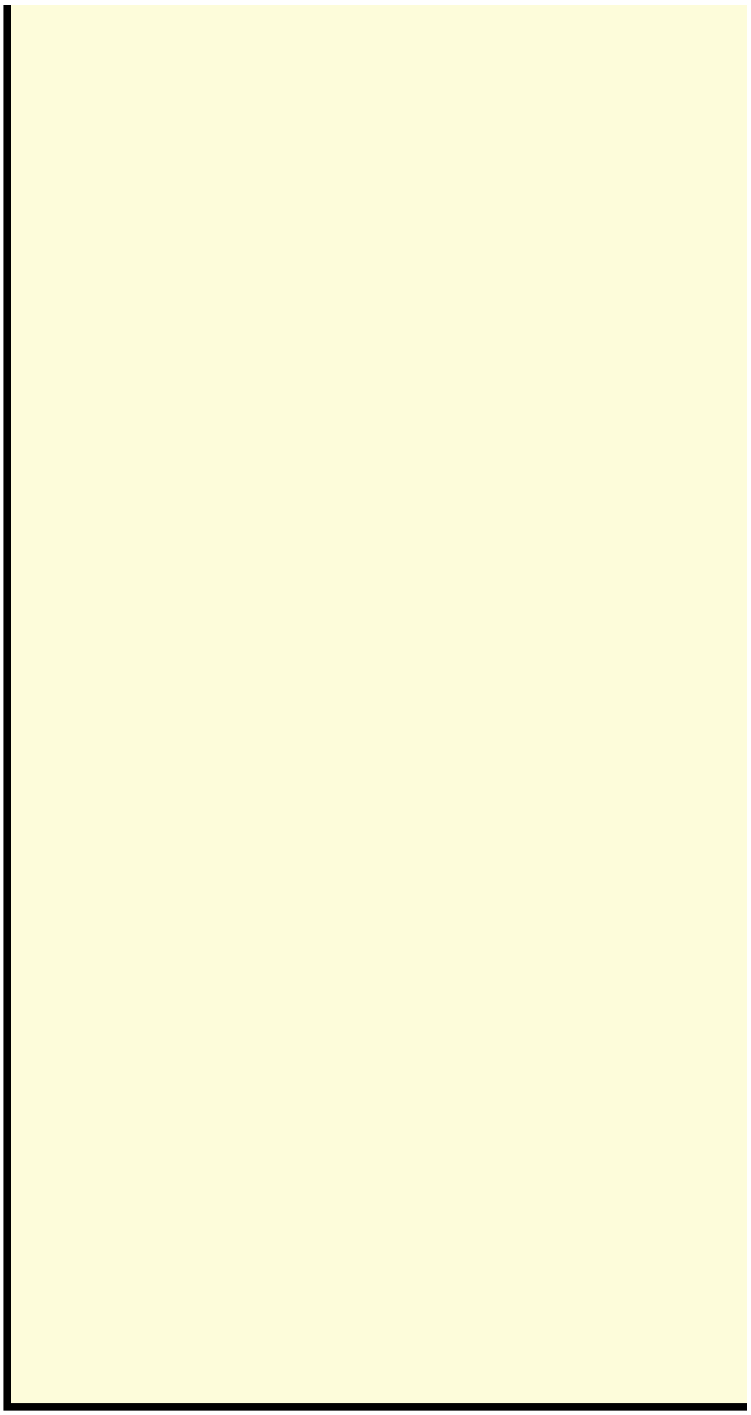

pokemon:Poke
+~Player()
Pokemon
s*
td::vector<int>
: int
string
(): std::vector<int>
ife(): int
(position:std::vector<int>): void
ife(currentLife:int): void
ition:std::vector<int>,currentLife:int,
e:std::string)
e:std::string): void
std::string

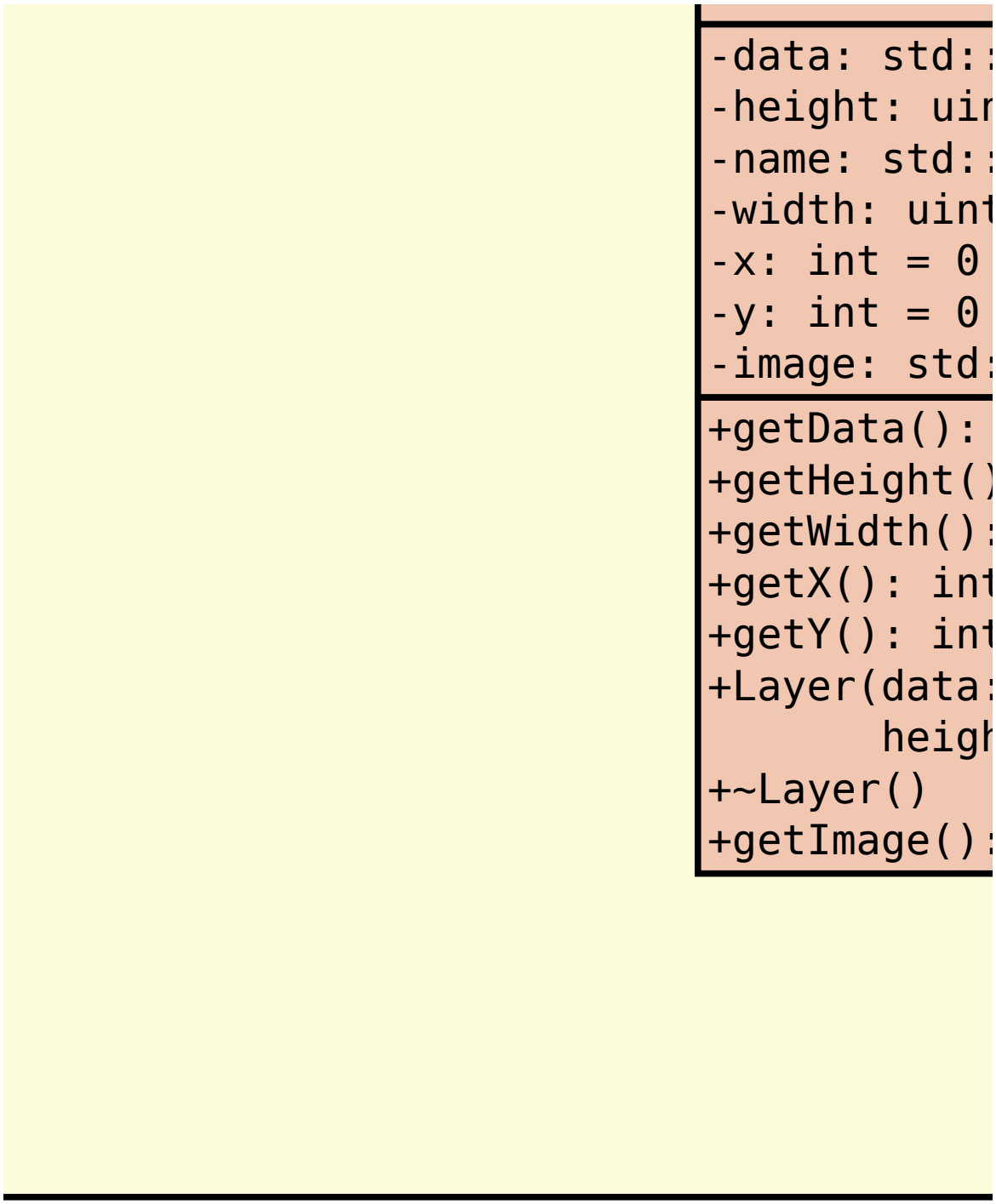

mon*)
-fullLife: u
-skills: std
-typeID: uin
-type: std::
+getFullLife
+getType():
+getSkills()
+getTypeID()
Layer

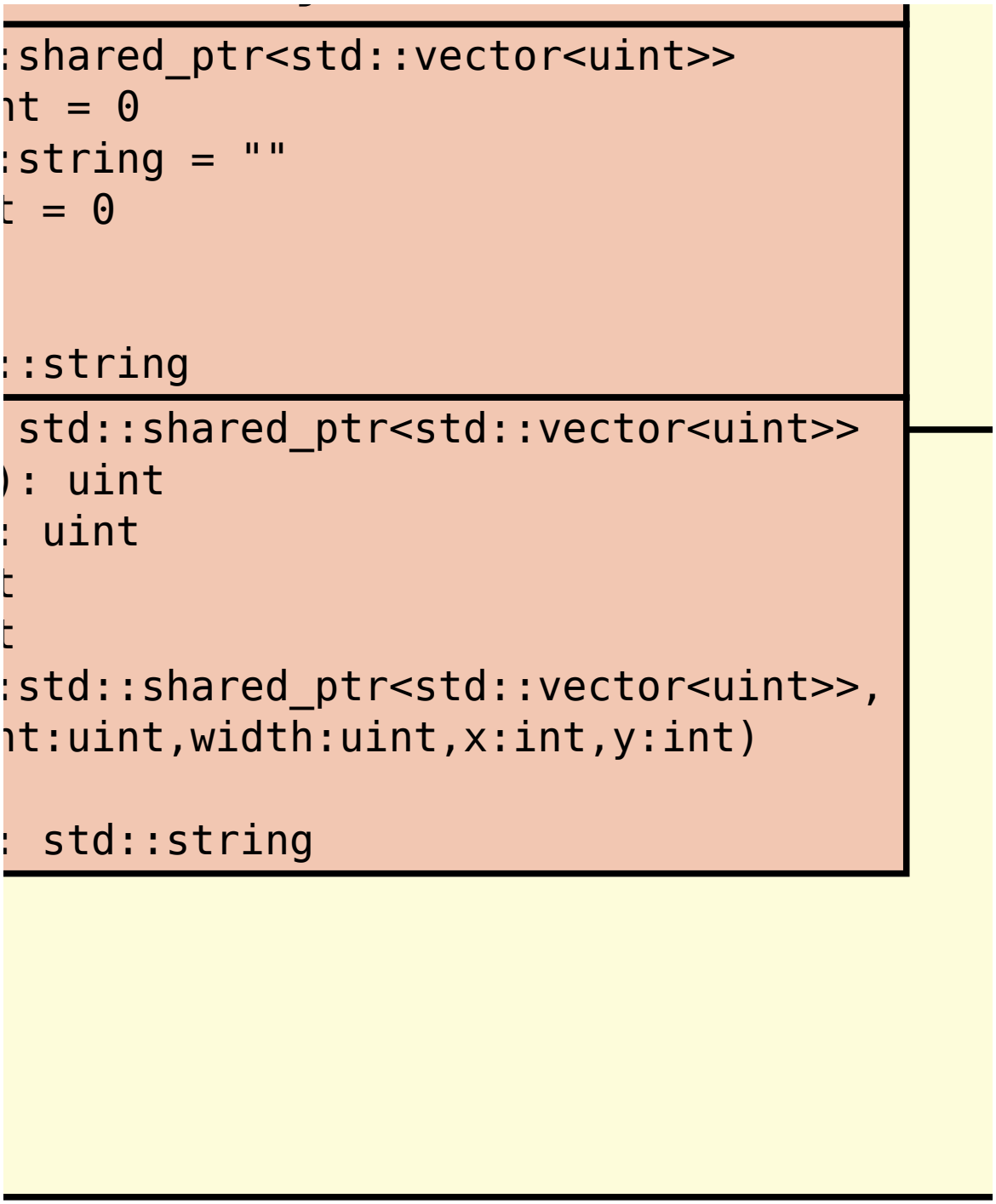

Stats
int
::vector<std::string>
t
string
(): uint
std::string
: std::vector<std::string>
: uint

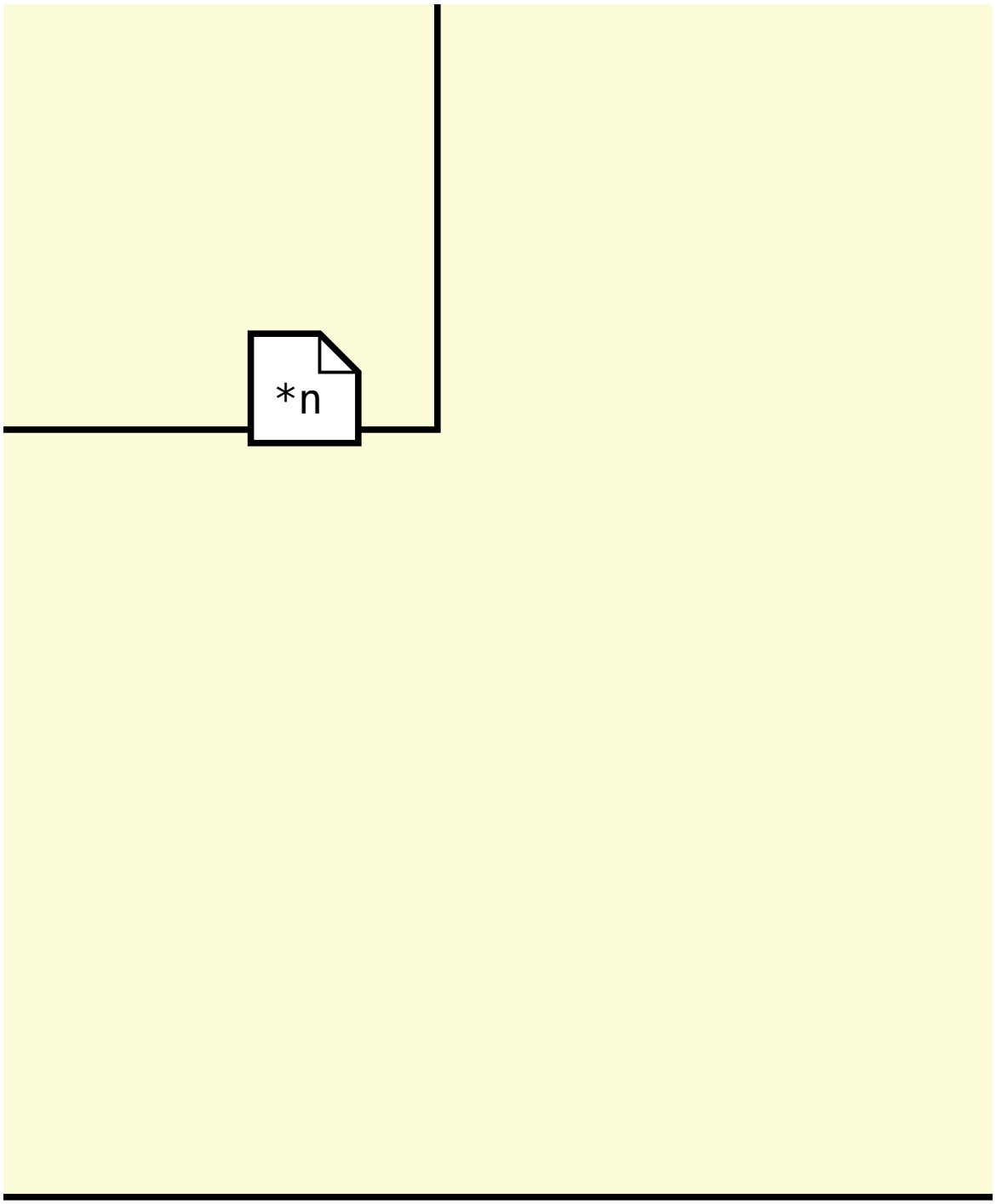

Map
-layers: std::vector<Layer>
-height: uint = 0
-width: uint = 0
-tileHeight: uint = 0
-tileWidth: uint = 0
-tileSet: TileSet
+getLayers(): std::vector<Layer>
+getTileSet(): TileSet
+getHeight(): uint
+getWidth(): uint
+getTileWidth(): uint
+getTileHeight(): uint

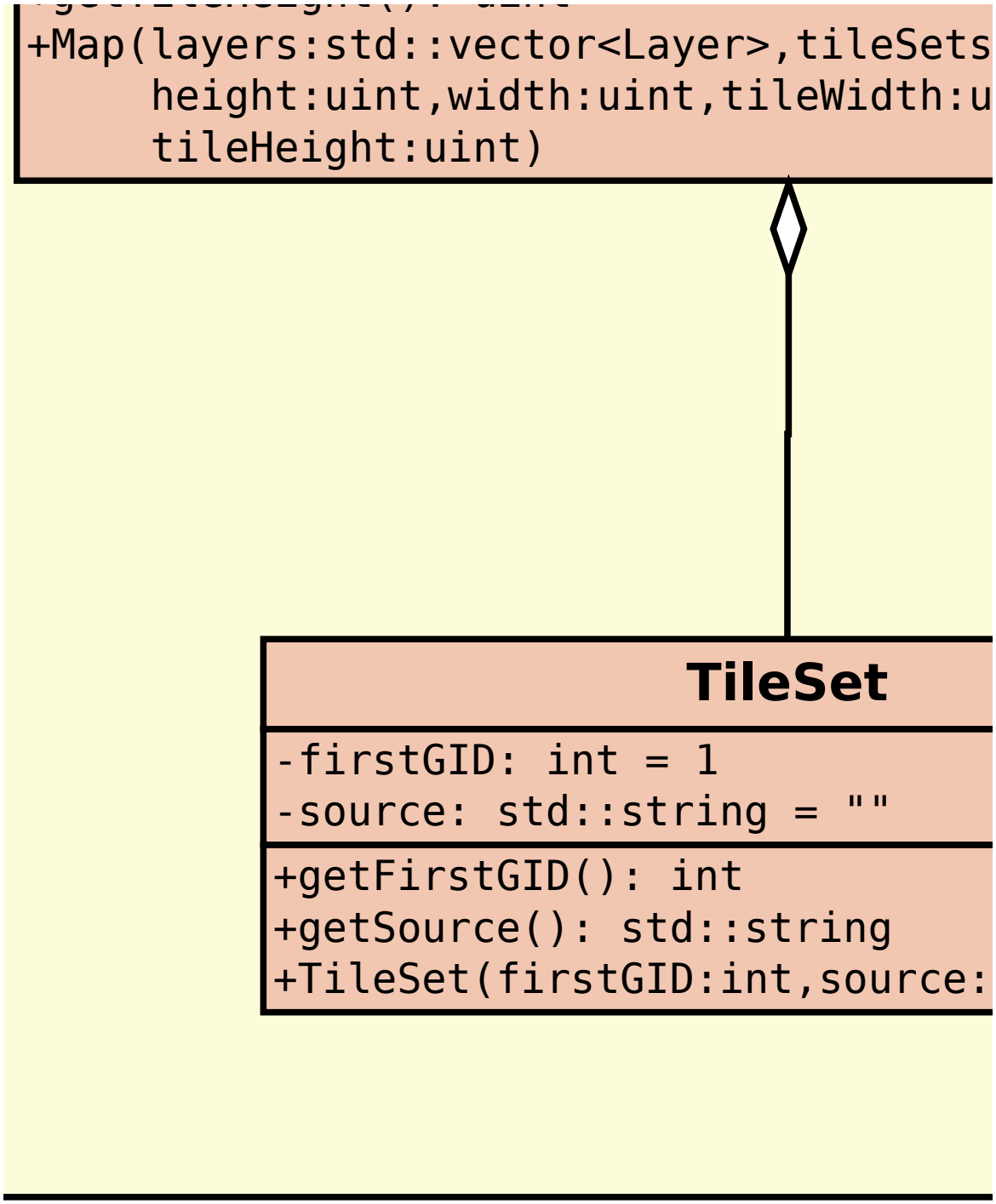


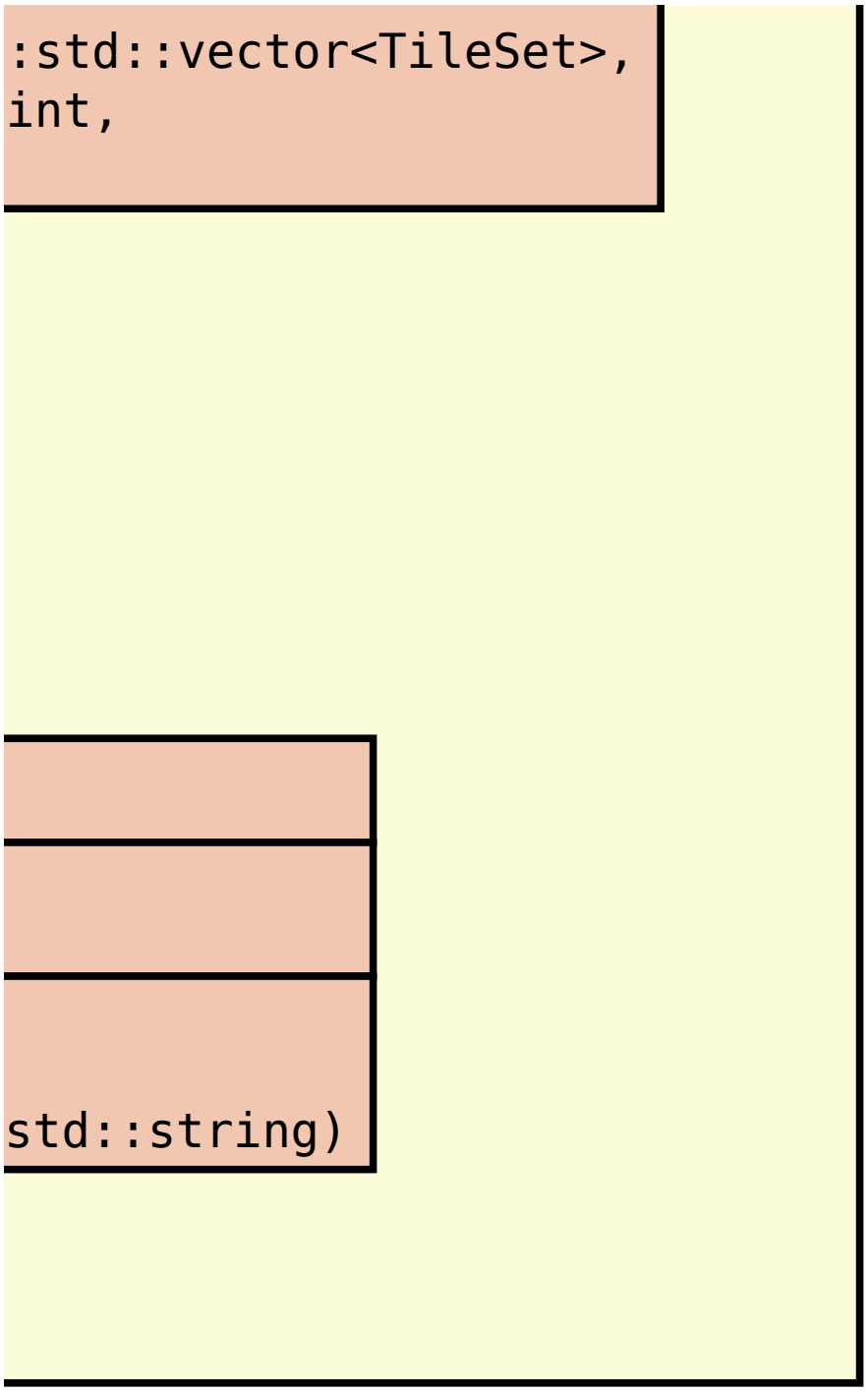



-data: std::
-height: uin
-name: std::
-width: uint
-x: int = 0
-y: int = 0
-image: std:
+getData():
+getHeight()
+getWidth():
+getX(): int
+getY(): int
+Layer(data:
heigh
+~Layer()
+getImage():


:shared_ptr<std::vector<uint>>
nt = 0
:string = ""
t = 0
::string
std::shared_ptr<std::vector<uint>>
): uint
: uint
t
t
:std::shared_ptr<std::vector<uint>>,
nt:uint,width:uint,x:int,y:int)
: std::string


*n


+Map(layers:std::vector<Layer>,tileSets
height:uint,width:uint,tileWidth:u
tileHeight:uint)
TileSet
-firstGID: int = 1
-source: std::string = ""
+getFirstGID(): int
+getSource(): std::string
+TileSet(firstGID:int,source:


:std::vector<TileSet>,
int,
std::string)