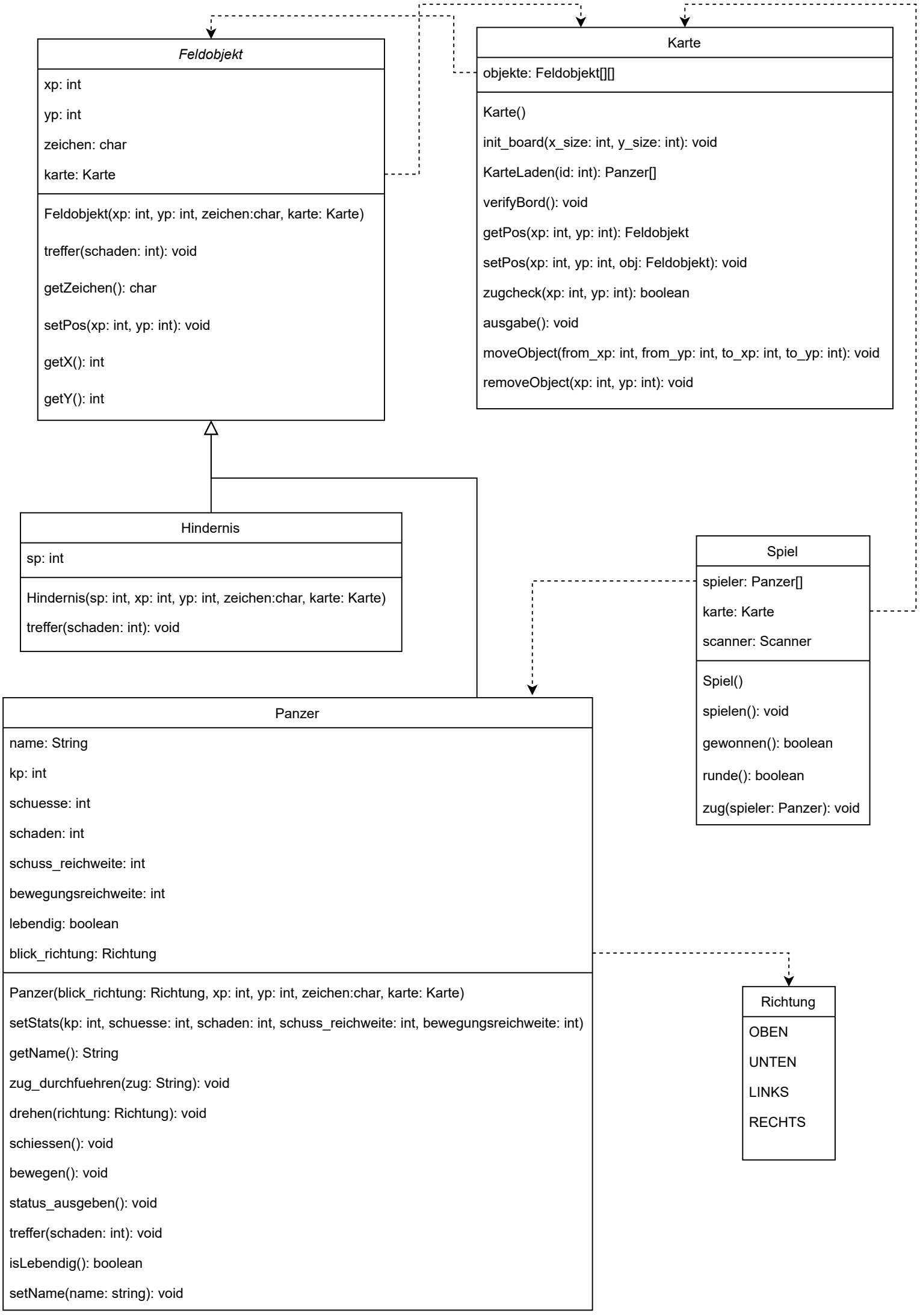
Feldobjekt
xp: int
yp: int
zeichen: char
karte: Karte
Feldobjekt(xp: int, yp: int, zeichen:char, karte: Karte)
treffer(schaden: int): void
getZeichen(): char
setPos(xp: int, yp: int): void
getX(): int
getY(): int
Karte
objekte: Feldobjekt[][]
Karte()
init_board(x_size: int, y_size: int): void
KarteLaden(id: int): Panzer[]
verifyBord(): void
getPos(xp: int, yp: int): Feldobjekt
setPos(xp: int, yp: int, obj: Feldobjekt): void
zugcheck(xp: int, yp: int): boolean
ausgabe(): void
moveObject(from_xp: int, from_yp: int, to_xp: int, to_yp: int): void
removeObject(xp: int, yp: int): void
Hindernis
sp: int
Hindernis(sp: int, xp: int, yp: int, zeichen:char, karte: Karte)
treffer(schaden: int): void
Spiel
spieler: Panzer[]
karte: Karte
scanner: Scanner
Spiel()
spielen(): void
gewonnen(): boolean
runde(): boolean
zug(spieler: Panzer): void
Panzer
name: String
kp: int
schuesse: int
schaden: int
schuss_reichweite: int
bewegungsreichweite: int
lebendig: boolean
blick_richtung: Richtung
Panzer(blick_richtung: Richtung, xp: int, yp: int, zeichen:char, karte: Karte)
setStats(kp: int, schuesse: int, schaden: int, schuss_reichweite: int, bewegungsreichweite: int)
getName(): String
zug_durchfuehren(zug: String): void
drehen(richtung: Richtung): void
schiessen(): void
bewegen(): void
status_ausgeben(): void
treffer(schaden: int): void
isLebendig(): boolean
setName(name: string): void
Richtung
OBEN
UNTEN
LINKS
RECHTS