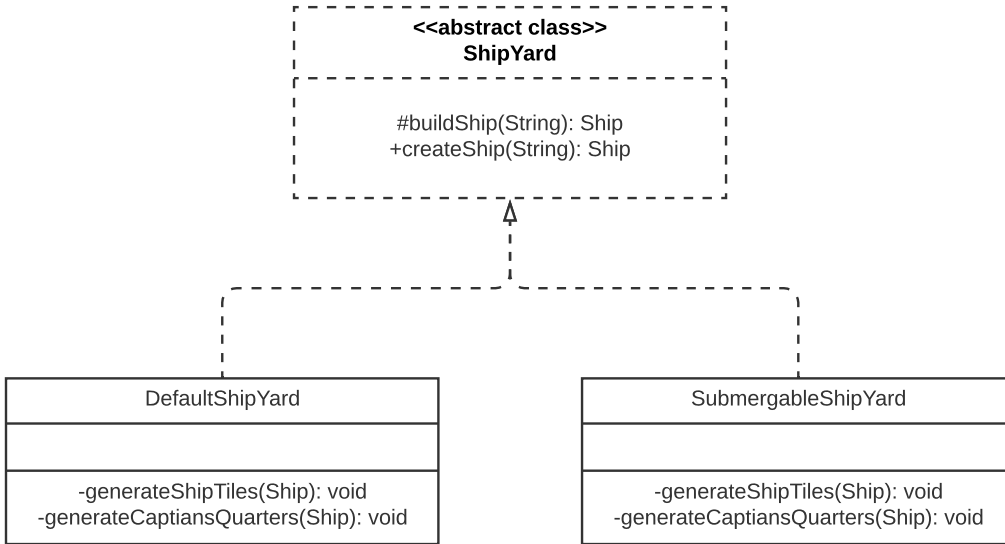# Factory Pattern
(Encapsulates Ship Creation)

Command Pattern
(Encapsulates undo stack)

<>
ShipYard

#buildShip(String): Ship
+createShip(String): Ship

DefaultShipYard

-generateShipTiles(Ship): void
-generateCaptiansQuarters(Ship): void

SubmergableShipYard

-generateShipTiles(Ship): void
-generateCaptiansQuarters(Ship): void

<<interface>>

Command

+execute(): boolean
+undo(): void

MoveFleetCommand

-moveDirection: Orientation
-player: Player

0..*   1

FleetControl

-fleetCommandStack:
Stack<MoveFleetCommand>
-player: Player

1

1

Player

- ign: String
- victories: int
-score: int
-missedShots: int
-totalShots: int

-game: Game
- board: Board
-weapons: List<Weapon>
-shipsToPlace: List<Ship>
-fleet: Fleet
-fleetController: FleetControl

-upgradeStatus: TierOneUpgrade
-mines: MineCollection
-powerUps: PowerupsCollection

+ placeShip(PlayerShipInput): boolean
+ attack(Board, PlayerFireInput):
List<AttackResult>
+moveFleet(Orientation): boolean
+placeMines(): void
+placePowerUps(): void