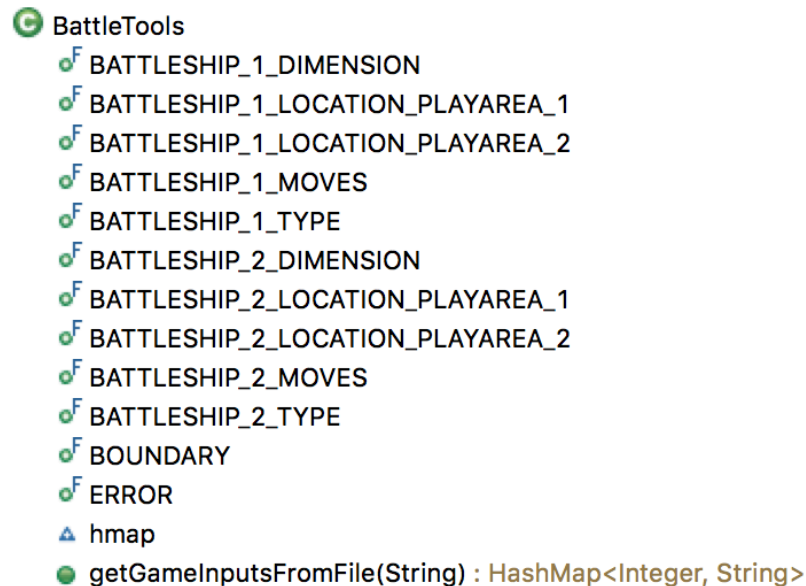


BattleTools

This class is used to perform support operations viz. IO operations etc. This class is responsible for reading input from the file, validating the inputs and storing it easily accessible hashmap viz. inputsMap.



BattleGround

This class represents the battlefield and has two battle-areas or play-areas for each player A and B. Each battle-area has two ships. The class evaluates all cell-coordinates representing dimensions of each ship in the respective battle-areas. These cell-coordinates are mapped, using two-dimensional array positions, to simple string literals viz. "E5", "B2".

BattleGround

- ▣ battleship1A
- ▣ battleship1B
- ▣ battleship1PositionPlayareaA
- ▣ battleship1PositionPlayareaB
- ▣ battleship2A
- ▣ battleship2B
- ▣ battleship2PositionPlayareaA
- ▣ battleship2PositionPlayareaB
- ▣ length
- ▣ movesPlayerA
- ▣ movesPlayerB
- 📦 positions
- ▣ width
- 📦 BattleGround(String)
- 🟢 getBattleGroundLength() : int
- 🟢 getBattleGroundWidth() : int
- 🟢 markAllBattleshipPositions() : void
- 🔴 markBattleshipPosition(HashMap<String, Integer>, int, int, int, int, int) : void
- 🟢 setBattleships(BattleShip, BattleShip, BattleShip, BattleShip) : void
- 🟢 setMovesPlayerA(String) : void
- 🟢 setMovesPlayerB(String) : void
- 🟢 start() : void

BattleShip

This class is used to encapsulate the core attributes of a battleship viz. Dimension of the ship, Location of the battleship, type of the ship viz. P or Q and in which playarea (A or B) the ship is stationed.

```

c BattleShip
  $ WEIGHT_OF_SHIP_TYPE_P
  $ WEIGHT_OF_SHIP_TYPE_Q
  length
  playarea
  type
  width
  xcoordinate
  ycoordinate
  c BattleShip(String, String, char, String)
  ● getLength() : int
  ● getPlayarea() : int
  ● getType() : int
  ● getWidth() : int
  ● getXpos() : int
  ● getYpos() : int

```

BattleShipGame

This is the main java application having System.main. This merely acts as the driver and invokes above classes to play the game.