

Report

Github link

[JavaAssignment](#)

Task

Task 1

Status: **completed**

Outlining: create `BattleShip.java` and extends `AbstractBattleShip` class. In constructor, use Random object to define the `shiporientation` and init other params. Write the member functions which define in `AbstractBattleShip`. Finish the `checkAttack` method. Check all possible situation. Define a array `shipCoordinatesRecord` to record the coordinate which has been hit.

```
BattleShip
  shipCoordinatesRecord
  BattleShip(String)
  checkAttack(int, int) : boolean
  getName() : String
  getHits() : int
  getShipOrientation() : String
  setHits(int) : void
  getShipCoordinates() : int[][]
  setShipCoordinates(int[][]) : void
```

Task 2

Status: **completed**

Outlining: Create `GameGrid.java` and extends `AbstractGameGrid`. Create the `initializeGrid()`. Finish the `generateShips()`. Finish the `placeShip()`. Create `PlayerGameGrid.java` and `OpponentGameGrid.java`, extends `GameGrid`. Define `printGrid()` in them.

```

GameGrid
GameGrid(int, int, int)
initializeGrid() : void
generateShips(int) : void
placeShip(BattleShip) : void
main(String[]) : void

```

Task 3

Status: **completed**

Outlining: Create `Game.java` and implement `GameControls`. create player's and opponent's grid and implement getter methods of them. Implement `exitGame()` method. Implement `checkVictory()`. Implement `playRound()` to make robot smart. In this case, opponent decision depends on the probability which calculate from "X" coordinates. If all coordinates are less than or equal to 0, its decision depends on the probability which calculate from the number of unknown coordinates. Unfortunately, the fact is I will lose to the robot in most cases.

```

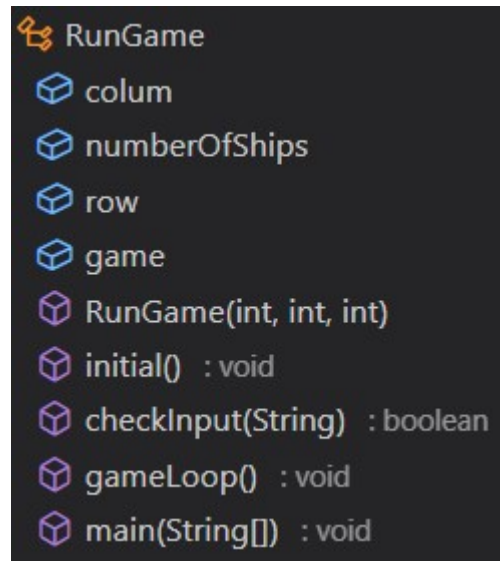
Game
playerGameGrid
opponentGameGrid
Game(int, int, int)
playRound(String) : void
decision() : int[]
opponentRound() : void
checkVictory() : boolean
exitGame(String) : void
getPlayersGrid() : GameGrid
getOpponentssGrid() : GameGrid

```

Task 4

Status: **completed**

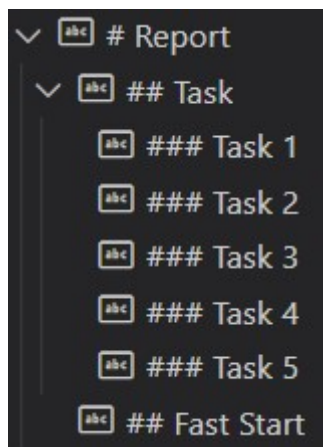
Outlining: Create `RunGame.java` and start it by create a `Game` object. Use params to get the height, hight and the number of ships. Use while loop function to make sure rounds are loop. Use regex `^[0-9]+, [0-9]+$` to match corret input and use `exitGame()` function which in game object to check the `exit`.



Task 5

Status: **completed**

Outlining: Conclude the tasks status and describe each with a short words.



Fast Start

Switch to the JAVA project root directory `assignment`

- package the project

```
mvn package
```

- run the jar with you params which are **height**, **width** and **number of ships**

```
java -jar .\target\assignment-1.0-SNAPSHOT.jar 5 5 3
```