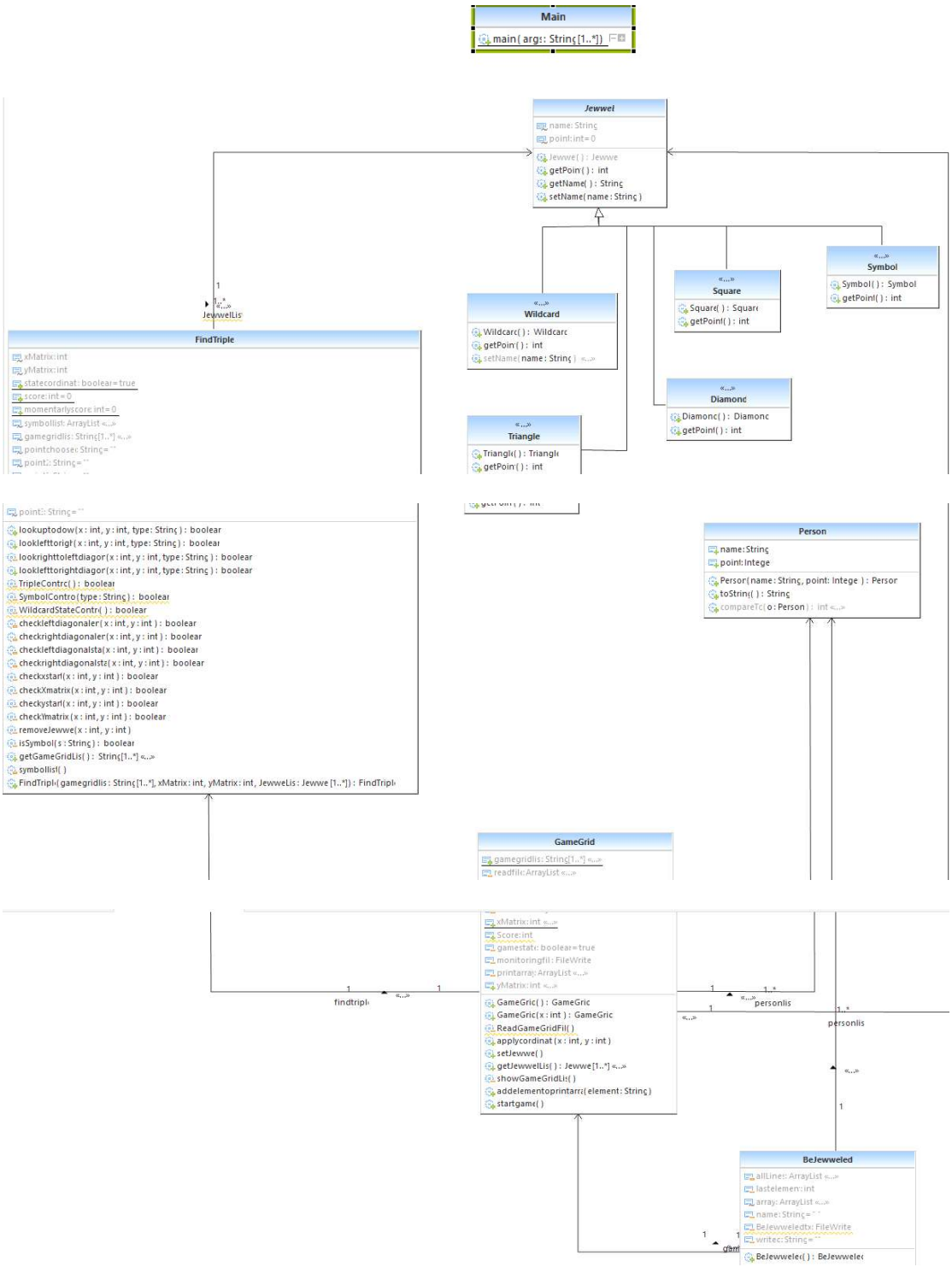
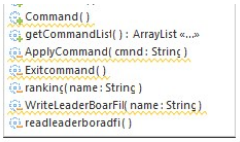
Main Class calls BeJewwled class. BeJewweled class has command method. It gets all commands from txt file and proccesing them.It does action by using GameGrid class. Send x and y cordinate to GameGrid class . When All commands end, ExitCommand() method runs and all of data are written to monitoring txt file. GameGrid class initially is setting jewwels. Then, get data from gamegrid.txt file and store them in matrix. Whenever gets data from matrix through applycordinate method, it performs action depends on jewwel type. if jewwel exist it sends data to FindTriple class for searching if triple exists. FindTriple class has load of methods. It is performing action up to triple type so it works effiecently. It has a lot of contol for each of jewwel types.If triple exist it calls remove jewwel method. and all jewwels are in triple are removed one by one. Then monitoring gamegrid list and score. Another class is jewwel. Jewwel is absract class. Each of element has class on the page of Jewwel class. Jewwel class stores nama and point of jewwels. Person and PersonComparator class were created in order to compare persons associated with their points. They implements Comparator<Person>.


Main
main(args: String[1..*])
Jewwel
name: String
point:int=0
Jewwe( ): Jewwe
getPoin( ): int
getName( ): String
setName(name: String)
Symbol
Symbol( ): Symbol
getPoint( ): int
Square
Square( ): Square
getPoint( ): int
Wildcard
Wildcard( ): Wildcard
getPoin( ): int
Diamond
Diamond( ): Diamond
getPoint( ): int
Triangle
Triangle( ): Triangle
getPoin( ): int
FindTriple
xMatrix:int
yMatrix:int
statecordinat: boolean=true
score:int=0
symbollist: ArrayList
gamegridlis: String[1..*]
point:: String=""
lookuptodow(x: int, y: int, type: String): boolean
lookleftorigh(x: int, y: int, type: String): boolear
lookrighttoleftdiagor(x: int, y: int, type: String): boolear
lookleftorightdiagor(x: int, y: int, type: String): boolear
TripleContrc( ): boolear
SymbolContro(type: String): boolear
WildcardStateContr( ): boolear
checkleftdiagonaler(x: int, y: int): boolear
checkrightdiagonaler(x: int, y: int): boolear
checkleftdiagonalsta(x: int, y: int): boolear
checkrightdiagonalsta(x: int, y: int): boolear
checkxstarl(x: int, y: int): boolear
checkXmatrix(x: int, y: int): boolear
checkystarl(x: int, y: int): boolear
checkYmatrix(x: int, y: int): boolear
removeJewwe(x: int, y: int)
isSymbol(s: String): boolear
getGameGridLis( ): String[1..*]
symbollist( )
FindTripl(gamegridlis: String[1..*], xMatrix: int, yMatrix: int, JewweLis: Jewwe[1..*]): FindTripl
Person
name: String
point: Intege
Person(name: String, point: Intege): Persor
toStrin( ): String
compareTo(o: Person): int
GameGrid
gamegridlis: String[1..*]
readfile: ArrayList
xMatrix: int
Score: int
gamestatr: boolear=true
monitoringfil: FileWrite
printarray: ArrayList
yMatrix: int
GameGric( ): GameGric
GameGric(x: int): GameGric
ReadGameGridFil( )
applycordinat(x: int, y: int)
setJewwe( )
getJewwelLis( ): Jewwe[1..*]
showGameGridLis( )
addelementoprintarr(element: String)
startgame( )
personlis
BeJewweled
allLines: ArrayList
lastelemen: int
array: ArrayList
name: String=""
BeJewweledtx: FileWrite
wntec: String=""
BeJewwele( ): BeJewwele


Command( )
getCommandList( ) : ArrayList «...»
ApplyCommand(cmnd : String)
Exitcommand( )
ranking(name : String)
WriteLeaderBoarFil(name : String)
readleaderboradfi( )


PersonComparato
compare(o1 : Persor, o2 : Person) : int «...»