public class Rook extends Piece{ //Entire Class Coded by Aman Modi

private boolean rMove;

public Rook(int r, int c, String s, String n){

super(r, c, s, n);

}

public boolean movePiece(int r, int c, String co){

boolean status;

if(!co.equals(getColor()))

return false;

if(getRow() == r)

status = validateHorzMovement(r, c);

else if(getCol() == c)

status = validateVertMovement(r, c);

else

return false;

rMove = true;

return status;

}

public boolean validateHorzMovement(int r, int c){

int currentRow = getRow();

int currentCol; boolean upDown;

if(c < getCol()){

currentCol = getCol() - 1;

upDown = true;

}

else if(c > getCol()){

currentCol = getCol() + 1;

upDown = false;

}

else{

return false;

}

while(currentCol > -1 && currentCol < array.length){

if(currentCol == c){

boolean status1 = checkOpen(r, c);

boolean status2 = capturePiece(r, c);

if(status1 || status2){

changePosition(r, c);

return true;

}

else

return false;

}

else if(currentCol != c){

boolean status = checkOpen(currentRow, currentCol);

if(!status){

return false;

}

}

if(upDown)

currentCol--;

else

currentCol++;

}

return true;

}

public boolean validateVertMovement(int r, int c){

int currentCol = getCol();

int currentRow; boolean upDown;

if(r < getRow()){

currentRow = getRow() - 1;

upDown = true;

}

else if(r > getRow()){

currentRow = getRow() + 1;

upDown = false;

}

else{

return false;

}

while(currentRow > -1 && currentRow < array.length){

if(currentRow == r){

boolean status1 = checkOpen(r, c);

boolean status2 = capturePiece(r, c);

if(status1 || status2){

changePosition(r, c);

return true;

}

else

return false;

}

else if(currentRow != r){

boolean status = checkOpen(currentRow, currentCol);

if(!status){

return false;

}

}

if(upDown)

currentRow--;

else

currentRow++;

}

return true;

}

public boolean getRMove()

{

return rMove;

}

}