**Moving Rectangle Game**

**Class Player**

Attributes:

**x** – x axis of the rectangle upper left corner  
**y** – y axis of the rectangle upper left corner  
**width** – rectangle width  
**height** – rectangle height  
**color** - rectangle color

**rect** – rectangle tuple for drawing  
**vel** – rectangle moving speed

Methods:

**\_\_int\_\_(self,x,y,width,height,color)**

**draw(self,win)** – drawing rectangle

**move(self)** – updating rectangle position   
 according to pressed ARROWS KEYS

**update(self)** – updating rectangle TUPLE

**Class Network**

Methods:

**\_\_init\_\_(self)** – creating client socket and   
 connecting to server

**getP(self)** – getting player object

**connect(self)** – connecting and receiving   
 initial player object from server

**send(self,data)** – sending own player object   
 server and receiving   
 opponent player object

**Client script**

Methods:

redrawWindow(win, player, player2)

-

* Creating pygame display
* Creating client socket by utilizing Network class
* Getting player object from server immediately after connection
* Generating pygame clock

**Infinite loop:**

* Setting clock REFRESH RATE to 60
* Sending own player object to server and immediately receiving opponent player object from server.
* Handling pygame.QUIT event
* Moving player by calling p.move() function
* Redrawing window by calling redrawWindow function

**Server script**

Methods:

**threaded\_client()** – for each player(client) a   
 threaded function is   
 invoked.

* Creating server socket
* bind
* listen(2)
* Creating two player with initial postion

**Infinite loop:**

* Connection accept()
* start new threaded client function with appropriate clientID

}

* Sending player object to client with appropriate initial position
* **Infinite loop:**waiting for client data (receive)  
  updating players list  
  sending the opponent player object to client

{