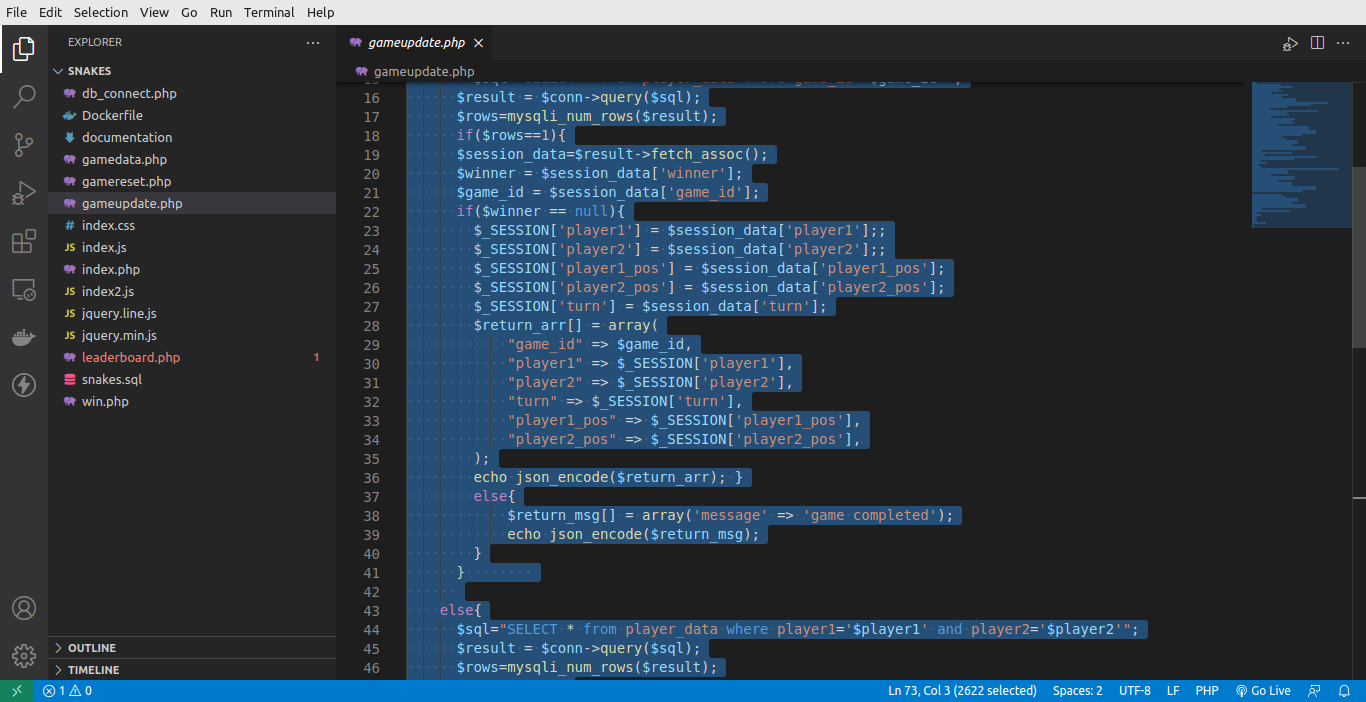
<paste your screen shots here>

*Sample screenshot of GameBoard*



<paste your screen shots here>

*Sample folder structure for all COM549 Assignment work*



Code for Game using the Preview in h[ttp://hilite.me](http://hilite.me/)

HTML

*(Sample input to create the output required)*



*HTML selected with NO line numbers with Preview copied and pasted as text*

<paste your code here from [http://hilite.me](http://hilite.me/)>

<!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="utf-8">

<title></title>

<link rel="stylesheet" href="./index.css">

</head>

<body>

<div>

<div class="grid-container">

<div id='table-canvas'>

<table class="grid-item board" id="board">

</table>

<canvas id="myCanvas" width="500px" height="500px">

</canvas>

</div>

<button id='showlead'>show leaderboard</button>

<div class="grid-item player" style='display:none'>

<h4 id='player-tag'>Player 1</h4>

<div class="dice">

<button type="button" name="button" id='roll-dice'>Roll Dice</button>

<button type="button" name="button" id='reset'>Reset</button>

<input type="text" name="" value="0" id="dice-value">

</div>

</div>

<form id='startform' method='POST'>

<input type="text" placeholder='player1' name='player1' id='player1'>

<input type="text" placeholder='player2' name='player2' id='player2'>

<input type="submit" value="Start game" name='submit' id='startgame'>

</form>

</div>

<script src="./jquery.min.js"></script>

<script src='./jquery.line.js'></script>

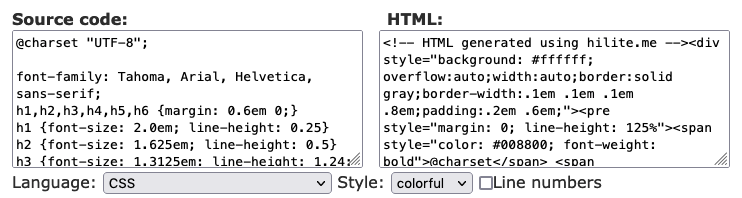
<script type="text/javascript" src='./index.js'></script>

</div>

</body>

</html>

CSS



*CSS selected* *with NO line numbers with Preview copied and pasted as text*

<paste your code here from [http://hilite.me](http://hilite.me/)>

**.grid-container**{

**display**: grid;

column-gap: **5px**

}

**.board**{

grid-column-start: **1**;

grid-column-end: **12**;

**background-color**: **rgb**(**62**, **188**, **195**);

**margin-bottom**:**3**%;

**margin-left**:**5**%;

**margin-top**: **7**%;

}

**.player**{

**position**: **absolute**;

**top**:**20**%;

grid-column-start: **1**;

grid-column-end: **12**;

**background-color**: **rgb**(**189**, **242**, **172**);

**margin-left**:**50**%

}

**.player-tag**{

**padding-left**:**40**%

}

**.status**{

**margin-left**:**25**%;

}

**.dice**{

**padding-left**: **25**%

}

**#board**{

**width**:**50px**

}

**#startform**{

**position**: **absolute**;

**top**:**20**%;

**left**:**50**%;

grid-column-start: **6**;

grid-column-end: **12**;

**background-color**: **rgb**(**189**, **242**, **172**);

**background-color**: **#07fa3c**;

}

button{

**width**:**10**vw;

**height**:**10**vh

}

input{

**margin-top**: **10px**;

**height**: **10**vh;

}

**#person1**{

**padding**: **0**%;

**margin**: **0**%;

**background-color**: blue;

**position**: **absolute**;

**width**:**15px**

}

**#person2**{

**background-color**: **rgb**(**219**, **111**, **9**);

**position**: **absolute**;

**padding**: **0**%;

**margin-left**: **1**%;

**width**:**15px**;

**color**:'white'

}

**#table-canvas**{

**display**: flex;

}

**#myCanvas**{

**display**:'none';

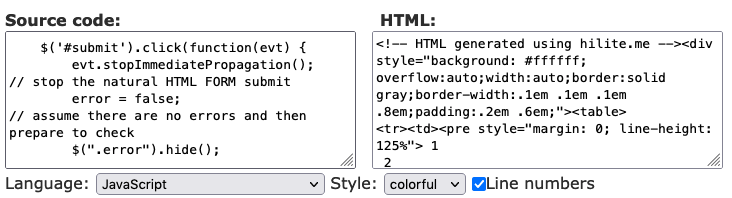
**margin-left**: **200px**;

**margin-top**: **100px**;

**position**: **absolute**;

}

jQuery



*JavaScript selected* *WITH line numbers with Preview copied and pasted as text*

<paste your code here from [http://hilite.me](http://hilite.me/)>

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198  199  200  201  202  203  204  205  206  207  208  209  210  211  212  213  214  215  216  217  218  219  220  221  222  223  224  225  226  227  228  229  230  231  232  233  234  235  236  237  238  239  240  241  242  243  244  245  246  247  248  249  250  251  252  253  254  255  256  257  258  259  260  261  262  263  264  265  266  267  268  269  270  271  272  273  274  275  276  277  278  279  280  281  282  283  284  285  286  287  288  289  290  291  292  293  294  295  296  297  298  299  300  301  302  303  304  305  306  307  308  309  310 | $("#startform").submit(**function**( event ) {  **var** $startform = $( **this** ),  player1 = $startform.find('#player1').val(),  player2 = $startform.find('#player2').val();  event.preventDefault();  data = {  player1: player1,  player2: player2  }  $.post("gamedata.php",  data,  get\_response= (response)=>{  res = JSON.parse(response)  $("#startform").css({'display':'none'})  $(".player").css({'display':'block'})  initial(res)  }  )  **const** initial = (res)=>{  $player1 = $('<div id="person1">1</div>')  $player2 = $('<div id="person2">2</div>')  res.map(val=>{  turn = val.turn  $player1\_pos = parseInt(val.player1\_pos)  $player2\_pos = parseInt(val.player2\_pos)  })  // $player1\_pos = 1  // $player2\_pos = 1  $('#person1').css({'position':$(`#${$player1\_pos}`).position()})  $('#person2').css({'position':$(`#${$player2\_pos}`).position()})  startPosition()  }  **function** startPosition () {  $(`#${$player1\_pos}`).append($player1)  $(`#${$player2\_pos}`).append($player2)  }  **var** $roll\_dice = $('#roll-dice')  $roll\_dice.on('click', rollDice)  snake1\_entry = **28**  snake1\_exit = **20**  snake2\_entry = **34**  snake2\_exit = **22**  ladder1\_entry = **19**  ladder1\_exit = **33**  ladder2\_entry = **21**  ladder2\_exit = **31**  ladder3\_entry = **15**  ladder3\_exit = **27**  **var** canvas = document.getElementById("myCanvas");  **var** snake1 = canvas.getContext("2d");  **var** snake2 = canvas.getContext("2d");  **var** ladder1 = canvas.getContext("2d");  **var** ladder2 = canvas.getContext("2d");  **var** ladder3 = canvas.getContext("2d");  **function** rollDice(){  **const** dice\_value = Math.floor(Math.random() \* **6**)+**1**  **var** $die = $('#dice-value')  $die.val(dice\_value)  console.log(dice\_value)  playerTurn(dice\_value)  }  **function** playerTurn (dice\_value) {  **if** (turn == 'player1') {  $('#player-tag').text('Player 2')  $player1\_pos += dice\_value  $player1.appendTo(`#${$player1\_pos}`)  **switch** ($player1\_pos) {  **case** snake1\_entry:  $player1\_pos=snake1\_exit  snake1.beginPath()  snake1.moveTo(**100**,**90**);  snake1.lineTo(**150**,**140**);  snake1.strokeStyle = "#fa2b07";  snake1.stroke();  **break**;  **case** snake2\_entry:  $player1\_pos=snake2\_exit  snake2.beginPath()  snake2.moveTo(**20**,**10**);  snake2.lineTo(**10**,**150**);  snake2.strokeStyle = "#fa2b07";  snake2.stroke();  snake2.stroke();  **break**  **case** ladder1\_entry:  $player1\_pos=ladder1\_exit  ladder1.beginPath()  ladder1.moveTo(**100**,**30**);  ladder1.lineTo(**200**,**150**);  ladder1.strokeStyle = "#07fa3c";  ladder1.stroke();  **break**  **case** ladder2\_entry:  $player1\_pos=ladder2\_exit  ladder2.beginPath()  ladder2.moveTo(**200**,**30**);  ladder2.lineTo(**100**,**150**);  ladder2.strokeStyle = "#07fa3c";  ladder2.stroke();  **break**  **case** ladder3\_entry:  $player1\_pos=ladder3\_exit  ladder3.beginPath()  ladder3.moveTo(**0**,**200**);  ladder3.lineTo(**9**,**80**);  ladder3.strokeStyle = "#07fa3c";  ladder3.stroke();  **break**  **default**:  **break**;  }  $player1.appendTo(`#${$player1\_pos}`)  nextplayer = 'player2'  console.log($player1\_pos)  }  **if** (turn == 'player2') {  $('#player-tag').text('Player 1')  $player2\_pos += dice\_value  $player2.appendTo(`#${$player2\_pos}`)  **switch** ($player2\_pos) {  **case** snake1\_entry:  $player2\_pos=snake1\_exit  snake1.beginPath()  snake1.moveTo(**100**,**90**);  snake1.lineTo(**150**,**140**);  snake1.strokeStyle = "#fa2b07";  snake1.stroke();  **break**;  **case** snake2\_entry:  $player2\_pos=snake2\_exit  snake2.beginPath()  snake2.moveTo(**20**,**10**);  snake2.lineTo(**10**,**150**);  snake2.strokeStyle = "#fa2b07";  snake2.stroke();  snake2.stroke();  **break**  **case** ladder1\_entry:  $player2\_pos=ladder1\_exit  ladder1.beginPath()  ladder1.moveTo(**100**,**30**);  ladder1.lineTo(**200**,**150**);  ladder1.strokeStyle = "#07fa3c";  ladder1.stroke();  **break**  **case** ladder2\_entry:  $player2\_pos=ladder2\_exit  ladder2.beginPath()  ladder2.moveTo(**200**,**30**);  ladder2.lineTo(**100**,**150**);  ladder2.strokeStyle = "#07fa3c";  ladder2.stroke();  **break**  **case** ladder3\_entry:  $player2\_pos=ladder3\_exit  ladder3.beginPath()  ladder3.moveTo(**0**,**200**);  ladder3.lineTo(**9**,**80**);  ladder3.strokeStyle = "#07fa3c";  ladder3.stroke();  **break**  **default**:  **break**;  }  $player2.appendTo(`#${$player2\_pos}`)  nextplayer = 'player1'  console.log($player2\_pos)  }  //console.log(nextplayer)  haswon()  dbupdate()  }  **function** dbupdate(){  console.log(res)  res.map(data=>{  game\_id = data.game\_id  })  data = {  game\_id:game\_id,  turn: nextplayer,  player1\_pos:$player1\_pos,  player2\_pos:$player2\_pos,  }  //console.log(data)  $.post("gameupdate.php",  data,  **function**(response){  console.log(response)  res = JSON.parse(response)  res.map(val=>{  turn = val.turn  })  }  )  }  **function** haswon () {  **if** ($player1\_pos >= **36**) {  res.map(data=>{  game\_id = data.game\_id  $player1 = data.player1  })  data = {  game\_id:game\_id,  turn:nextplayer,  player1\_pos:$player1\_pos,  player2\_pos:$player2\_pos,  winner:$player1,  }  console.log(data)  $.post("win.php",  data,  )  alert('Player 1 has won!')  } **else** **if** ($player2\_pos >= **36**) {  res.map(data=>{  game\_id = data.game\_id  $player2 = data.player2  })  data = {  game\_id:game\_id,  turn:nextplayer,  player1\_pos:$player1\_pos,  player2\_pos:$player2\_pos,  winner:$player2,  }  console.log(data)  $.post("win.php",  data,  )  alert('Player 2 has won!')  }  }  })  $('#reset').click(resetGame)  **function** resetGame () {  **var** $die = $('#dice-value')  $die.val(**0**)  $player1\_pos = **1**  $player2\_pos = **1**  console.log($('#1').position())  $('#person1').css({'position':$('#1').position()})  $('#person2').css({'position':$('#1').position()})  resetPosition($player1, $player2)  **function** resetPosition ($player1, $player2) {  $('#1').append($player1)  $('#1').append($player2)  }  res.map(data=>{  game\_id = data.game\_id  })  data = {  game\_id:game\_id  }  console.log(data.game\_id)  $.post("gamereset.php",  data,  **function**(response){  console.log(response)  }  )  }  **function** createTable () {  **var** $board = $('#board')  **var** id = **36**  **var** rowClass = **6**  **for** (**var** r = **0**; r < **6**; r++) {  **var** $row = $('<tr>')  $row.attr('class', rowClass--)  **for** (**var** c = **0**; c < **6**; c++) {  **var** $column = $('<td>')  $column.css({ 'width': '800px', 'height': '0px', padding:'20px'})  $column.attr('id', id--)  $column.attr('id') % **2** === **0** ? $column.css('backgroundColor', '#ffff00') : $column.css('backgroundColor', 'white') // giving grid bg colors.  $row.each(**function**() {  $(**this**).attr('class') % **2** === **0** ? $row.append($column) : $row.prepend($column)  })  $column.html(id + **1**).addClass('cell')  }  $board.append($row)  }  }  createTable()  clicked = **1**  $('#showlead').click(showleaderboard)  **function** showleaderboard(){  **if**(clicked%**2**!=**0**){  $('#leaderboard').css({'display':'flex'})  //return clicked+=1  }  **else**{  $('#leaderboard').css({'display':'none'})  }  **return** clicked+=**1**  } |

PHP

Graphical user interface, text, application, email

Description automatically generated

<paste your code here from [http://hilite.me](http://hilite.me/)>

db\_connect.php

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | <?php  $username = 'allan';  $password = 'mXy2p t1-!k ';  $server = 'localhost';  $database = 'snakes';  $conn = **new** mysqli($server, $username, $password, $database);  ?> |

**gamedata.php**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76 | <?php  ini\_set('display\_errors', **1**);  ini\_set('display\_startup\_errors', **1**);  error\_reporting(**E\_ALL**);  **include**('db\_connect.php');  session\_start();  $player1=$\_REQUEST['player1'];  $player2=$\_REQUEST['player2'];  $sql="SELECT \* from player\_data where player1='$player1' and player2='$player2'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $winner = $session\_data['winner'];  $game\_id = $session\_data['game\_id'];  **if**($winner == **null**){  $\_SESSION['player1'] = $player1;  $\_SESSION['player2'] = $player2;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $\_SESSION['winner'] = $winner;  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  "winner"=> $winner  );  **echo** json\_encode($return\_arr);  }  **else**{  $return\_msg[] = **array**('message' => 'game completed');  **echo** json\_encode($return\_msg);  }  }  **else**{  $sql="INSERT INTO `player\_data` (`game\_id`, `player1`, `player2`, `player1\_pos`, `player2\_pos`, `winner`) VALUES  (NULL, '$player1', '$player2', '1', '1', NULL);";  $result = $conn->query($sql);  **if**($result){  $sql="SELECT \* from player\_data where player1='$player1' and player2='$player2'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $game\_id = $session\_data['game\_id'];  $\_SESSION['player1'] = $player1;  $\_SESSION['player2'] = $player2;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $\_SESSION['winner'] = $session\_data['winner'];  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  "winner"=> $\_SESSION['winner']  );  **echo** json\_encode($return\_arr);  }  **else**{  **echo** 'failed';  }  }  }  ?> |

**gamereset.php**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66 | <?php  ini\_set('display\_errors', **1**);  ini\_set('display\_startup\_errors', **1**);  error\_reporting(**E\_ALL**);  **include**('db\_connect.php');  session\_start();  $game\_id=$\_REQUEST['game\_id'];  $sql="UPDATE player\_data set player1\_pos='1', player2\_pos='1', turn='player1',winner=null  where game\_id='$game\_id'";  $result = $conn->query($sql);  **if**($result){  $sql="SELECT \* from player\_data where game\_id='$game\_id'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $winner = $session\_data['winner'];  $game\_id = $session\_data['game\_id'];  **if**($winner == **null**){  $\_SESSION['player1'] = $session\_data['player1'];;  $\_SESSION['player2'] = $session\_data['player2'];;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $\_SESSION['winner'] = $winner;  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  );  **echo** json\_encode($return\_arr);  }  }    **else**{  $sql="SELECT \* from player\_data where player1='$player1' and player2='$player2'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $game\_id = $session\_data['game\_id'];  $winner = $session\_data['winner'];  **if**($winner == **null**){  $\_SESSION['player1'] = $player1;  $\_SESSION['player2'] = $player2;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  );  **echo** json\_encode($return\_arr);  }  }  }  }  ?> |

**gameupdate.php**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73 | <?php  ini\_set('display\_errors', **1**);  ini\_set('display\_startup\_errors', **1**);  error\_reporting(**E\_ALL**);  **include**('db\_connect.php');  session\_start();  $game\_id=$\_REQUEST['game\_id'];  $turn=$\_REQUEST['turn'];  $player1\_pos=$\_REQUEST['player1\_pos'];  $player2\_pos=$\_REQUEST['player2\_pos'];  $sql="UPDATE player\_data set turn='$turn',player1\_pos='$player1\_pos',  player2\_pos='$player2\_pos' where game\_id='$game\_id'";  $result = $conn->query($sql);  **if**($result){  $sql="SELECT \* from player\_data where game\_id='$game\_id'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $winner = $session\_data['winner'];  $game\_id = $session\_data['game\_id'];  **if**($winner == **null**){  $\_SESSION['player1'] = $session\_data['player1'];;  $\_SESSION['player2'] = $session\_data['player2'];;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  );  **echo** json\_encode($return\_arr); }  **else**{  $return\_msg[] = **array**('message' => 'game completed');  **echo** json\_encode($return\_msg);  }  }    **else**{  $sql="SELECT \* from player\_data where player1='$player1' and player2='$player2'";  $result = $conn->query($sql);  $rows=mysqli\_num\_rows($result);  **if**($rows==**1**){  $session\_data=$result->fetch\_assoc();  $game\_id = $session\_data['game\_id'];  **if**($winner == **null**){  $\_SESSION['player1'] = $player1;  $\_SESSION['player2'] = $player2;  $\_SESSION['player1\_pos'] = $session\_data['player1\_pos'];  $\_SESSION['player2\_pos'] = $session\_data['player2\_pos'];  $\_SESSION['turn'] = $session\_data['turn'];  $return\_arr[] = **array**(  "game\_id" => $game\_id,  "player1" => $\_SESSION['player1'],  "player2" => $\_SESSION['player2'],  "turn" => $\_SESSION['turn'],  "player1\_pos" => $\_SESSION['player1\_pos'],  "player2\_pos" => $\_SESSION['player2\_pos'],  );  **echo** json\_encode($return\_arr);  }  }  }  }  **else**{  **echo** **null**;  }  ?> |

Sources

<list your sources you used outside of the module material>