

# Maths Homework Generator

**Name:** Roshni

# Objectives of System

## *Admin*

1. The admin must be able to view all links between each student and their teacher
2. The admin must be able to delete links between a student and teacher that are no longer at Kidz POTC
3. The admin must be able to view all links between each student and their parent
4. The admin must be able to delete links between a student and parent that are no longer at Kidz POTC
5. The admin must be able to change a code for a homework sheet

## *Database*

1. The database must be able to record to progress of each student
2. The database must be able to record student details such as name, date of birth and school for each student
3. The database must be able to record parent details such as first name, last name and email address for each parent
4. The database must be able to record teacher details such as first name, last name and email address for each teacher
5. The database must be able to hold records of up to 250 students

## *Parent*

1. The system should automatically send emails to parents when a homework has been completed with the score their child got
2. Parent should know how long child took to complete homework, via the email
3. Parent should be able to log in and view their family
4. Parent should be able to build a family
5. Parent should be able to access their child's overall scores for each homework sheet

## *Student*

1. The system must be able to handle a minimum of 5 students completing the homework at once
2. In order to access the system, the student should only have to enter their username, password and code for that particular homework sheet
3. Once completed the 10 questions, the students should be able to see a feedback sheet for that homework attempt which includes score and a general comment about the overall attempt
4. The student should be able to see their score for each question on the feedback sheet
5. The student should be able to see percent for each type of question on the feedbacksheet
6. Students should know how long they took to complete the homework

### **Question Generator System (for students)**

1. The questions generated for each topic should be random
2. The numbers used in each question should be randomised
3. Once a student completes a question, there should be a comment (i.e. green tick ✓ or red cross ✗)
4. The question styles should be a mix of multiple-choice questions, short answer questions and long answer questions
5. For multiple choice questions, the student should be able to select what they think the right answer is
6. For short and long answer questions, the students should be able to type what they think the right answer is
7. For long questions, the questions should be split into 2 parts
8. If a student gets any question incorrect, there should be a mini explanation as to what the correct method is and the correct final answer
9. The numbers used in the randomly generated questions should be whole numbers, or decimal where appropriate
10. The correct answers to each problem should be whole numbers, or decimal where appropriate
11. Each option in the multiple-choice questions should be whole numbers, or decimal where appropriate
12. For multiple choice, short answer and long answer questions, the student may have to provide an explanation of how they reached the final answer
13. If students get 3 questions in a row correct, then increase the difficulty level
14. If students get 3 questions in a row incorrect, then decrease the difficulty level

### **Teacher**

1. The system should automatically send emails to teachers when a homework has been completed with the score their student got
2. Teachers should also know the students' progress in comparison to other topics
3. Teacher should know how long child took to complete homework, via the email
4. Allow the teacher to print/ save as a pdf a particular students questions, scores and attempts for a particular test

# Screenshots of the system

Maths @ Kidz

Username

Password

Submit

If you don't have an account, register [here](#)

If you forgot your password, please click [here](#)

### Student Sign Up

First Name

Surname

School

Please select your school

Date of birth

dd/mm/yyyy

Username

Password

Re enter password

Submit

**TOPIC: Addition QUESTION NUMBER : 8**

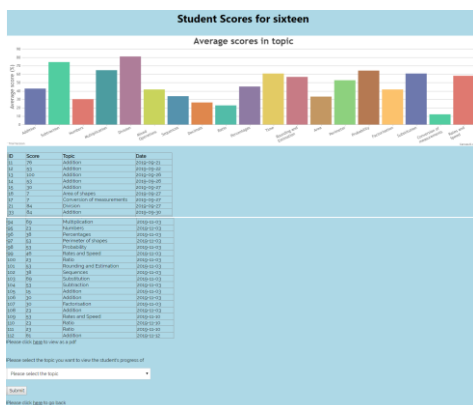
A) What is  $85 + 57$ ? B) What is  $85 + 57 + 88$ ?

Part A

Part B

Reasoning for answers

Submit



**TOPIC: Addition QUESTION NUMBER : 1**

What is  $43 + 44$ ?

☐ 87  
☐ 88  
☐ 86

Reasoning for answer chosen

Submit

**TOPIC: Addition QUESTION NUMBER : 4**

Claire went to the shop and bought 81 chocolate chip cookies, 34 milk chocolate cookies and 74 white chocolate cookies. How many did she buy in total?

Reasoning for answer

Submit

The question was : What is  $43 + 44$ ?

The answer you chose was : correct

☒

Next Question

The question was : Claire went to the shop and bought 81 chocolate chip cookies, 34 milk chocolate cookies and 74 white chocolate cookies. How many did she buy in total?

Your response was 100

This was : wrong

☒

The correct answer was : 189

The correct method is:  $81 + 34 + 74$

Next Question

## Coding

admin.php

Technique  
Class

Technique - Inheritance

```

1. <?php
2.     //child class of user class
3.     //oop - inheritance
4.     require "user.php";
5.     class admin extends user {
6.         function construct ($username,$password) {
7.             parent:: construct($username,$password);
8.         }
9.     }
10. ?>

```

adminSignUp.html

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Admin Sign Up</center></b></h1>
13.        </header>
14.        <br>
15.        <!--form to sign up to the system, makes sure that the page follows the
16.        standard design using the design API-->
17.        <form action="adminSignUp.php" style="margin:auto;width:60%" method="post">
18.            <label><b>Username</b></label>
19.            <input class="w3-input w3-
20.            border" type="text" name="username" required>
21.            <label><b>Password</b></label>
22.            <input class="w3-input w3-border" type="text" name="pw" required>
23.            <label><b>Re enter Password</b></label>
24.            <input class="w3-input w3-
25.            border" type="text" name = "confirmpw" required>
26.            <br>
27.            <input class="w3-button w3-block w3-dark-grey" type="submit">
28.            <br>
29.            <p>Please click <a href="register.php">here</a> to go back</p>
30.        </form>
31.    </body>
32. </html>

```

## adminSignUp.php

```

1. <?php
2. function genLoginID($pid){ //function to add p to the id, s can differentiate
   between the different types of people
3.     $id="a".($pid);
4.     return $id;
5. }
6. function password_strength_check($password, $min_len = 6, $max_len = 9, $req_digit
   = 1, $req_lower = 1, $req_upper = 1, $req_symbol = 1) {
7.     //function to check strength of password
8.     // Build regex string depending on requirements for the password
9.     // url ref - https://stackoverflow.com/questions/11873990/create-preg-match-
   for-password-validation-allowing
10.    $regex = '/^';
11.    if ($req_digit == 1) { $regex .= '(?=.*\d)'; }
   // Match at least 1 digit
12.    if ($req_lower == 1) { $regex .= '(?=.*[a-
13.    z])'; } // Match at least 1 lowercase letter
   if ($req_upper == 1) { $regex .= '(?=.*[A-
14.    Z])'; } // Match at least 1 uppercase letter
   if ($req_symbol == 1) { $regex .= '(?=.*[^\a-zA-
15.    Z\d])'; } // Match at least 1 character that is none of the above
16.    $regex .= '{' . $min_len . ',' . $max_len . '}$/';
17.    if(preg_match($regex, $password)) {
18.        return TRUE;
19.    } else {
20.        return FALSE;
21.    }
22. if ($_POST["pw"]==$_POST["confirmpw"]){ //only proceeds when passwords are the same
23.     if (password_strength_check($_POST["pw"]) == TRUE ){ //make sure the password
   has a good strength
24.         require "connection.php";
25.         require "admin.php";
26.         $conn = connection::getConn();
27.         //instantite parent object
28.         $h = new admin($_POST["username"],$_POST["pw"]);
29.         $username = $h->getUsername();
30.         $password = $h->getPassword();
31.         $phash = password_hash($password,PASSWORD_DEFAULT);
32.         if (password_verify($password, $phash)){
33.             //insert the details into table
34.             $sql = "INSERT INTO `admin` (`username`, `password`) VALUES ('$username
   ', '$phash')";
35.             if ($conn->query($sql) === TRUE) {
36.                 echo "WELCOME admin You have successfully created an account."<br>
   ">";
37.                 //get their ID so they can use for future ref
38.                 $sql = "SELECT MAX(adminID) as max FROM admin";
39.                 $result = $conn->query($sql);
40.                 if ($result->num_rows > 0) {
41.                     while($row=$result->fetch_assoc()){
42.                         $id = $row["max"];
43.                     }
44.                     $loginUN=$username;
45.                     $loginPW=$phash;
46.                     $loginID = "a".$id;

47. $sql = "INSERT INTO `user` (`id`, `username`, `password`, `sid`, `t id`,`pid`)
   VALUES ('$loginID', '$loginUN', '$loginPW', NULL, NULL, NULL)";
48.     if ($conn->query($sql) === TRUE) {
49.         echo "Please click "<a href="login.html">here</a>." to be re
   directed to the login page";
50.     }else{

```

**Technique – Aggregate SQL Function**

```

51.             echo "Connection error. Please try again later";
52.             echo "<br>";
53. echo "Please click ".<a href="http://al2020-
2.tanglincomputing.com/al2020-
13/coursework/adminSignUp.html">here</a>'. " to be redirected to the register page";
54.         }
55.     } else {
56.         echo "Connection error. Please try again later";
57.         echo "<br>";
58.         echo "Please click ".<a href="http://al2020-
2.tanglincomputing.com/al2020-
13/coursework/adminSignUp.html">here</a>'. " to be redirected to the register page";
59.     }
60. }
61. }else{
62.     echo "Password too weak. Must have a digit, a lowercase letter, uppercase l
etter, special character and between 6 to 9 characters.";
63.     echo "<br>";
64.     echo "Please click ".<a href="http://al2020-2.tanglincomputing.com/al2020-
13/coursework/adminSignUp.html">here</a>'. " to be redirected to the register page";
65. }
66.
67. }else{
68.     echo "different passwords entered";
69.     echo "<br>";
70.     echo "Please click ".<a href="http://al2020-2.tanglincomputing.com/al2020-
13/coursework/adminSignUp.html">here</a>'. " to be redirected to the register page";
71.
72. }
73.
74. $conn->close();
75. ?>

```

### buildClass.html

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Build Classroom</center></b></h1>
13.        </header>
14.        <br>
15.        <!--
form to allow teacher to make a connection with student, so that when homeworks are
completed by students, they can recieve emails about progress-->
16.        <form style="margin:auto;width:60%" action="buildClass.php" method="post">
17.            <label><b>Teacher ID </b></label>
18.            <input class="w3-input w3-
border" type="number" name="teacherID" required><br>
19.            <label><b>Student ID</b></label>
20.            <input class="w3-input w3-
border" type="number" name="studentID" required><br>

```

```

21.         <input type="submit" class="w3-button w3-block w3-dark-grey">
22.         <br>
23.         <br>
24.         <p>Please click <a href="homepage.php">here</a> to go back</p>
25.     </form>
26. </body>
27. </html>

```

buildClass.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Build Classroom</center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16. </html>
17. <?php
18.     $tid = $_POST['teacherID'];
19.     $sid = $_POST['studentID'];
20.     require "connection.php";
21.     $conn = connection::getConn();
22.     //gets the students ID
23.     $sql = "SELECT * FROM student WHERE studentID='$sid'";
24.     $result = $conn->query($sql);
25.     if ($result->num_rows > 0) {
26.         while($row = $result->fetch_assoc()) {
27.             $studentID = $row['studentID'];
28.         }
29.         $sql = "SELECT * FROM teacher WHERE teacherID='$tid'";
30.         $result = $conn->query($sql);
31.         if ($result->num_rows > 0) {
32.             while($row = $result->fetch_assoc()) {
33.                 $teacherID = $row['teacherID'];
34.             }
35.             //inserts this into a table so can be referred to in the future
36. $sql = "SELECT * FROM classroom WHERE teacherID='$teacherID' and studen tID =
37. '$studentID'";
38.     $result = $conn->query($sql);
39.     if ($result->num_rows > 0) {
40.         echo "This classroom already exists";
41.     }else{
42.         $sql = "INSERT INTO `classroom` (`teacherID`, `studentID`) VALUES (
43. '$teacherID', '$studentID')";
44.         if ($conn->query($sql) === TRUE) {
45.             echo "classroom created";
46.         }
47.     }
48. }else{
49.     echo "teacher does not exist";
50. }
51. }else{
52.     echo "student does not exist";
53. }
54. //get the teachers ID

```

**High Level Systems  
Overview** – Teacher add  
a student to their  
classroom



```

53.     echo "<br>";
54.     echo "Please click ".<a href="homepage.php">here</a>'. " to go back";
55. ?>

```

*buildFamily.html*

Coding Style – Well designed User Interface

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Build Family</center></b></h1>
13.    </header>
14.    <br>
15.    <!--
form so that parent can make connection with student, and so will recieve emails
about students homework attempt -->
16.    <form style="margin:auto;width:60%" action="buildFamily.php" method="post">
17.      <label><b>Parent ID </b></label>
18.      <input class="w3-input w3-
border" type="number" name="parentID" required><br>
19.      <label><b>Student ID</b></label>
20.      <input class="w3-input w3-
border" type="number" name="studentID" required><br>
21.      <input type="submit" class="w3-button w3-block w3-dark-grey">
22.      <br>
23.      <br>
24.      <p>Please click <a href="homepage.php">here</a> to go back</p>
25.    </form>
26.  </body>
27. </html>

```

*buildFamily.php*

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Build Family</center></b></h1>
13.    </header>
14.    <br>
15.  </body>
16. </html>
17. <?php //form to check parent can build family with selected studentID and theirID
18.   $pid = $_POST['parentID'];
19.   $sid = $_POST['studentID'];
20.   require "connection.php";

```

**Objective  
14**

 Parent build  
family

```

21. $conn = connection::getConn();
22. //get the student ID
23. $sql = "SELECT * FROM student WHERE studentID='$sid'";
24. $result = $conn->query($sql);
25. if ($result->num_rows > 0) {
26.     while($row = $result->fetch_assoc()) {
27.         $studentID = $row['studentID'];
28.     }
29.     //get the parent ID
30. $sql = "SELECT * FROM parent WHERE parentID='$pid'";
31. $result = $conn->query($sql);
32. if ($result->num_rows > 0) {
33.     while($row = $result->fetch_assoc()) {
34.         $parentID = $row['parentID'];
35.     }
36.     //stores student ID and parent ID
    into a table so can be used for future reference
37. $sql = "SELECT * FROM family WHERE parentID='$parentID' and studentID = '$stude
    ntID'";
38. $result = $conn->query($sql);
39. if ($result->num_rows > 0) {
40.     echo "This family already exists";
41. }else{
42.     $sql = "INSERT INTO `family` (`studentID`, `parentID`) VALUES ('$studentID'
    , '$parentID')";
43.     if ($conn->query($sql) === TRUE) {
44.         echo "family created";
45.     }
46. }
47. }else{
48.     echo "parent does not exist";
49. }
50.
51. }else{
52.     echo "student does not exist";
53. }
54. echo "<br>";

55. echo "Please click ".<a href="homepage.php">here</a>". to go back";
56. ?>
  
```

**High-Level System Overview-** parents  
add a student to their family

**Coding Style –** Good exception handling

changePasswordForm.php

```
1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
      t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Forgot Password</center></b></h1>
13.    </header>
14.    <br>
15.    <!--Form in case user has forgotten their password, they will be able to
      reset it-->
16.    <form style="margin:auto;width:60%" action="changePasswordFormAction.php" m
      ethod="post">
17.      <label><b>Username</b></label>
18.      <input class="w3-input w3-
        border" type="text" name="username" required>
19.      <label><b>Password</b></label>
```

```

20.         <input class="w3-input w3-border" type="text" name="pw" required>
21.         <label><b>Re enter password</b></label>
22.         <input class="w3-input w3-
border" type="text" name="confirmpw" required>
23.         <br>
24.         <input type="submit" class="w3-button w3-block w3-dark-grey">
25.         <br>
26.     </form>
27. </body>
28. </html>

```

#### changePasswordFormAction.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Forgot Password</center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16. </html>
17. <?php //Check to ensure that the password that the user wants to reset too is
possible (meets requirements and is same when entered twice)
18.     function password_strength_check($password, $min_len = 6, $max_len = 9, $req_di
git = 1, $req_lower = 1, $req_upper = 1, $req_symbol = 1) {
19.         // Build regex string depending on requirements for the password
20.         //function to ensure the password entered is strong enough
21.         // url ref - https://stackoverflow.com/questions/11873990/create-preg-
match-for-password-validation-allowing
22.         $regex = '/^';
23.         if ($req_digit == 1) { $regex .= '(?=.*\d)'; } // Match at least 1 digit
24.         if ($req_lower == 1) { $regex .= '(?=.*[a-
z])'; } // Match at least 1 lowercase letter
25.         if ($req_upper == 1) { $regex .= '(?=.*[A-
Z])'; } // Match at least 1 uppercase letter
26.         if ($req_symbol == 1) { $regex .= '(?=.*[^a-zA-
Z\d])'; } // Match at least 1 character that is none of the above
27.         $regex .= '{' . $min_len . ',' . $max_len . '}$/';
28.         if(preg_match($regex, $password)) {
29.             return TRUE;
30.         } else {
31.             return FALSE;
32.         }
33.     }
34.     require "connection.php";
35.     $conn = connection::getConn();
36.     $username = $_POST['username'];
37.     $sql = "SELECT * FROM `user` WHERE username = '$username' ";
38.     $result = $conn->query($sql);
39.     if ($result->num_rows > 0) {
40.         while($row=$result->fetch_assoc()){
41.             if ($_POST["pw"]==$_POST["confirmpw"]){ //ensures that the password
entered are the same
42.                 if (password_strength_check($_POST["pw"]) == TRUE ){
43.                     $password = $_POST["pw"];
44.                     $phash = password_hash($password,PASSWORD_DEFAULT);

```

```

45.         if (password_Verify($password, $phash)){
46.             $sql = "UPDATE `user` SET `password`='$phash' WHERE `userna
me`='$username'";
47.             if ($conn->query($sql) === TRUE) {
48.                 echo "Password updated";
49.             } else {
50.                 echo "Error, please try again later";
51.             }
52.             echo "Please click '<a href='\"login.html\">here</a>'." to g
o to login page";
53.         }
54.     }else{
55.         echo "passwords does not meet strength requirements. Minimum 6
and maximum 9 in length with at least 1 digit, 1 uppercase letter, 1 lowercase lett
er and a special letter";
56.     }
57. }else{
58.     echo "passwords are different";
59. }
60. }
61. }
62. else{
63.     echo "username does not exist";
64. }
65. $conn->close();
66. ?>

```

codeChange.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Code Change</center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16.</html>
17.<?php //Form to allow admin to change the code to access a particular topic
18.    require "connection.php";
19.    $conn = connection::getConn();
20.    //select the topic that you want to change the ID of
21.    $sql = "SELECT * FROM `topic` WHERE `topicID`>0";
22.    $result = $conn->query($sql);
23.    echo "Please select the topic you want to change the ID for";
24.    echo "<br>";
25.    echo "<html>";
26.    echo "<body>";
27.    echo "<br>";
28.    echo "<form style='width:30%' action='codeChanging.php' method='post'>";
29.    echo "<select class='w3-input w3-border' name='id'>";
30.    echo '<option value="" disabled selected>Please select the topic</option>';
31.    if ($result->num_rows > 0) {
32.        while($row = $result->fetch_assoc()) {
33.            unset($id, $name);
34.            $id = $row['codeID'];
35.            $name = $row['topicName'];

```

### High-Level Systems

#### Overview – to view

homework sheet access  
codes

```

36.         echo '<option value="'. $id. '">' . $name. '</option>';
37.     }
38. }
39. echo "</select>";
40. echo "<br>";
41. echo "<input type='text' name='code' maxlength='5' minlength='5'><br>";
42. echo "<br>";
43. echo "<input type='submit' value='Submit'>";
44. echo "</form>";
45. echo "</body>";
46. echo "</html>";
47. ?>

```

codeChanging.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Code Change</center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16. </html>

17. <?php //check to ensure that the code the admin wants to change to is possible to change to
18.     require "connection.php";
19.     $conn = connection::getConn();
20.     $id = $_POST['id'];
21.     $new = $_POST['code'];
22.     //identifies if code already exists
23.     $sql = "SELECT `topicID` FROM `topic` WHERE `codeID` = '$new'";
24.     $result = $conn->query($sql);
25.     if ($result->num_rows > 0) {
26.         echo "code already exists. please change to a unique code";
27.         echo "<br>";
28.         echo "Please click ". '<a href="codeChange.php">here</a>' . " to go back";
29.     }
30.     else {
31.         //changes code
32.         $sql = "SELECT `topicID` FROM `topic` WHERE `codeID` = '$id'";
33.         $result = $conn->query($sql);
34.         if ($result->num_rows > 0) {
35.             while($row = $result->fetch_assoc()) {
36.                 $topicID = $row['topicID'];
37.
38.                 $sql = "UPDATE `topic` SET `codeID` = '$new' WHERE `topicID` = '$topicID'";
39.                 $result = $conn->query($sql);
40.                 if ($result === TRUE) {
41.                     echo "The code has been changed to " . $new;
42.                     echo "<br>";
43.
44.                     echo "Please click ". '<a href="homepage.php">here</a>' . " to go back";
45.                 } else {

```

**High-Level Systems Overview** – to update homework access sheet code

### Objective 10

Admin change code for homework sheet

**SQL – Admin manage code**

Roshni

```
44.         echo "Connection error. Please try again later";  
45.         echo "<br>";
```

```

46.         echo "Please click ".<a href="codeChange.php">here</a>'.< a href="#">back</a>';
47.     }
48.
49.     }
50. }
51. }
52. ?>

```

connection.php

Technique - Class

```

1. <?php
2. class connection //this class allows pages to connect to the db
3. {
4.     public static function getConn()
5.     {
6.         $servername = "localhost";
7.         $username = "al2020-13";
8.         $password = "36*M5cxz";
9.         $dbname = "al2020-13";
10.        // Create connection
11.
12.        $conn = new mysqli($servername, $username, $password, $dbname);
13.        if ($conn->connect_error) {
14.            die("Connection failed: " . $conn->connect_error);
15.        }
16.        return $conn;
17.    }
18. }
19. ?>

```

High Level System  
Overview - Connecting  
to the database

contactUs.php

```

1. <?php
2. // the message to send to admin if someone wanted to ask a question
3. $msg = "Internet Searcher name - ".$_POST["Name"]."\n Internet Searcher email - 
".$_POST["Email"]."\n Internet Searcher message - ".$_POST["Message"];
4. // use wordwrap() if lines are longer than 70 characters
5. $msg = wordwrap($msg,70);
6. // send email
7. mail("cs.cswk@gmail.com","Internet Browser - ".$_POST["Name"],$msg);
8. ?>
9. <html>
10. <head>
11. <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
12. <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
13. <style>
14.     body {font-family: "Raleway", Arial, sans-serif}
15.     .w3-row img {margin-bottom: -8px}
16. </style>
17. </head>
18. <body style="background-color:#add8e6;">
19.     <p> Your information has been sent! Thank you! </p>
20. </body>
21. </html>

```



## Technique - Class

emailer.php

```

1.  <?php
2.      class email {
3.          //parent class used to send email to parents and teachers
4.          protected $emailAddress;
5.          protected $firstName;
6.          protected $surname;
7.          protected $childName;
8.          protected $topic;
9.          protected $score;
10.         protected $timeComplete;
11.
12.         function construct ($emailAddress,$firstName,$surname,$childName,$topic,$
score,$timeComplete) {
13.             $this->emailAddress = $emailAddress;
14.             $this->firstName = $firstName;
15.             $this->surname = $surname;
16.             $this->childName = $childName;
17.             $this->topic = $topic;
18.             $this->score=$score;
19.             $this->timeComplete = $timeComplete;
20.         }
21.         //function that sends the email
22.         function emailing(){
23.             $to = $emailAddress;
24.             $subject = "$childName"." homework attempt";
25.             $txt = "Hello! ".$childName." completed homework on the topic ".$topic.
". The student scored ".$score." and took a total of ".(int)($timeComplete/60).
minutes to do the homework";
26.             $message = wordwrap($txt,70,"\n");
27.             $headers = "From: homework@kidz.com" . "\r\n";
28.             mail($to,$subject,$message,$headers);
29.         }
30.         function getEmailAddress() {
31.             return $this->emailAddress;
32.         }
33.         function getFirstName() {
34.             return $this->firstName;
35.         }
36.         function getSurname() {
37.             return $this->surname;
38.         }
39.         function getChildName() {
40.             return $this->childName;
41.         }
42.         function getTopic() {
43.             return $this->topic;
44.         }
45.         function getScore() {
46.             return $this->score;
47.         }
48.         function getTimeComplete() {
49.             return $this->topicComplete;
50.         }
51.     }
52.  ?>

```

**Technique**  
Polymorphism

emailParent.php

Technique - Inheritance

**Technique**  
Class

```

1. <?php
2.     require "emailer.php";
3.     //child class that sends email to parent
4.     //oop - polymorphism
5.     class emailParent extends email {
6.         function construct ($emailAddress,$firstName,$surname,$childName,$topic,$
           score,$timeComplete) {
7.             parent:: construct($emailAddress,$firstName,$surname,$childName,$topic
           ,$score,$timeComplete);
8.         }
9.         function emailing(){
10.            $to = $this->emailAddress;
11.            $subject = "$this->childName"." homework attempt";
12.            $txt = "Hello! Your child, ".$this->childName." completed homework on
           the topic ".$this->topic.". The student scored ".$this->score." and took a total
           of ".$this->timeComplete." minutes to do the homework";
13.            $message = wordwrap($txt,70,"\n");
14.            $headers = "From: homework@kidz.com" . "\r\n";
15.            mail($to,$subject,$message,$headers);
16.        }
17.    }
18. ?>

```

**Objective 12**

Parent know how  
long child took to  
complete  
homework

**Technique**  
Polymorphism

emailTeacher.php

Technique - Class

Technique - Inheritance

```

1. <?php
2.     class emailTeacher extends email {
3.         //child class that sends email to teacher
4.         //oop - polymorphism
5.         function construct ($emailAddress,$firstName,$surname,$childName,$topic,$
           score,$timeComplete) {
6.             parent:: construct($emailAddress,$firstName,$surname,$childName,$topic
           ,$score,$timeComplete);
7.         }
8.         function emailing(){
9.            $to = $this->emailAddress;
10.            $subject = "$this->childName"." homework attempt";
11.            $txt = "Hello! Your student, ".$this->childName." completed homework
           on the topic ".$this->topic.". The student scored ".$this->score." and took a
           total of ".$this->timeComplete." minutes to do the homework";
12.            $message = wordwrap($txt,70,"\n");
13.            $headers = "From: homework@kidz.com" . "\r\n";
14.            mail($to,$subject,$message,$headers);
15.        }
16.    }
17. ?>

```

**Objective 38**

Teacher know  
how long  
student took to  
complete  
homework

**Technique**  
Polymorphism

feedback.php

## Coding Style – Minimal use of global variables

```

1. <?php
2. require "connection.php"; //page which gives the students feedback on the homework
   for that attempt
3. $conn = connection::getConn();
4. session_start();
5. $id = $_SESSION['userData'][2];
6. $plainID = substr($id, 1);
7. $sql = "SELECT * FROM `student` WHERE `studentID`=" . $plainID;
8. $result = $conn->query($sql);
9. if ($result->num_rows > 0) {
10.     while($row = $result->fetch_assoc()) {
11.         $firstName = $row["firstName"];
12.     }
13. }
14. ?>
15.
16. <html>
17.     <head>
18.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
19.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
   t" type="text/css">
20.         <style>
21.             body {font-family: "Raleway", Arial, sans-serif}
22.             .w3-row img {margin-bottom: -8px}
23.         </style>
24.     </head>
25.     <body style="background-color:#add8e6;">
26.         <header>
27.             <h1><b><center>Feedback Sheet for <?php echo $firstName;?></center></b>
   </h1>
28.         </header>
29.         <br>
30.     </body>
31. </html>
32. <?php
33.     session_start();
34.     $surname = "";
35.     $email = "";
36.     $firstNameParent = "";
37.     $firstNameStudent = "";
38.     $topic = $_SESSION['userScoreData'][0];

```

**High-Level Systems Overview –**  
students view feedback on  
homework attempt

## Coding Style – Minimal use of global variables

```

39. //calculate percent from total correct / total wrong (includes 2 for long quest
   ions - 1 mark for A and 1 mark for B)
40. $percent = (int)(($_SESSION['userScoreData'][2]/$_SESSION['userScoreData'][1])*
   100);
41. $score = $percent;
42. echo "The percentage you scored was: ".$percent."%";
43. echo "<br>";
44. //calculate diff in time - so total time to complete homework
45. $diff = $_SESSION['userScoreData'][7]->diff(new DateTime());
46. $minutes = ($diff->days * 24 * 60) + ($diff->h * 60) + $diff->i;
47. echo "Total time taken: ".$minutes." minutes";
48. echo "<br>";
49. $number = $_SESSION['userData'][2];
50. //get just the number part of the student ID
51. $identity = (int) filter_var($number, FILTER_SANITIZE_NUMBER_INT);
52. //get the details of the parent in order to send coherent email (first name,
   surname, email)

```

**Objective 18**

Student see  
overall **score**  
and comment  
(see next page)

**Objective 21**

Student know  
how long took  
to complete the  
homework

**Algorithm  
Design –**

**Coding  
Style –**  
Managed  
casting of  
types

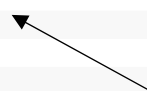
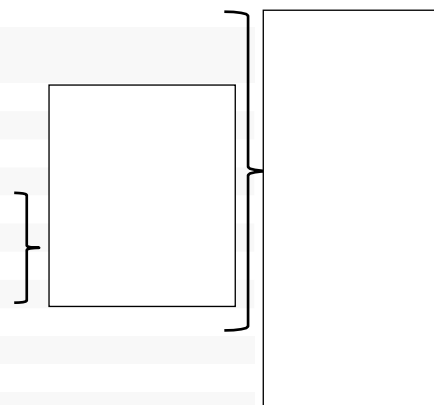
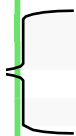
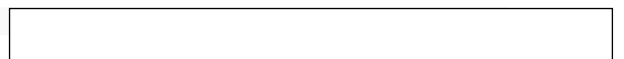
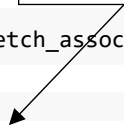
Calculate  
and print  
the overall  
percentage  
and time  
taken to  
complete

```

53. $parent="SELECT parent.firstName, parent.surname, parent.email FROM `parent`,`f
    amily` WHERE family.studentID = $identity AND family.parentID = parent.parentID";
54.     $result = $conn->query($parent);
55.     if ($result->num_rows>0) {
56.         while($row = $result->fetch_assoc()) {

```

**Technique** – Cross-table  
parametrised SQL



```

57.         $surname = $row["surname"];
58.         $email = $row["email"];
59.         $firstNameParent = $row["firstName"];
60.     }
61. } else {
62.     //echo "0 results";
63. }
64. //get the student name to send coherent email (first name)
65. $student="SELECT * FROM `student` WHERE student.studentID = '$identity'";
66. $result = $conn->query($student);
67. if ($result->num_rows>0) {
68.     while($row = $result->fetch_assoc()) {
69.         $firstNameStudent = $row['firstName'];
70.     }
71. } else {
72.     echo "Error. Please try again later";
73. }
74. //get the details of the teacher in order to send coherent email (first name, s
    urname, email)
75. $teacher="SELECT teacher.firstName, teacher.surname, teacher.email FROM `teache
    r`,`classroom` WHERE classroom.studentID = '$identity' AND classroom.teacherID = te
    acher.teacherID";
76. $result = $conn->query($teacher);
77. if ($result->num_rows>0) {
78.     while($row = $result->fetch_assoc()) {
79.         $surnameTeacher = $row["surname"];
80.         $emailTeacher = $row["email"];
81.         $firstNameTeacher = $row["firstName"];
82.     }
83. } else {
84.     echo "Error. Please try again later";
85. }
86. $dateToday = date("Y-m-d");

87. //insert into a table, the important information about the test
88. $sql = "INSERT INTO `test`('studentID','studentScore','Date') VALUES
    ('$identity','$score','$dateToday')";
89. if ($conn->query($sql) == TRUE) {
90. } else {
91.     echo "Connection error";
92. }
93. //call class required to send email to parent
94. require "emailParent.php";
95. //call class required to send email to teacher
96. require "emailTeacher.php";
97. //instantiate class to send email to parent
98. $p = new emailParent($email,$firstNameParent,$surname,$firstNameStudent,$topic,
    $score,$minutes);
99. //send email to parent
100. $email = $p->emailing();
101. //instantiate class to send email to teacher
102. $t = new emailTeacher($emailTeacher,$firstNameTeacher,$surnameTeacher,
    $firstNameStudent,$topic,$score,$minutes);
103. //send email to teacher
104. $email = $t->emailing();
105. // to give general feedback about the test
106. if ($score>89){
107.     echo "Well done!! Keep it up!";
108. }
109.
110.
111.
112.
113.
114.
115.
116.
    }
    e
    l
    s
    e
    i
    f

```

Technique – Cross-table  
parametrised SQL

### Objective 1

Database  
record  
progress of  
each student

### Objective 11

Send email to  
parent when  
homework  
complete

### Objective 36

Send email to  
teacher when  
homework  
complete

### Objective 18

Student see  
overall score  
(see previous  
page) and  
comment

### Algorithm Design –

send  
email to  
the parent  
and the  
teacher  
the  
students  
score,  
topic  
name,  
time

Roshni

```
(
    ){ echo "Could be better!";
$else {
s    echo "Oh dear, try harder next week!";
c    }
o    //get the test ID
r    $testID ="SELECT * FROM test ORDER BY testID DESC LIMIT 1";
e
>
6
9
a
n
d
$
s
c
o
r
e
<
9
0
)
{
e
c
h
o
"
G
o
o
d
j
o
b
!
!
"
;
}
e
l
s
e
i
f
(
$
s
c
o
r
e
>
4
9
a
n
d
$
s
c
o
r
e
<
7
0
```

## Algorithm

### Design –

General  
feedback about  
the attempt  
based on overall  
score

```

117.         $result = $conn->query($testID);
118.         if ($conn->query($testID) == TRUE) {
119.             if ($result->num_rows > 0) {
120.                 while($row = $result->fetch_assoc()) {
121.                     $tid = $row["testID"];
122.                 } }else {
123.                     echo "Connection error";
124.                 }
125.                 //insert into table the test ID and topic id, for future reference
126. $sql = "INSERT INTO `testAndTopic` (`testID`, `topicID`) VALUES ('$tid',
127. '{$_SESSION['userScoreData']}[4]}')";
128.         if ($conn->query($sql) == TRUE) {
129.             } else {
130.                 echo "Connection error";
131.             }
132.             echo "<br>";
133.             echo "<b>Scores for each question</b>";
134.             echo "<table style='width:50%'>";
135.             echo "<tr>";
136.             echo "<th>Question number</th>";
137.             echo "<th>Score achieved</th>";
138.             echo "<th>Maximum possible score</th>";
139.             echo "</tr>";
140.             $length = sizeof($_SESSION['userScoreData'][9]);
141.             for ($x = 0; $x <= ($length-1); $x++) {
142.                 if ($x<7){
143.                     $val = $x+1;
144.                 }else if ($x % 2 ==1){
145.                     if ($x==7){
146.                         $val = ($x+1)."a";
147.                     }
148.                     if ($x==9){
149.                         $val = ($x)."a";
150.                     }
151.                     if ($x==11){
152.                         $val = ($x-1)."a";
153.                     }
154.                 }else{
155.                     if ($x==8){
156.                         $val = ($x)."b";
157.                     }
158.                     if ($x==10){
159.                         $val = ($x-1)."b";
160.                     }
161.                     if ($x==12){
162.                         $val = ($x-2)."b";
163.                     }
164.                 }
165.                 echo "<tr>";
166.                 echo "<td>$val</td>";
167.                 echo "<td>{$_SESSION['userScoreData']}[9][$x]}</td>";
168.                 echo "<td>1</td>";
169.                 echo "</tr>";

```

**Objective 19**

Student see  
score for each  
question

**Algorithm**

**Design –**  
Print question  
number and the  
score they got

```

170.         echo "</table>";
171.         echo "<br>";
172.         echo "<b>Percents for each type of question</b>";
173.         echo "<table >";
174.         echo "<tr>";
175.         echo "<th>Type</th>";
176.         echo "<th>Percent achieved</th>";
177.         echo "</tr>";
178.         echo "<tr>";
179.         $mcq = (int)(((($_SESSION['userScoreData'])[9][0]+$_SESSION['userScoreData']
[9][1]+$_SESSION['userScoreData'] [9][2])/3)*100);

```

**Objective 20**

Student see  
percent for each  
type of question

**Algorithm**

**Design –**  
Calculate

Roshni

180.

echo " <td>MCQ</td>";

and print  
MCQ



```

181.         echo "<td>$mcq</td>";
182.         echo "    </tr>";
183.         echo "<tr>";
184.         $short = (int)((($_SESSION['userScoreData'][9][3]+
$_SESSION['userScoreData'][9][4]+$_SESSION['userScoreData'][9][5]+$_SESSION['userScoreData'][9][6])/4)*100);
185.         echo "    <td>Short Answer</td>";
186.         echo "<td>$short</td>";
187.         echo "    </tr>";
188.         echo "<tr>";
189.         $long = (int)((($_SESSION['userScoreData'][9][7]+
$_SESSION['userScoreData'][9][8]+$_SESSION['userScoreData'][9][9]+
$_SESSION['userScoreData'][9][10]+$_SESSION['userScoreData'][9][11]+
$_SESSION['userScoreData'][9][12])/6)*100);
190.         echo "    <td>Long Answer</td>";
191.         echo "<td>$long</td>";
192.         echo "    </tr>";
193.         echo "</table>";
194.         echo "<br>";
195.         echo "Please click ".<a href="logout.php">here</a>." to log out";
196.     ?>

```

**Algorithm Design –**  
Calculate and print short questions average

**Algorithm Design –**  
Calculate and print long question average

*forgotPassword.php*

```

1. <html> <!--Form for user to fill in if forgotten password-->
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Forgot Password</center></b></h1>
13.        </header>
14.        <br>
15.        <!--forgot passwords-->
16.        <form action="forgotPasswordCheck.php" style="margin:auto;width:50%" method="post">
17.            <label><b>Username</b></label>
18.            <input type="text" name="username" class="w3-input w3-border" ><br>
19.            <input type="submit" class="w3-button w3-block w3-dark-grey">
20.            <br>
21.            <br>
22.            <p>Please click <a href="login.html">here</a> to go back</p>
23.        </form>
24.    </body>
25. </html>

```

### forgotPasswordCheck.php

```
1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         t" type="text/css">
6.         <style>
7.             body {font-family: "Raleway", Arial, sans-serif}
8.             .w3-row img {margin-bottom: -8px}
9.         </style>
10.    </head>
11.    <body style="background-color:#add8e6;">
12. </body>
13. </html>
14. <?php
15.     require "connection.php";
16.     $conn = connection::getConn();
17.     $username = $_POST['username'];
18.     //check to see if user exists in system. If exist, they can change password, if
19.     not, go back to log in page
20.     $sql = "SELECT * FROM `user` WHERE username = '$username' ";
21.     $result = $conn->query($sql);
22.     if ($result->num_rows > 0) {
23.         while($row=$result->fetch_assoc()){
24.             require "changePasswordForm.php";
25.         }
26.     }else{
27.         echo "username does not exist";
28.         echo "<br>";
29.         echo "Please click ".<a href="login.html">here</a>". to go back";
30.     }
31.     $conn->close();
32. ?>
```

### form.html

```
1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         t" type="text/css">
6.         <style>
7.             body {font-family: "Raleway", Arial, sans-serif}
8.             .w3-row img {margin-bottom: -8px}
9.         </style>
10.    </head>
11.    <body style="background-color:#add8e6;">
12.        <header>
13.            <h1><b><center>Manage Family</center></b></h1>
14.        </header>
15.        <br>
16.        <!-- form with a drop down list which will allow user to select topic they
17.        want to see the progress on and questions they were given to answer -->
18.        <p style="text-align: center">Please select the topic you want to view the student's progress of
19.        <form action="selectTest.php" style="margin:auto;width:40%" method="post">
20.        <select class="w3-input w3-border" name="topicID">
21.            <option value="" disabled selected>Please select the topic</option>
22.            <option value="1">Addition</option>
23.            <option value="2">Subtraction</option>
24.            <option value="3">Numbers</option>
25.            <option value="4">Multiplication</option>
26.            <option value="5">Division</option>
```

```

25.         <option value="6">Mixed Operations</option>
26.         <option value="7">Sequences</option>
27.         <option value="8">Decimal</option>
28.         <option value="9">Ratios</option>
29.         <option value="10">Percentage</option>
30.         <option value="11">Time</option>
31.         <option value="12">Round and Estimate</option>
32.         <option value="13">Area of shapes</option>
33.         <option value="14">Perimeter of shapes</option>
34.         <option value="15">Probability</option>
35.         <option value="16">Factorisation</option>
36.         <option value="17">Substitution</option>
37.         <option value="18">Conversion of measurements</option>
38.         <option value="19">Rates and Speed</option>
39.     </select>
40.     <input class="w3-button w3-block w3-dark-grey" type="submit">
41. </form>
42. </body>
43. </html>

```

home.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="style.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <div class="w3-content" style="max-width:1500px">
12.            <!-- Header -->
13.            <header class="w3-container w3-xlarge w3-padding-24">
14.                <a href="#about" class="w3-left w3-button w3-white">MATHS @ KIDZ</a>
15.                <a href="login.html" class="w3-right w3-button w3-white">login / sign up</a>
16.            </header>
17.            <!-- Photo Grid Of Pictures -->
18.            <div class="w3-row">
19.                <div class="w3-half">
20.                    
21.                    
22.                    
23.                </div>
24.                <div class="w3-half">
25.                    
26.                    
27.                </div>
28.            </div>
29.            <!-- End Page Content -->
30.        </div>
31.        <!-- About Section -->
32.        <footer class="w3-light-grey w3-padding-64 w3-center" id="about">

```

```

33.         <h2>About</h2>
34.         <form style="margin:auto;width:60%" action="contactUs.php" target="_bla
nk" method="post">
35.             <p>kidz maths homework for lit kids</p>
36.             <p class="w3-large w3-text-
pink">Do not hesitate to contact us!</p>
37.             <div class="w3-section">
38.                 <label><b>Name</b></label>
39.                 <input class="w3-input w3-
border" type="text" required name="Name">
40.             </div>
41.             <div class="w3-section">
42.                 <label><b>Email</b></label>
43.                 <input class="w3-input w3-
border" type="text" required name="Email">
44.             </div>
45.             <div class="w3-section">
46.                 <label><b>Message</b></label>
47.                 <input class="w3-input w3-border" required name="Message">
48.             </div>
49.             <button type="submit" class="w3-button w3-block w3-dark-
grey">Send</button>
50.         </form>
51.         <br>
52.         <p>Powered by <a href="https://www.w3schools.com/w3css/default.asp" tar
get="_blank" class="w3-hover-text-green">w3.css</a></p>
53.     </footer>
54. </body>
55. </html>

```

homepage.php

```

1. <?php //display pages user can access based on type of user
2.     require "connection.php";
3.     $conn = connection::getConn();
4.     session_start();
5.     $id = $_SESSION['userData'][2];
6.     if (strpos($id, 's') !== false) { //url ref - https://stackoverflow.com/questio
ns/4366730/how-do-i-check-if-a-string-contains-a-specific-word
7.         $plainID = substr($id, 1);
8.         $sql = "SELECT * FROM `student` WHERE `studentID`=".$plainID;
9.         $result = $conn->query($sql);
10.        if ($result->num_rows > 0) {
11.            while($row = $result->fetch_assoc()) {
12.                $firstName = $row["firstName"];
13.            }
14.        }
15.    } else if (strpos($id, 'p') !== false){
16.        $plainID = substr($id, 1);
17.        $sql = "SELECT * FROM `parent` WHERE `parentID`=".$plainID;
18.        $result = $conn->query($sql);
19.        if ($result->num_rows > 0) {
20.            while($row = $result->fetch_assoc()) {
21.                $firstName = $row["firstName"];
22.            }
23.        }
24.    } else if (strpos($id, 't') !== false){
25.        $plainID = substr($id, 1);
26.        $sql = "SELECT * FROM `teacher` WHERE `teacherID`=".$plainID;
27.        $result = $conn->query($sql);
28.        if ($result->num_rows > 0) {
29.            while($row = $result->fetch_assoc()) {
30.                $firstName = $row["firstName"];
31.            }
32.        }

```

Coding Style – Good use of global variables

```

33.     }
34. }else if (strpos($id, 'a') !== false) {
35.     $firstName = "ADMIN ";
36. }
37. ?>
38. <html>
39.     <head>
40.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
41.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="style
sheet" type="text/css">
42.         <style>
43.             body {font-family: "Raleway", Arial, sans-serif}
44.             .w3-row img {margin-bottom: -8px}
45.         </style>
46.     </head>
47.     <body style="background-color:#add8e6;">
48.         <header><i><b><?php echo "hello ".$firstName;?></b></i></header>
49.     </body>
50. </html>
51. <?php
52. session_start();
53. //ensures there is an ID that is being passed
54. if(!isset($_SESSION['id'])||(trim($_SESSION['id'])=='')){
55.     $id = $_SESSION['userData'][2];
56.     //outputs relevant message and gives them access to relevant things dependi
ng on access
57.     if (strpos($id, 's') !== false) { //url ref - https://stackoverflow.com/que
stions/4366730/how-do-i-check-if-a-string-contains-a-specific-word
58.         echo "<br>";
59.         echo "Please click '<a href="questionSet.html">here</a>'." to get acc
ess to questions";
60.         $plainID = substr($id, 1);
61.         $sql = "SELECT * FROM `student` WHERE `studentID`=".$plainID;
62.         $result = $conn->query($sql);
63.         if ($result->num_rows > 0) {
64.             while($row = $result->fetch_assoc()) {
65.                 $firstName = $row["firstName"];
66.             }
67.         }
68.     } else if (strpos($id, 'p') !== false){
69.         echo "<br>";
70.         echo "Please click '<a href="buildFamily.html">here</a>'." to build f
amily";
71.         echo "<br>";
72.         $idPrint = "?id=$id";
73.         $link = "selectStudentFamilyView.php".$idPrint;
74.         $href = "<a href=$link> here</a>";
75.         echo "Please click ".$href." to access student data";
76.         echo "<br>";
77.         $idPrint = "?id=$id";
78.         $link = "seeFamily.php".$idPrint;
79.         $href = "<a href=$link> here</a>";
80.         echo "Please click ".$href." to view family";
81.         echo "<br>";
82.         echo "Please click '<a href="logout.php">here</a>'." to go log out";
83.         $plainID = substr($id, 1);
84.         $sql = "SELECT * FROM `parent` WHERE `parentID`=".$plainID;
85.         $result = $conn->query($sql);
86.         if ($result->num_rows > 0) {
87.             while($row = $result->fetch_assoc()) {
88.                 $firstName = $row["firstName"];
89.             }
90.         }
91.     }else if (strpos($id, 't') !== false){
92.         echo "<br>";

```

```

93.      echo "Please click '.'<a href='\"buildClass.html\">here</a>'." to build cl
    assroom";
94.      echo "<br>";
95.      echo "Please click '.'<a href='\"select.html\">here</a>'." to access stude
    nt data";
96.      echo "<br>";
97.      $idPrint = "?id=$id";
98.      $link = "seeClassroom.php".$idPrint;
99.      $href = "<a href=$link> here</a>";
100.         echo "Please click ".$href." to view classroom";
101.         echo "<br>";
102.         echo "Please click '.'<a href='\"logout.php\">here</a>'." to go log
    out";
103.         $plainID = substr($id, 1);
104.         $sql = "SELECT * FROM `teacher` WHERE `teacherID`=".$plainID;
105.         $result = $conn->query($sql);
106.         if ($result->num_rows > 0) {
107.             while($row = $result->fetch_assoc()) {
108.                 $firstName = $row["firstName"];
109.             }
110.         }
111.         }else if (strpos($id, 'a') !== false) {
112.             echo 'You are admin';
113.             echo "<br>";
114.             echo "Please click '.'<a href='\"seeAllClassrooms.php\">here</a>'."
    to view and manage all classrooms";
115.             echo "<br>";
116.             echo "Please click '.'<a href='\"seeAllFamilies.php\">here</a>'." t
    o view and manage all families";
117.             echo "<br>";
118.             echo "Please click '.'<a href='\"codeChange.php\">here</a>'." to vi
    ew and manage the codes";
119.             echo "<br>";
120.             echo "Please click '.'<a href='\"logout.php\">here</a>'." to go log
    out";
121.         }
122.     }
123.     else{
124.         echo "Incorrect username, password, ID or code";
125.         sleep(10);
126.         //destroys all session variables and redirects them to login page
127.         session_unset();
128.         session_destroy();
129.         header('Location:login.html');
130.     }
131. }>

```

login.html

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <center><p>Maths @ Kidz</p></center>
13.    </header> <!--login form (where user inputs stuff) to access system-->
14.    <form style="margin:auto;width:50%" action="loginCheck.php" method="post">
15.      <label><b>Username</b></label>
16.      <input class="w3-input w3-border" type="text" name="username" required width=50%><br>
17.      <label><b>Password</b></label>
18.      <input class="w3-input w3-border" type="password" name="password" required width=50%><br>
19.      <input class="w3-button w3-block w3-dark-grey" type="submit" value="Submit">
20.      <p>If you don't have an account, register <a href="register.php">here</a></p>
21.      <p>If you forgot your password, please click <a href="forgotPassword.php">here</a></p>
22.    </form>
23.  </body>
24. </html>

```

### Objective 17

Student only has to enter **username**, **password** and code to access sheet (See below page 348)

loginCheck.php

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.    </header>
13.  </body>
14. </html>
15. <?php //check log in details from details passed from login.html page
16.   require "user.php";
17.   require "connection.php";
18.   $conn = connection::getConn();
19.   session_start();
20.   //instantiate user object
21.   $h = new user($_POST["username"],$_POST["password"]);
22.   $username = $h->getUsername();
23.   $password = $h->getPassword();
24.   //check if user exists, else redirects them back to the login page
25.   $sql = "SELECT `id`,`username`,`password`,`sid`,`tid`,`pid` FROM `user` WHERE username = '$username' ";
26.   $result = $conn->query($sql);
27.   if ($result->num_rows > 0) {

```

**High-Level Systems Overview** – checking that the user exists in the system

**Password Hash Function** – verify the password entered is the same as the hash stored

```

28.     while($row=$result->fetch_assoc()){
29.         if (password_verify($password, $row["password"])){
30.
31.             $_SESSION['userData']=array($row["username"],$row["password"],$row[
32. "id"]);
33.             header('Location:homepage.php?firstName='.$firstName);
34.
35.         } else{
36.             echo "<br>";
37.             echo "The details you have entered are incorrect";
38.             echo "<br>";
39.             echo "Please click ".<a href="login.html">here</a>". to try again
40. ";
41.         }
42.     } else{
43.         echo "Wrong Details";
44.         echo "Please click ".<a href="login.html">here</a>". to try again";
45.     }
46. }
47. $conn->close();
48. ?>

```

**Data structure - Implode**

**Coding Style – Minimal use of global variables**

*logout.php*

```

1. <html> <!--form to log out user by destroying all sessions-->
2. <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5. t" type="text/css">
6.     <style>
7.         body {font-family: "Raleway", Arial, sans-serif}
8.         .w3-row img {margin-bottom: -8px}
9.     </style>
10. </head>
11. <body style="background-color:#add8e6;">
12.     <header>
13.         <br>
14.         <br>
15.         <h1><b><center>You have successfully logged out </center></b></h1>
16.         <br>
17.         <br>
18.         <h1><b><center><i>Please click <a href="home.php">here</a> to enter the
19. system again</i></center></b></h1>
20.     </header>
21.     <br>
22. </body>
23. <?php
24.     session_start();
25.     session_unset();
26.     session_destroy();
27. ?>

```

*manageClassroom.php*

```

1. <html>
2. <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5. t" type="text/css">
6.     <style>
7.         body {font-family: "Raleway", Arial, sans-serif}
8.         .w3-row img {margin-bottom: -8px}
9.     </style>

```



```
9.     </head>
10.    <body style="background-color:#add8e6;">
```

```

11.         <header>
12.             <h1><b><center>Manage Classroom </center></b></h1>
13.         </header>
14.         <br>
15.     </body>
16. </html>
17. <?php    //page to allow admin to see all classroom links that exist and write one
           they want to delete
18.     require "connection.php";
19.     $conn = connection::getConn();
20.     $dataS = [];
21.     //displays all students and teachers
22.     $sql = "select student.firstName, student.surname, classroom.classroomID from c
           lassroom,teacher,student where classroom.teacherID = teacher.teacherID and classroo
           m.studentID = student.studentID";
23.     $result = $conn->query($sql);
24.     $count = 0;
25.     if ($result->num_rows > 0) {
26.         while($row = $result->fetch_assoc()) {
27.             $count = $count + 1;
28.             $dataS[]=
29.                 'studentFName' =>$row["firstName"],
30.                 'studentSName' => $row["surname"],
31.                 'id' => $row['classroomID']
32.             ];
33.         }
34.     } else {
35.         echo "0 families";
36.     }
37.     $dataT=[];
38.     $sql = "select teacher.firstName, teacher.surname from classroom,teacher,studen
           t where classroom.teacherID = teacher.teacherID and classroom.studentID = student.s
           tudentID";
39.     $result = $conn->query($sql);
40.     $count = 0;
41.     if ($result->num_rows > 0) {
42.         while($row = $result->fetch_assoc()) {
43.             $count = $count + 1;
44.             $dataT[]=
45.                 'teacherFName' =>$row["firstName"],
46.                 'teacherSName' => $row["surname"]
47.             ];
48.         }
49.         echo "<table border=1 frame=box rules=all>";
50.         echo "<tr><td width=20%>" . "<b>Student F Name</b>" . "</td><td width=20%>"
           . "<b>Student S Name</b>" . "</td><td width=20%>" . "<b>Teacher F Name</b>" . "</t
           d><td width=20%>" . "<b>Teacher S Name</b>" . "</td><td width=20%>" . "<b>ID</b>" .
           "</td></tr>";
51.         for ($x = 0; $x <= $count; $x++) {
52.             echo "<tr><td width=20%>" . $dataS[$x]['studentFName'] . "</td><td width=
           20%>" . $dataS[$x]['studentSName'] . "</td><td width=20%>" . $dataT[$x]['teacherF
           Name'] . "</td><td width=20%>" . $dataT[$x]['teacherSName'] . "</td><td width=20%>"
           . $dataS[$x]['id'] . "</td></tr>";
53.         }
54.         echo "</table>";
55.         echo "<br>";
56.     }
57. ?>
58. <html>
59.     <head>
60.     </head>
61.     <body>
62.         <form action="manageClassroomDo.php" style="width:30%" method="post">
63.             <br>
64.             <label><b>Delete Classroom ID</b></label>
65.             <input class="w3-input w3-border" type="number" name="cid"><br>

```

**High-Level Systems Overview** – for admin to select a classroomID they want to delete

```

66.         <input class="w3-button w3-block w3-dark-
        grey" type="submit" value="Submit">
67.     </form>
68. </body>
69. </html>

```

*manageClassroomDo.php*

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
        t" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Manage Classroom</center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16. </html>

```

**High-Level Systems Overview** – To remove classroomID selected by admin from the database

```

17. <?php //allow admin to delete a classroom link stated in manageClassroom form
18.     require "connection.php";
19.     $conn = connection::getConn();
20.     $id = $_POST['cid'];
21.     //delete from db with this cid
22.     $sql = "DELETE from classroom where classroom.classroomID=$id";
23.     //$result = $conn->query($sql);
24.     if ($conn->query($sql) === TRUE) {
25.         echo "Error. Classroom ID does not exist";
26.     } else {
27.         // echo "Error. Classroom ID does not exist";
28.     }
29.     echo "<br>";
30.     echo "Please click ".<a href="seeAllClassrooms.php" >here</a>." to go back";
31.
32. ?>

```

### Objective 7

Admin able to delete link between student and teacher

**SQL – Admin manage classroom links**

*manageFamily.php*

```
1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
    t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Manage Family </center></b></h1>
13.    </header>
14.    <br>
15.  </body>
16. </html>
17. <?php //page to allow admin to see all family links and delete any if need to
18.   require "connection.php";
```

```

19.     $conn = connection::getConn();
20.     $dataS = [];
21.     //selects and students and their parent
22.     $sql = "select student.firstName, student.surname, family.familyID from family,
parent,student where family.parentID = parent.parentID and family.studentID = stude
nt.studentID";
23.     $result = $conn->query($sql);
24.     $count = 0;
25.     if ($result->num_rows > 0) {
26.         while($row = $result->fetch_assoc()) {
27.             $count = $count + 1;
28.             $dataS[]=[
29.                 'studentFName' =>$row["firstName"],
30.                 'studentSName' => $row["surname"],
31.                 'id' => $row["familyID"] 32.
            ];
33.         }
34.     } else {
35.         echo "0 families";
36.     }
37.     $dataP=[];
38.     $sql = "select parent.firstName, parent.surname from family,parent,student wher
e family.parentID = parent.parentID and family.studentID = student.studentID";
39.     $result = $conn->query($sql);
40.     $count = 0;
41.     if ($result->num_rows > 0) {
42.         while($row = $result->fetch_assoc()) {
43.             $count = $count + 1;
44.             $dataP[]=[
45.                 'parentFName' =>$row["firstName"],
46.                 'parentSName' => $row["surname"] 47.
            ];
48.         }
49.         echo "<table border=1 frame=box rules=all>";
50.         echo "<tr><td width=20%>" . "<b>Student F Name</b>" . "</td><td width=20%>"
. "<b>Student S Name</b>" . "</td><td width=20%>" . "<b>Parent F Name</b>" . "</td>
<td width=20%>" . "<b>Parent S Name</b>" . "</td><td width=20%>" . "<b>ID</b>" . "
</td></tr>";
51.         for ($x = 0; $x <= $count; $x++) {
52.             echo "<tr><td width=20%>" . $dataS[$x]['studentFName'] . "</td><td widt
h=20%>" . $dataS[$x]['studentSName'] . "</td><td width=20%>" . $dataP[$x]['parentFN
ame'] . "</td><td width=20%>" . $dataP[$x]['parentSName'] . "</td><td width=20%>" .
$dataS[$x]['id'] . "</td></tr>";
53.         }
54.         echo "</table>";
55.     }
56. }
57. echo "<br>";
58. echo "Please click ".<a href="homepage.php" >here</a>'. " to go back";
59. echo "<br>";
60. ?>
61. <html>
62.     <head>
63.     </head>
64.     <body>

65.         <form action="manageFamilyDo.php" style="width:30%" method="post">
66.             <br>
67.             <br>
68.             <label><b>Delete Family ID</b></label>
69.             <input class="w3-input w3-border" type="number" name="fid"><br>
70.             <input class="w3-button w3-block w3-dark-
grey" type="submit" value="Submit">
71.         </form>
72.     </body>
73. </html>

```

**High-Level Systems  
Overview** – for admin to  
select a family ID to delete

## manageFamilyDo.php

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Manage Family</center></b></h1>
13.    </header>
14.    <br>
15.  </body>
16. </html>
17. <?php //deletes family id selected in manageFamily form
18.   require "connection.php";
19.   $conn = connection::getConn();
20.   $id = $_POST['fid'];
21.   //delete from db with this family id
22.   $sql = "DELETE FROM family WHERE family.familyID=$id";
23.   // $result = $conn->query($sql);
24.   if ($conn->query($sql) === TRUE) {
25.     echo "Error. Family ID does not exist";
26.     echo "<br>";
27.   } else {
28.     $sql = "SELECT *FROM `family` WHERE `familyID`=$id";
29.     $result = $conn->query($sql);
30.     if ($result->num_rows > 0) {
31.       // output data of each row
32.       while($row = $result->fetch_assoc()) {
33.         $studentID = $row["studentID"];
34.         $parentID = $row["parentID"];
35.       }
36.     }

37.     $sql = "DELETE FROM student WHERE student.studentID=$studentID";
38.     if ($conn->query($sql) === TRUE) {
39.       echo "Student successfully removed from the system";
40.       echo "<br>";
41.     }
42.     $sql = "DELETE FROM parent WHERE parent.parentID=$parentID";
43.     if ($conn->query($sql) === TRUE) {
44.       echo "Parent successfully removed from the system";
45.       echo "<br>";
46.     }
47.   }
48.   echo "Please click ".<a href="seeAllFamilies.php" >here</a>". to go back";
49. ?>

```

**Objective 9**

Admin able to delete link between student and parent

**SQL – Admin**  
manage family links

**High-Level Systems Overview** – for familyID selected by admin to be removed from table

parents.php

Technique - Inheritance

Technique  
Class

```

1. <?php
2. //child class of user class
3. //oop - inheritance
4. require "user.php";
5. class parents extends user {
6.     private $firstName;
7.     private $surname;
8.     private $email;
9.     private $id;
10.    function construct ($username,$password,$firstName,$surname,$email) {
11.        parent:: construct($username,$password);
12.        $this->firstName = $firstName;
13.        $this->surname = $surname;
14.        $this->email=$email;
15.    }
16.    function getFirstName() {
17.        return $this->firstName;
18.    }
19.    function getSurname() {
20.        return $this->surname;
21.    }
22.    function getEmail() {
23.        return $this->email;
24.    }
25. }
26. ?>

```

questionSet.html

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head> <!--Form for student to enter homework code to access the sheets-->
10.    <body style="background-color:#add8e6;">
11.        <form style="margin:auto;width:40%" action="questionSet.php" method="post">
12.            <br>
13.            <label><center><b>Homework code</b></center></label>
14.            <br>
15.            <input class="w3-input w3-
16.            border" type="text" name="code" required><br>
17.            <input type="submit" class="w3-button w3-block w3-dark-grey">
18.        </form>
19.    </body>
20. </html>

```

**Objective 17**

Student access  
system with only  
username,  
password (see  
above page 348)  
and code

## questionSet.php

```

1. <?php
2.     session_start();
3.     $codeEnter = $_POST["code"];
4.     require "connection.php";
5.     $conn = connection::getConn();

6.     //checks if user entered a valid code, then sets all the necessary information
    like topicName, topicID, totalWrong so it can all be passed and used when necessary
    , else allows user to enter again
7.     $sql = "SELECT * FROM `topic` WHERE codeID='$codeEnter'";
8.     $result = $conn->query($sql);
9.     if ($result->num_rows > 0) {
10.        while($row = $result->fetch_assoc()) {
11.            $topicName = $row["sheetName"];
12.            $topicID = $row["topicID"];
13.            $topicFullName = $row["topicName"];

14.            $questionNumber = 0; //score calc (long has 2)
15.            $totalCorrect = 0;

16.            $totalWrong = 0;
17.            $time = new DateTime(); //initial time

18.            $questionCount = 0; //user see question number they are on(out of 10)
19.            $level = 2;
20.            $previous = array();
21.            $_SESSION['userScoreData']=array($topicName,$questionNumber,$totalCorrect,$totalWrong,$topicID,$topicFullName,$questionCount,$time,$level,$previous);
22.            $goTo = "sort.php";
23.            header($goTo);
24.        }
25.    } else {
26.        echo "Invalid code. Please try again";
27.        sleep(10);
28.        $location = "Location:questionSet.html";
29.        header($location);
30.    }
31. ?>

```

**High-Level Systems Overview –**  
students entering the homework  
system for a particular topic

**Coding Style – Consistent Style Throughout**

**Coding Style**

**Meaningful Identifier  
Names**

**Coding Style – Self Documenting Code**

**Coding Style – Managed casting of types**

**Data structure - Implode**

**Coding Style**

File paths  
parametrised

## register.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;"><center>
11.        <br>
12.        <br>
13.        <!-- so that user can sign up under the correct type -->
14.        <p>Click<a href="registerStudentLink.php">Here</a> if you are a student</p>
15.        <p>Click <a href="registerTeacher.html">Here</a> if you are a teacher</p>
16.        <p>Click <a href="registerParent.html">Here</a> if you are a parent</p>
17.        <p>Click<a href="registerAdminVerify.html">Here</a> if you are an admin</p>
18.    </center>
19. </body>

```





*registerAdminVerify.html*

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Admin Verification</center></b></h1>
13.    </header>
14.    <br>
15.    <!--form so that can check admin is admin -->
16.    <form action="registerAdminVerify.php" style="margin:auto;width:60%" method
="post">
17.      <label><b>Verification Code</b></label>
18.      <input class="w3-input w3-
border" type="number" name="verf" required><br>
19.      <input class="w3-button w3-block w3-dark-grey" type="submit">
20.      <br>
21.      <p>Please click <a href="register.php">here</a> to go back</p>
22.    </form>
23.  </body>
24. </html>

```

*registerAdminVerify.php*

```

1. <?php //if admin knows code (from registerAdminVerify form) then can proceed
2.   if ($_POST['verf'] == '83751733092558'){
3.     $goTo = "Location: adminSignUp.html";
4.     header($goTo);
5.   } else{
6.     $goTo = "Location: register.php";
7.     header($goTo);
8.   }
9. ?>

```

**Coding Style – File paths parametrised**

*registerParent.html*

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Parent Sign Up</center></b></h1>
13.    </header>
14.    <br>
15.    <!-- so a new parent can register with correct details-->
16.    <form action="registerParent.php" style="margin:auto;width:60%" method="pos
t">
17.      <label><b>First Name</b></label>

```

**Coding Style – Well designed User Interface**

```

18.         <input class="w3-input w3-
border" type="text" name="firstName" required>
19.         <label><b>Surname</b></label>
20.         <input class="w3-input w3-
border" type="text" name="surname" required>
21.         <label><b>Email</b></label>
22.         <input class="w3-input w3-border" type="email" name="email" required>
23.         <label><b>Username</b></label>
24.         <input class="w3-input w3-
border" type="text" name="username" required>
25.         <label><b>Password</b></label>
26.         <input class="w3-input w3-border" type="text" name="pw" required>
27.         <label><b>Re enter Password</b></label>
28.         <input class="w3-input w3-
border" type="text" name="confirmpw" required>
29.         <br>
30.         <input class="w3-button w3-block w3-dark-grey" type="submit">
31.         <br>
32.         <br>
33.         <p>Please click <a href="register.php">here</a> to go back</p>
34.     </form>
35. </body>
36. </html>

```

registerParent.php

```

1. <?php //register parent if meets all conditions
2.     function genLoginID($pid){ //function to add p to the id, so can differentiate
    between the different types of people
3.         $id="p".($pid);
4.         return $id;
5.     }
6.     function password_strength_check($password, $min_len = 6, $max_len = 9, $req_di
    git = 1, $req_lower = 1, $req_upper = 1, $req_symbol = 1) {
7.         //function to check strength of password
8.         // Build regex string depending on requirements for the password
9.         // url ref - https://stackoverflow.com/questions/11873990/create-preg-
    match-for-password-validation-allowing
10.        $regex = '/^';
11.        if ($req_digit == 1) { $regex .= '(?=.*\d)'; }
    // Match at least 1 digit
12.        if ($req_lower == 1) { $regex .= '(?=.*[a-
    z])'; } // Match at least 1 lowercase letter
13.        if ($req_upper == 1) { $regex .= '(?=.*[A-
    Z])'; } // Match at least 1 uppercase letter
14.        if ($req_symbol == 1) { $regex .= '(?=.*[^a-zA-
    Z\d])'; } // Match at least 1 character that is none of the above
15.        $regex .= '{' . $min_len . ',' . $max_len . '}$/';
16.        if(preg_match($regex, $password)) {
17.            return TRUE;
18.        } else {
19.            return FALSE;
20.        }
21.    }
22.
23.     function formatName($fname,$sname){ //url ref- https://www.w3schools.com/php/php
    _ref_string.asp
24.        //function to format name properly, so it returns correct capitalisation of
    name
25.        $lowerFName=strtolower($fname);
26.        $properFName = ucfirst($lowerFName);
27.        $lowerSName=strtolower($sname);
28.        $properSName = ucfirst($lowerSName);
29.        return $properFName." ".$properSName;
30.    }

```

### High-Level Systems Overview – Sign up process for parents

### Objective 3

Database able  
to store  
essential parent  
details

```

31.     if ($_POST["pw"]==$_POST["confirmpw"]){ //only proceeds when passwords are the
same
32.         if (password_strength_check($_POST["pw"]) == TRUE) { //make sure the
password has a good strength
33.             require "connection.php";
34.             require "parents.php";
35.             $conn = connection::getConn();
36.             //instantite parent object

37.             $h = new parents($_POST["username"],$_POST["pw"],$_POST["firstName"],$_
POST["surname"],$_POST["email"]);
38.             $firstName = $h->getFirstName();
39.             $surname = $h->getsurname();
40.             $email = $h->getEmail();
41.             $username = $h->getUsername();
42.             $password = $h->getPassword();
43.             $phash = password_hash($password,PASSWORD_DEFAULT);
44.             //insert the details into table
45.             $sql = "INSERT INTO `parent` (`firstName`, `surname`, `email`) VALUES (
`$firstName`, `$surname`, `$email`)";
46.             if ($conn->query($sql) === TRUE) {
47.                 $fullName = formatName($firstName,$surname);
48.                 echo "WELCOME ".$fullName.", You have successfully created an accou
nt."."<br>";

49.                 //get their ID so they can use for future ref
50.                 $sql = "SELECT MAX(parentID) as max FROM parent";
51.                 $result = $conn->query($sql);
52.                 if ($result->num_rows > 0) {
53.                     while($row=$result->fetch_assoc()){
54.                         $id = $row["max"];
55.                     }
56.                     $loginUN=$username;
57.                     $loginPW=$phash;
58.                     $loginID = "p".$id;
59.                     echo "Please remember the following detail(s): ".$<br>";
60.                     echo "ID is: ".$loginID."<br>";
61. $sql = "INSERT INTO `user` (`id`, `username`, `password`, `sid`, `t id`, `pid`)
VALUES (`$loginID`, `$loginUN`, `$loginPW`, NULL, NULL, `$id`)";
62.                 if ($conn->query($sql) === TRUE) {
63.                     echo "Please click ".<a href="login.html">here</a>". to be re
directed to the login page";
64.                 }else{
65.                     echo "Connection error";
66.                 }
67.             } else {
68.                 echo "Connection error";
69.             }
70.         }else{
71.             echo "Password too weak. Must have a digit, a lowercase letter, upperca
se letter, special character and between 6 to 9 characters.";
72.             echo "<br>";
73.             echo "Please click ".<a href="http://al2020-
2.tanglincomputing.com/al2020-
13/coursework/registerStudentLink.php">here</a>". to be redirected to the register
page";
74.         }
75.     }else{
76.         echo "different passwords entered";
77.         echo "<br>";
78.         echo "Please click ".<a href="http://al2020-2.tanglincomputing.com/al2020-
13/coursework/registerStudentLink.php">here</a>". to be redirected to the register
page";
79.     }
80. $conn->close();
?>

```

## registerStudent.php

```

1. <?php //register student if meet all conditions
2.     function genLoginID($sid){ //puts an s in front of the number, so can differentiate between the different types of users
3.         $id="s".($sid);
4.         return $id;
5.     }
6.     function formatName($fname,$sname){ //url ref- https://www.w3schools.com/php/php_ref_string.asp
7.         //function to format the name in nice capitalisation
8.         $lowerFName=strtolower($fname);
9.         $properFName = ucfirst($lowerFName);
10.        $lowerSName=strtolower($sname);
11.        $properSName = ucfirst($lowerSName);
12.        return $properFName." ".$properSName;
13.    }
14.    function password_strength_check($password, $min_len = 6, $max_len = 9, $req_digit = 1, $req_lower = 1, $req_upper = 1, $req_symbol = 1) {
15.        // Build regex string depending on requirements for the password
16.        //function to ensure the password entered is strong enough
17.        // url ref - https://stackoverflow.com/questions/11873990/create-preg-match-for-password-validation-allowing
18.        $regex = '/^';
19.        if ($req_digit == 1) { $regex .= '(?=.*\d)'; } // Match at least 1 digit
20.        if ($req_lower == 1) { $regex .= '(?=.*[a-z])'; } // Match at least 1 lowercase letter
21.        if ($req_upper == 1) { $regex .= '(?=.*[A-Z])'; } // Match at least 1 uppercase letter
22.        if ($req_symbol == 1) { $regex .= '(?=.*[^\a-zA-Z\d])'; } // Match at least 1 character that is none of the above
23.        $regex .= '{' . $min_len . ',' . $max_len . '}$/';
24.
25.        if(preg_match($regex, $password)) {
26.            return TRUE;
27.        } else {
28.            return FALSE;
29.        }
30.    }
31.    require "connection.php";
32.    $conn = connection::getConnection();
33.    $un = $_POST["username"];
34.    $sql = "SELECT * FROM `user` WHERE username='$un'";
35.    $result = $conn->query($sql);
36.    if ($result->num_rows > 0) {
37.        echo "username already exists";
38.        echo "<br>";
39.        echo "Please click ".<a href="http://al2020-2.tanglincomputing.com/al2020-13/coursework/registerStudentLink.php">here</a>." to be redirected to the register page";
40.    }else{
41.        if ($_POST["pw"]==$_POST["confirmpw"]){ //ensures that the password entered are the same
42.            if (password_strength_check($_POST["pw"]) == TRUE ){
43.                require "student.php";
44.                //instantiate a new student object
45.                $h = new student($_POST["firstName"],$_POST["surname"],$_POST["school"],$_POST["dob"],$_POST["username"],$_POST["pw"]);
46.                $firstName = $h->getFirstName();
47.                $surname = $h->getSurname();
48.                $school = $h->getSchool();
49.                $dob = $h->getDOB();
50.                $username = $h->getUsername();
51.                $password = $h->getPassword();

```

**Coding Style – Self Documenting Code**

**Coding Style – Meaningful Identifier Names**

**Coding Style – Consistent style throughout**

Password Hash Function -  
password is hashed

## Objective 2

Database  
record student  
details

SQL –  
Insert new  
student  
into  
database

```

52.         $phash = password_hash($password,PASSWORD_DEFAULT);
53.         if (password_Verify($password, $phash)){
54.             //insert details into db for future ref
55.             $sql = "INSERT INTO `student` (`firstName`, `surname`, `school`
56.             , `dateOfBirth`) VALUES ('$firstName', '$surname', '$school', '$dob')";
57.             if ($conn->query($sql) === TRUE) {
58.                 $fullName = formatName($firstName,$surname);
59.                 echo "WELCOME ".$fullName.", You have successfully created
60.                 an account."."<br>";
61.                 //gets the student ID for future use
62.                 $sql = "SELECT MAX(studentID) as max FROM student";
63.                 $result = $conn->query($sql);
64.                 if ($result->num_rows > 0) {
65.                     while($row=$result->fetch_assoc()){
66.                         $id = $row["max"];
67.                     }
68.                     $loginUN=$username;
69.                     $loginPW=$phash;
70.                     $loginID=genLoginID($id);
71.                     echo "Please remember the following details: ".$<br>";
72.                     echo "ID is: ".$loginID."<br>";
73.                     //inserts the student into user table for future ref
74.                     $sql = "INSERT INTO `user` (`id`, `username`, `password`
75.                     , `sid`, `tid`, `pid`) VALUES ('$loginID', '$loginUN', '$loginPW', '$id', NULL, NUL
76.                     L)";
77.                     echo "Please click ".<a href="http://al2020-
78.                     2.tanglincomputing.com/al2020-
79.                     13/coursework/login.html">here</a> to be redirected to the login page";
80.                     if ($conn->query($sql) === TRUE) {
81.                         }else{
82.
83.                     }
84.                     echo "Connection error";
85.                 }
86.             }
87.         } else {
88.             echo "Connection error";
89.         }
90.     } else {
91.         //if password too weak, redirect to registration page
92.         echo "Password too weak. Must have a digit, a lowercase letter, upp
93.         ercase letter, special character and between 6 to 9 characters.";
94.         echo "<br>";
95.         echo "Please click ".<a href="http://al2020-
96.         2.tanglincomputing.com/al2020-
97.         13/coursework/registerStudentLink.php">here</a> to be redirected to the register
98.         page";
99.     }
100. } else {
101.     //if password different, redirect to registration page
102.     echo "Different passwords entered";
103.     echo "<br>";
104.     echo "Please click ".<a href="http://al2020-
105.     2.tanglincomputing.com/al2020-
106.     13/coursework/registerStudentLink.php">here</a> to be redirected to the register
107.     page";
108. }
109. }
110. }
111. $conn->close();
112. ?>

```

Password Hash Function  
Hashed password stored  
in database

## Coding Style

Good  
exception  
handling

## registerStudentLink.php

```

1. <?php //form for student to sign up
2.     $time = new DateTime('now');
3.     $newtime = $time->modify('-10 year')->format('Y-m-d');
4. ?>
5. <html>
6.     <head>
7.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
8.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
9.         <style>
10.             body {font-family: "Raleway", Arial, sans-serif}
11.             .w3-row img {margin-bottom: -8px}
12.         </style>
13.     </head>
14.     <body style="background-color:#add8e6;">
15.         <header>
16.             <h1><b><center>Student Sign Up</center></b></h1>
17.         </header>
18.         <br>
19.         <!--login form (where user inputs stuff-->
20.         <form style="margin:auto;width:60%" action="registerStudent.php" method="post">
21.             <label><b>First Name</b></label>
22.             <input class="w3-input w3-border" type="text" name="firstName" required>
23.             <label><b>Surname</b></label>
24.             <input class="w3-input w3-border" type="text" name="surname" required>
25.             <label><b>School</b></label>
26.             <select class="w3-input w3-border" name = "school" >
27.                 <option value="" disabled selected>Please select your school</option>
28.                 <option value = "CIS">Canadian International School</option>
29.                 <option value = "OFS">Overseas Family School</option>
30.                 <option value = "OSS">Outram Secondary School</option>
31.                 <option value = "TSS">Tanglin Secondary School</option>
32.                 <option value = "TTS">Tanglin Trust School</option>
33.                 <option value = "YSS">Yuying Secondary School</option>
34.             </select>
35.             <label><b>Date of birth</b></label>
36.             <input class="w3-input w3-border" type="date" name="dob" max="<?php echo $newtime;?>" required>
37.             <label><b>Username</b></label>
38.             <input class="w3-input w3-border" type="text" name="username" required>
39.             <label><b>Password</b></label>
40.             <input class="w3-input w3-border" type="text" name="pw" required>
41.             <label><b>Re enter password</b></label>
42.             <input class="w3-input w3-border" type="text" name="confirmpw" required>
43.             <br>
44.             <input type="submit" class="w3-button w3-block w3-dark-grey">
45.             <br>
46.             <br>
47.             <p>Please click <a href="register.php">here</a> to go back</p>
48.         </form>
49.     </body>
50. </html>

```

registerTeacher.html

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Teacher Sign Up</center></b></h1>
13.    </header>
14.    <br>
15.    <!-- so a new teacher can register with correct details-->
16.    <form action="registerTeacher.php" style="margin:auto;width:60%" method="po
st">
17.      <label><b>First Name</b></label>
18.      <input class="w3-input w3-
border" type="text" name="firstName" required>
19.      <label><b>Surname</b></label>
20.      <input class="w3-input w3-
border" type="text" name="surname" required>
21.      <label><b>Email</b></label>
22.      <input class="w3-input w3-border" type="email" name="email" required>
23.      <label><b>Username</b></label>
24.      <input class="w3-input w3-
border" type="text" name="username" required>
25.      <label><b>Password</b></label>
26.      <input class="w3-input w3-border" type="text" name="pw" required>
27.      <label><b>Re enter Password</b></label>
28.      <input class="w3-input w3-
border" type="text" name="confirmpw" required>
29.      <br>
30.      <input class="w3-button w3-block w3-dark-grey" type="submit">
31.      <br>
32.      <br>
33.      <p>Please click <a href="register.php">here</a> to go back</p>
34.    </form>
35.  </body>
36. </html>

```

registerTeacher.php

**Coding Styles – Annotation used effectively when required**

```

1. <?php //check teacher sign up meets requirements
2.   function genLoginID($tid){ //add t infront of number so that can easily differe
ntiate between different types of users in the db
3.     $id="t".($tid);
4.     return $id;
5.   }
6.   function formatName($fname,$sname){ //url ref- https://www.w3schools.com/php/php
_ref_string.asp
7.     //to format name nicely with correct capitalisation
8.     $lowerFName=strtolower($fname);
9.     $properFName = ucfirst($lowerFName);
10.    $lowerSName=strtolower($sname);
11.    $properSName = ucfirst($lowerSName);
12.    return $properFName." ".$properSName;
13.  }

```



```

14.     function password_strength_check($password, $min_len = 6, $max_len = 9, $req_di
    git = 1, $req_lower = 1, $req_upper = 1, $req_symbol = 1) {
15.         // Build regex string depending on requirements for the password
16.         //check password entered is of correct strength, meeting the requirements
17.         // url ref - https://stackoverflow.com/questions/11873990/create-preg-
    match-for-password-validation-allowing
18.         $regex = '/^';
19.         if ($req_digit == 1) { $regex .= '(?=.*\d)'; }
    // Match at least 1 digit
20.         if ($req_lower == 1) { $regex .= '(?=.*[a-
    z])'; } // Match at least 1 lowercase letter
21.         if ($req_upper == 1) { $regex .= '(?=.*[A-
    Z])'; } // Match at least 1 uppercase letter
22.         if ($req_symbol == 1) { $regex .= '(?=.*[^a-zA-
    Z\d])'; } // Match at least 1 character that is none of the above
23.         $regex .= '{' . $min_len . ',' . $max_len . '}$/';
24.
25.         if(preg_match($regex, $password)) {
26.             return TRUE;
27.         } else {
28.             return FALSE;
29.         }
30.     }
31.     if ($_POST["pw"]==$_POST["confirmpw"]){ //checks the passwords are the same
32.         if (password_strength_check($_POST["pw"]) == TRUE ){ //checks that the
    password meets the requirements
33.             require "connection.php";
34.             require "teacher.php";
35.             $conn = connection::getConn();
36.             if ($conn->connect_error) {
37.                 die("Connection failed: " . $conn->connect_error);
38.             }
39.             //instantiate teacher object
40. $h = new teacher($_POST["username"],$_POST["pw"],$_POST["firstName"],$_
    POST["surname"],$_POST["email"]);
41.             $firstName = $h->getFirstName();
42.             $surname = $h->getSurname();
43.             $email = $h->getEmail();
44.             $username = $h->getUsername();
45.             $password = $h->getPassword();
46.             $phash = password_hash($password,PASSWORD_DEFAULT);
47.             if (password_Verify($password, $phash)){
48.                 //insert details into table for future use
49.                 $sql = "INSERT INTO `teacher` (`firstName`, `surname`, `email`)
    VALUES ('$firstName', '$surname', '$email')";
50.                 if ($conn->query($sql) === TRUE) {
51.                     $fullName = formatName($firstName,$surname);
52.                     echo "WELCOME ".$fullName.", You have successfully created an a
    ccount."."<br>";
53.                     //get the teacher ID of this teacher added
54.                     $sql = "SELECT MAX(teacherID) as max FROM teacher";
55.                     $result = $conn->query($sql);
56.                     if ($result->num_rows > 0) {
57.                         while($row=$result->fetch_assoc()){
58.                             $id = $row["max"];
59.                         }
60.                         $loginUN=$username;
61.                         $loginPW=$phash;
62.                         $loginID=genLoginID($id);
63.                         echo "Please remember the following details: ".$<br>";
64.                         echo "Username is: ".$loginUN."<br>";
65.                         echo "Password is: ".$loginPW."<br>";
66.                         echo "ID is: ".$loginID."<br>";
67.                         //insert their details into the user db
68.                         $sql = "INSERT INTO `user` (`id`, `username`, `password`, `sid`
    , `tid`, `pid`) VALUES ('$loginID', '$loginUN', '$loginPW', NULL, '$id', NULL)";

```

**Objective 4**

Database  
records  
teacher details

```

69.         if ($conn->query($sql) === TRUE) {
70.             echo "Please click ".<a href="login.html">here</a>". to be
e redirected to the login page";
71.         }else{
72.             echo "Error: " . $sql . "<br>" . $conn->error;
73.         }
74.     } else {
75.         echo "Error: " . $sql . "<br>" . $conn->error;
76.     }
77. } else {
78.     echo "Error: " . $sql . "<br>" . $conn->error;
79. }
80. }else{
81.     //if password does not meet strength requirements, redirects to the
register page
82.     echo "Password too weak. Must have a digit, a lowercase letter, upperca
se letter, special character and between 6 to 9 characters.";
83.     echo "<br>";
84.     echo "Please click ".<a href="registerTeacher.html">here</a>". to be
redirected to the register page";
85. }
86. }else{
87.     //if passwords entered are not the same, redirects user to register page
88.     echo "different passwords entered";
89.     echo "<br>";
90.     echo "Please click ".<a href="registerTeacher.html">here</a>". to be redi
rected to the register page";
91. }
92. $conn->close();
93. ?>

```

*seeAllClassrooms.php*

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>All Classrooms </center></b></h1>
13.    </header>
14.    <br>
15.  </body>
16. </html>
17. <?php //let admin see all classroom connections
18.   require "connection.php";
19.   $conn = connection::getConn();
20.   $sql = "SELECT COUNT(studentID) as total FROM student"; //get total number of students
21.   $result = $conn->query($sql);
22.   $row = $result->fetch_assoc();
23.   echo '<b>'. "Total number of students: ". $row['total'];
24.   echo "<br>";
25.   $sql = "SELECT COUNT(teacherID) as total FROM teacher"; //get total number of teachers
26.   $result = $conn->query($sql);
27.   $row = $result->fetch_assoc();

```

**SQL – Admin see total number of students**

**Technique – Aggregate Function**

```

28.     echo "Total number of teachers: ". $row['total'];
29.     echo "<br>";
30. $sql = "SELECT COUNT(classroomID) as total FROM classroom"; //get total number of
    classrooms
31.     $result = $conn->query($sql);
32.     $row = $result->fetch_assoc();
33.     echo "Total number of classrooms: ". $row['total'].'</b>';
34.     echo "<br>";
35.     echo "<br>";
36.     $dataS = [];
37. $sql = "select student.firstName, student.surname from classroom,teacher,student
    where classroom.teacherID = teacher.teacherID and classroom.studentID = student.s
    tudentID";
38.     $result = $conn->query($sql);
39.     $count = 0;
40.     if ($result->num_rows > 0) {
41.         while($row = $result->fetch_assoc()) {
42.             $count = $count + 1;
43.             $dataS[]=[
44.                 'studentFName' =>$row["firstName"],
45.                 'studentSName' => $row["surname"]
46.             ];
47.         }
48.     } else {
49.         echo "0 students";
50.     }
51.     $dataT=[];
52. $sql = "select teacher.firstName, teacher.surname from classroom,teacher,student
    where classroom.teacherID = teacher.teacherID and classroom.studentID = student.s
    tudentID";
53.     $result = $conn->query($sql);
54.     $count = 0;
55.     if ($result->num_rows > 0) {
56.         while($row = $result->fetch_assoc()) {
57.             $count = $count + 1;
58.             $dataT[]=[
59.                 'teacherFName' =>$row["firstName"],
60.                 'teacherSName' => $row["surname"]
61.             ];
62.         }
63.
64.         echo "<table border=1 frame=box rules=all>";
65.
66.         echo "<tr><td width=20%>" . "<b>Student F Name</b>" . "</td><td width=20%>"
        . "<b>Student S Name</b>" . "</td><td width=20%>" . "<b>Teacher F Name</b>" . "</t
        d><td width=20%>" . "<b>Teacher S Name</b>" . "</td></tr>";
67.         for ($x = 0; $x <= $count; $x++) {
68.             echo "<tr><td width=20%>" . $dataS[$x]['studentFName'] . "</td><td widt
        h=20%>" . $dataS[$x]['studentSName'] . "</td><td width=20%>" . $dataT[$x]['teacherF
        Name'] . "</td><td width=20%>" . $dataT[$x]['teacherSName'] . "</td></tr>";
69.         }
70.         echo "</table>";
71.         echo "<br>";
72.         echo "Please click ".<a href="manageClassroom.php" >here</a>". to manage
        classroom";
73.         echo "<br>";
74.     } else {
75.         echo "0 families";
76.     }
77.     echo "<br>";
78.     echo "Please click ".<a href="homepage.php" >here</a>". to go back";
79.     ?>

```

**Objective 6**

Admin view all  
links between  
students and  
teachers

Get all the student details

SQL – Admin  
manage all  
classroom links

Display table of current  
students in a classroom,  
listing the student's name  
and teacher's name

**High-Level Systems**

**Overview** – for admin to  
view all classrooms

seeAllFamilies.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>All families </center></b></h1>
13.        </header>
14.        <br>
15.    </body>
16. </html>
17. <?php    //let admin see all family connections
18.    require "connection.php";
19.    $conn = connection::getConn();
20.    $sql = "SELECT COUNT(studentID) as total FROM student"; //get total number of students
21.    $result = $conn->query($sql);
22.    $row = $result->fetch_assoc();
23.    echo '<b>'. "Total number of students: ". $row['total'];
24.    echo "<br>";
25.    $sql = "SELECT COUNT(parentID) as total FROM parent"; //get total number of parents
26.    $result = $conn->query($sql);
27.    $row = $result->fetch_assoc();
28.    echo "Total number of parents: ". $row['total'];
29.    echo "<br>";
30.    $sql = "SELECT COUNT(familyID) as total FROM family"; //get total number of families
31.    $result = $conn->query($sql);
32.    $row = $result->fetch_assoc();
33.    echo "Total number of families: ". $row['total']. '</b>';
34.    echo "<br>";
35.    echo "<br>";
36.    $dataS = [];
37.    $sql = "select student.firstName, student.surname from family,parent,student where family.parentID = parent.parentID and family.studentID = student.studentID";
38.    $result = $conn->query($sql);
39.    $count = 0;
40.    if ($result->num_rows > 0) {
41.        while($row = $result->fetch_assoc()) {
42.            $count = $count + 1;
43.            $dataS[]=[
44.                'studentFName' =>$row["firstName"],
45.                'studentSName' => $row["surname"]
46.            ];
47.        }
48.    } else {
49.        echo "0 families";
50.    }
51.    $dataP=[];
52.    $sql = "select parent.firstName, parent.surname from family,parent,student where family.parentID = parent.parentID and family.studentID = student.studentID";
53.    $result = $conn->query($sql);
54.    $count = 0;
55.    if ($result->num_rows > 0) {
56.        while($row = $result->fetch_assoc()) {
57.            $count = $count + 1;

```

### Objective 8

Admin view all links between students and parents

SQL – Admin manage family links

```

58.         $dataP[]=[
59.             'parentFName' => $row["firstName"],
60.             'parentSName' => $row["surname"] 61.
        ];
62.     }
63.     echo "<table border=1 frame=box rules=all>";
64.     echo "<tr><td width=20%>" . "<b>Student F Name</b>" . "</td><td width=20%>"
        . "<b>Student S Name</b>" . "</td><td width=20%>" . "<b>Parent F Name</b>" . "</td>
        <td width=20%>" . "<b>Parent S Name</b>" . "</td></tr>";
65.     for ($x = 0; $x <= $count; $x++) {
66.         echo "<tr><td width=20%>" . $dataS[$x]['studentFName'] . "</td><td width=20%>" . $dataS[$x]['studentSName'] . "</td><td width=20%>" . $dataP[$x]['parentFName'] . "</td><td width=20%>" . $dataP[$x]['parentSName'] . "</td></tr>";
67.     }
68.     echo "</table>";
69.     echo "<br>";
70. echo "Please click ".<a href="manageFamily.php" >here</a>". to manage families";
71.     } else {
72.         echo "0 families";
73.     }
74. ?>

```

### High-Level Systems Overview

Admin viewing all existing families

seeClassroom.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Classroom</center></b></h1>
13.        </header>
14.    </body>
15. </html>
16. <?php //let teacher see classroom connections they have
17.     require "connection.php";
18.     $conn = connection::getConn();
19.     $teacherID = substr($_GET['id'], 1);
20.     //gets all the students where the teacher is equal to the one entered and outputs it all
21.     $sql="SELECT * FROM student INNER JOIN classroom ON student.studentID = classroom.studentID and classroom.teacherID='$teacherID'";
22.     $result = $conn->query($sql);
23.     echo "Here is a list of your students - names and school";
24.     echo "<br>";
25.     echo "<table border=1 frame=box rules=all>";
26.     echo "<tr><td width=20%>" . "<b>First name </b>".</td><td width=20%>".<b>Surname</b>" . "</td><td width=20%>".<b>School</b>" . "</td></tr>";
27.     if ($result->num_rows > 0) {
28.         while($row = $result->fetch_assoc()) {
29.             echo "<tr><td width=20%>" . $row['firstName'] . "</td><td width=20%>" . $row['surname'] . "</td><td width=20%>" . $row['school'] . "</td></tr>";
30.         }
31.     } else {
32.         echo "0 results found";
33.     }
34.     echo "</table>";
35.     echo "<br>";

```

High-Level Systems Overview – Teacher see who is in their classroom

```

36.     echo "Please click ".<a href="homepage.php">here</a>'. " to go back";
37. ?>

```

seeFamily.php

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Family</center></b></h1>
13.    </header>
14.  </body>
15.</html>
16.<?php //let parent see family connections they have
17.  require "connection.php";
18.  $conn = connection::getConn();
19.  $parentID = substr($_GET['id'], 1);
20.  //selects all students which have the parent ID of
    the one shared through the page and outputs it all
21.  $sql="SELECT * FROM student INNER JOIN family ON student.studentID = family.studentID and family.parentID='$parentID'";
22.  $result = $conn->query($sql);
23.  echo "Here is a list of your children - names and id";
24.  echo "<br>";
25.  echo "<table border=1 frame=box rules=all>";
26.  echo "<tr><td width=20%>". "<b>First name </b>". "</td><td width=20%>". "<b>Surname</b>". "</td><td width=20%>". "<b>ID</b>". "</td></tr>";
27.  if ($result->num_rows > 0) {
28.    while($row = $result->fetch_assoc()) {
29.      echo "<tr><td width=20%>". $row['firstName'] . "</td><td width=20%>".
    $row['surname'] . "</td><td width=20%>". $row['studentID'] . "</td></tr>";
30.    }
31.  } else {
32.    echo "0 results found";
33.  }
34.  echo "</table>";
35.  echo "<br>";
36.  echo "Please click ".<a href="homepage.php">here</a>'. " to go back";
37. ?>

```

Technique – Cross-table parametrised SQL

### Objective 13

Parents able to view their family

High-Level Systems Overview-  
for parents to view their family

select.html

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">
11.    <header>
12.      <h1><b><center>Student Progress Tracker</center></b></h1>
13.    </header>

```

```

14.         <br>
15.         <!--form to write student ID of students who want to view progress of-->
16.         <form action="select.php" style="margin:auto;width:60%" method="post">
17.             <label><b>Student ID</b></label>
18.             <input class="w3-input w3-
border" type="number" name="studentID" required><br>
19.             <input class="w3-button w3-block w3-dark-grey" type="submit">
20.             <br>
21.             <br>
22.             <p>Please click <a href="homepage.php">here</a> to go back</p>
23.         </form>
24.     </body>
25. </html>

```

select.php

```

1. <?php //Displays students progress - graph and table and select topic further look
2.     require "connection.php";
3.     $conn = connection::getConn();
4.     session_start();
5.     $_SESSION["studentID"]=$_POST["studentID"];
6.     $id = $_SESSION["studentID"];
7.     $plainID = substr($id, 0);
8.     $sql = "SELECT * FROM `student` WHERE `studentID`=".$plainID;
9.     $result = $conn->query($sql);
10.    if ($result->num_rows > 0) {
11.        while($row = $result->fetch_assoc()) {
12.            $firstName = $row["firstName"];
13.        }
14.    }
15. ?>
16. <html>
17.     <head>
18.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
19.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
20.         <style>
21.             body {font-family: "Raleway", Arial, sans-serif}
22.             .w3-row img {margin-bottom: -8px}
23.         </style>
24.     </head>
25.     <body style="background-color:#add8e6;">
26.         <header>
27.             <h1><b><center>Student Scores for <?php echo $firstName;?></center></b>
</h1>
28.         </header>
29.     </body>
30. </html>
31. <?php
32.     $_SESSION["studentID"]=$_POST["studentID"];
33.     $data = [];

```

### Objective 37

Teacher see  
students  
progress in  
comparison to  
other topics

**High-Level Systems Overview –**  
Teacher views a students  
progress on the homework  
sheets

```

34.     //get all the test id,names, score and date for that student and put into an
array
35.     $sql = "SELECT * FROM `test`,`topic`,`testAndTopic` WHERE test.studentID='{$_SE
SSION["studentID"]}' AND test.testID=testAndTopic.testID AND testAndTopic.topicID=t
opic.topicID";
36.     $result = $conn->query($sql);

37.
38.
39.
40.
41.
42.
43.
44.     $count = 0;
    if ($result->num_rows > 0) {
        while($row = $result-
>fetch_assoc()) {
            $count = $count + 1;
            $data[]=[
                'testID' => $row["testID"],

```

Roshni

```
'studentScore'
=>$row["studentScore"],
'topicName' =>
$row["topicName"],
```

**Algorithm Design –**  
student tests scores and  
information are put into table, if  
they exist, else an error message is  
displayed



```

45.         'date' =>$row["Date"],
46.         'topicID' =>$row["topicID"] 47.
47.     ];
48.     }
49. } else {
50.     echo "student has not don't any tests";
51. }
52. $table = array();
53. for ($x = 0; $x <=20; $x++){
54.     $counter = 0;
55.     $total = 0;
56.     for ($y = 0; $y <= ($count-1); $y++) {
57.         if ($x == $data[$y]['topicID']){
58.             $total = $total + ($data[$y]['studentScore']);
59.             $counter = $counter + 1;
60.         }
61.     }
62.     if ($counter>0){
63.         $average = $total / $counter;
64.     }else{
65.         $average = 0;
66.     }
67.     array_push($table,$average);
68.     $counter = 0;
69. }
70. echo "<br>";
71. $counter = 0;
72. $dataPoints = array(
73.     array("y" => $table[$counter+1], "label" => "Addition" ),
74.
75.     array("y" => $table[$counter+2], "label" => "Subtraction" ),
76.     array("y" => $table[$counter+3], "label" => "Numbers" ),
77.     array("y" => $table[$counter+4], "label" => "Multiplication" ),
78.     array("y" => $table[$counter+5], "label" => "Division" ),
79.     array("y" => $table[$counter+6], "label" => "Mixed Operations" ),
80.     array("y" => $table[$counter+7], "label" => "Sequences" ),
81.     array("y" => $table[$counter+8], "label" => "Decimals" ),
82.     array("y" => $table[$counter+9], "label" => "Ratio" ),
83.     array("y" => $table[$counter+10], "label" => "Percentages" ),
84.     array("y" => $table[$counter+11], "label" => "Time" ),
85.     array("y" => $table[$counter+12], "label" => "Rounding and Estimation" ),
86.     array("y" => $table[$counter+13], "label" => "Area" ),
87.     array("y" => $table[$counter+14], "label" => "Perimeter" ),
88.     array("y" => $table[$counter+15], "label" => "Probability" ),
89.     array("y" => $table[$counter+16], "label" => "Factorisation" ),
90.     array("y" => $table[$counter+17], "label" => "Substitution" ),
91.     array("y" => $table[$counter+18], "label" => "Conversion of measurements" )
92. );

```

**Graph API -**  
Populate array

**Algorithm Design –**  
Creating the bar chart to  
show the average scores

**Graph API-**  
Populate data values

```

93. <?>
94. <!DOCTYPE HTML>
95. <html>

```

**Technique – Calling parametrized Web Service APIs**

```

96. <head>
97. <script>
98.     window.onload = function() {
99.         var chart = new CanvasJS.Chart("chartContainer", {

```

364100.

Roshni

```
107.         ge score (%)"
108.     },
109.     data: [{
        type: "column",

102.     ,
        title: {
103.             text: "
                A
                v
                e
                r
                a
                g
                e
                s
                c
                o
                r
                e
                s
                i
                n
                t
                o
                p
                i
                c
                "

104.     },
105.     axis: {
106.         title: "
                A
                v
                e
                r
                a
```

**Graph API**  
Calling the API

**Algorithm  
Design –**  
Drawing  
the graph  
onto the  
screen

```

110.         yValueFormatString: "#,##0.## percent",
111.         dataPoints: <?php echo json_encode($dataPoints, JSON
NUMERIC_CHECK); ?>
112.     }
113. });
114.     chart.render();
115.
116.     }
117. </script>
118. </head>
119. <body>
120.     <div id="chartContainer" style="height: 370px; width: 100%;"></div>
121.     <script src="https://canvasjs.com/assets/script/canvasjs.min.js"></s
cript>
122.     </body>
123. </html>
124. <?php
125.     echo "<br>";
126.     echo "<table margin:auto border=1 frame=box rules=all>";
127.     echo "<tr><td width=10%>" . "<b>ID </b>" . "</td><td width=20%>" . "<b>Sco
re</b>" . "</td><td width=40%>" . "<b>Topic</b>" . "</td><td width=20%>" . "<b>Date</b>" . "</td></tr>";
128.     for ($x = 0; $x <= ($count-1); $x++) {
129.         echo "<tr><td width=10%>" . ($data[$x]['testID']) . "</td><td width=20
%>" . ($data[$x]['studentScore']) . "</td><td width=40%>" . ($data[$x]['topicName']) . "</t
d><td width=20%>" . ($data[$x]['date']) . "</td></tr>";
130.     }
131.     echo "</table>";
132.     echo "Please click ". "<a href='selectStudentPDF.php' target='_blank'>her
e</a>". " to view as a pdf";
133.     echo "<br>";
134.     echo "<br>";
135.     ?>
136. <html>
137.     <body>
138.         <p>Please select the topic you want to view the student's progress
of
139.         <form action="selectTest.php" method="post" style="width:30%">
140.             <select name="topicID" class="w3-input w3-border" >
141.                 <option value="" disabled selected>Please select the topic
</option>
142.                 <option value="1">Addition</option>
143.                 <option value="2">Subtraction</option>
144.                 <option value="3">Numbers</option>
145.                 <option value="4">Multiplication</option>
146.                 <option value="5">Division</option>
147.                 <option value="6">Mixed Operations</option>
148.                 <option value="7">Sequences</option>
149.                 <option value="8">Decimal</option>
150.                 <option value="9">Ratios</option>
151.                 <option value="10">Percentage</option>
152.                 <option value="11">Time</option>
153.                 <option value="12">Round and Estimate</option>
154.                 <option value="13">Area of shapes</option>
155.                 <option value="14">Perimeter of shapes</option>
156.                 <option value="15">Probability</option>
157.                 <option value="16">Factorisation</option>
158.                 <option value="17">Substitution</option>
159.                 <option value="18">Conversion of measurements</option>
160.                 <option value="19">Rates and Speed</option>
161.             </select>
162.             <br>
163.             <input type="submit">
164.             <br>
165.         <p>Please click <a href="homepage.php">here</a> to go back</p>

```

**Algorithm  
Design –  
Selecting  
the topic**

```

166.         <br>
167.     </form>
168. </body>
169. </html>

```

*selectStudentFamily.php*

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php //allow parent to view one of their childrens tests
14.     require "connection.php";
15.     session_start();
16.     $conn = connection::getConn();
17.     $_SESSION["studentID"]=$_POST["studentID"];
18.     $_SESSION["parentID"]=$_POST["parentID"];
19.     $len = strlen($_SESSION["parentID"]);
20.     $id = substr($_SESSION["parentID"], 1, $len);

21.     //select student details where there is a family, else dont display anything
22.     $sql2 = "select * from student, parent, family where student.studentID = family
.studentID and parent.parentID = family.parentID and family.parentID = '$id' and fa
mily.studentID = '{$_SESSION['studentID']}'";
23.     $result2 = $conn->query($sql2);
24.     $num2 = $result2->num_rows;
25.     if ($num2 == 1) {
26.         while($row = $result2->fetch_assoc()) {
27.             echo "<table border=1 frame=box rules=all>";

28.             echo "<tr><td width=10%><b>ID </b></td><td width=20%>." <b>Score</b> "
.</td><td width=50%>." <b>Topic</b> " <b>Topic</b> "</td></tr>";
29.             //select and output all the testID, score and topicName for the student

30.             $sql = "SELECT * FROM `test`,`topic`,`testAndTopic` WHERE test.studentID= '{$_SESSION['studentID']}' AND test.testID=testAndTopic.testID AND testAndTopic.t
opicID=topic.topicID";
31.             $result = $conn->query($sql);
32.             if ($result->num_rows > 0) {
33.                 while($row = $result->fetch_assoc()) {
34.                     echo "<tr><td width=10%>".$row["testID"]."</td><td width=20%>
".$row["studentScore"]."</td><td width=50%>".$row["topicName"]."</td></tr>";
35.                 }
36.                 echo "</table>";
37.                 //require "topic.php";
38.             } else {
39.                 echo "Child not completed any sheets";
40.             }
41.         }
42.     } else {
43.         echo "<html>";
44.         echo "<body>";
45.         echo "<br>";
46.         echo "<br>";
47.         echo "<br>";
48.         echo "<p><center><b>You have not created a family with this child</b></cent
er></p>";
49.         echo "</body>";

```

**SQL** – Check family exists between student and parent ID given

### Objective 15

Parent able to access their students overall scores for each topic

**High-Level Systems Overview** – to view a child's progress and attempt on the homeworks

```

50.         echo "</html>";
51.     }
52. ?>
53. <html>
54.     <body>
55.         <p><center>Please click <a href="homepage.php">here</a> to go back</center>
56.     </p>
57.     </body>
58. </html>

```

*selectStudentFamilyView.php*

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.        <header>
12.            <h1><b><center>Student Progress Tracker</center></b></h1>
13.        </header>
14.        <!--form for parent to access their childs info -->
15.        <form action="selectStudentFamily.php" method="post" style="margin:auto;wid
16.            <label><b>Student ID</b></label>
17.            <input class="w3-input w3-
18.            <input type="number" name="studentID" required><br>
19.            <input type="hidden" name="parentID" value= <?php echo $_GET['id'];?> >
20.            <br>
21.            <input class="w3-button w3-block w3-dark-grey" type="submit">
22.            <br>
23.            <p>Please click <a href="homepage.php">here</a> to go back</p>
24.        </form>
25.    </body>
26. </html>

```

*selectStudentPDF.php*

```

1. <?php //teacher print score, topic and date as a pdf for all topics for 1student
2.     require "connection.php";
3.     $conn = connection::getConn();
4.     session_start();
5.     $id = $_SESSION["studentID"];
6.     $plainID = substr($id, 0);
7.     $sql = "SELECT * FROM `student` WHERE `studentID`=" . $plainID;
8.     $result = $conn->query($sql);
9.     if ($result->num_rows > 0) {
10.        while($row = $result->fetch_assoc()) {
11.            $firstName = $row["firstName"];
12.        }
13.    }
14.    require('fpdf.php');
15.    $pdf = new FPDF();
16.    $pdf->AddPage();
17.    $pdf->SetFont('Arial', 'B', 16);

```

Create pdf

```

18. $data = [];
19. $message = "Student Scores for ".$firstName;
20. $pdf->Cell(40,10,$message);
21. $pdf->Ln(10);
22. $pdf->SetFont('Arial','',12);
23. //get all the test id,names, score and date for that student and put into an
    array
24. $sql = "SELECT * FROM `test`,`topic`,`testAndTopic` WHERE test.studentID='{$_SESSION["studentID"]}' AND test.testID=testAndTopic.testID AND testAndTopic.topicID=testAndTopic.topicID";
25. $result = $conn->query($sql);
26. $count = 0;
27. if ($result->num_rows > 0) {
28.     while($row = $result->fetch_assoc()) {
29.         $count = $count + 1;
30.         $data[]=[
31.             'studentScore' =>$row["studentScore"],
32.             'topicName' => $row["topicName"],
33.             'date' =>$row["Date"] 34.
        ];
35.     }
36. }
37. $pdf->Cell(40,10,'student score');
38. $pdf->Cell(100,10,'topic name');
39. $pdf->Cell(40,10,'date');
40. $pdf->Ln(10);
41. $pdf->SetFont('Arial','B',12);
42. // $pdf->Cell(40,10,strval($data[0]['testID']));
43. $pdf->Ln(10);
44. foreach($data as $key=>$row){
45.     $pdf->Cell(40,10,$data[$key]['studentScore']);
46.     $pdf->Cell(100,10,$data[$key]['topicName']);
47.     $pdf->Cell(40,10,$data[$key]['date']);
48.     $pdf->Ln(10);
49. }
50. $pdf->Output();
51. ?>

```

SQL – See student performance overall

**Data structure**  
Adding elements to an associative array

**Data structure – Printing an associative array**

*selectStudentTestPDF.php*

```

1. <?php //teacher print question, score and reason as pdf for a topic
2. require "connection.php";
3. $conn = connection::getConn();
4. session_start();
5. $id = $_SESSION["studentID"];
6. $plainID = substr($id, 0);
7. $sql = "SELECT * FROM `student` WHERE `studentID`='".$plainID";
8. $result = $conn->query($sql);
9. if ($result->num_rows > 0) {
10.     while($row = $result->fetch_assoc()) {
11.         $firstName = $row["firstName"];
12.     }
13. }
14. require('fpdf.php');
15. $pdf = new FPDF();
16. $pdf->AddPage();
17. $pdf->SetFont('Arial','B',16);
18. $topic = $_SESSION["topic"];
19. $student = "s".$_SESSION["studentID"];
20. $data = [];
21. $message = "Student Scores for ".$firstName;
22. $pdf->Cell(40,10,$message);
23. $pdf->Ln(10);
24. $pdf->SetFont('Arial','',12);

```

### Objective 39

Teacher able to print question, score and reason as a pdf

Roshni



```

25. $pdf->Cell(100,10,'Note: MCQ and Short Answer questions are all worth 1 mark, 1
ong questions are worth 2');
26. $pdf->Ln(10);
27.
28. //get all the test id,names, score and date for that student and put into an
array
29. $sql = "SELECT * FROM scoreBook INNER JOIN question ON scoreBook.questionID =
question.questionsID and question.topicID='$topic' and scoreBook.studentID='{$stude
nt}'";
30. $result = $conn->query($sql);
31. $count = 0;
32. if ($result->num_rows > 0) {
33.     while($row = $result->fetch_assoc()) {
34.         $count = $count + 1;
35.         $data[]=[
36.             'questions' =>$row["questions"],
37.             'score' => $row["score"],
38.             'reason'=> $row["Reason"]
39.         ];
40.     }
41. }
42. $pdf->Cell(100,10,'Question');
43. $pdf->Cell(20,10,'score');
44. $pdf->Cell(40,10,'Reason');
45. $pdf->Ln(5);
46. $pdf->SetFont('Arial','',8);
47. $pdf->Ln(10);
48. foreach($data as $key=>$row){
49.     $x = $pdf->GetX();
50.     $y = $pdf->GetY();

51.     $pdf -> Line(10, $y, $x+200, $y);
52.     $pdf->Ln(3);
53.     $pdf->MultiCell(100,3,$data[$key]['questions']);
54.     $x = $pdf->GetX();
55.     $y = $pdf->GetY();
56.     $pdf->SetXY($x + 100, $y-3);
57.     $pdf->Cell(20,3,$data[$key]['score']);
58.     $x = $pdf->GetX();
59.     $y = $pdf->GetY();
60.     $pdf->SetXY($x + 10, $y-3);
61.     $pdf->MultiCell(40,3,$data[$key]['reason']);
62.     $x = $pdf->GetX();
63.     $y = $pdf->GetY();
64.     $pdf->Ln(10);
65. }

66. $pdf->Output();
67. ?>

```

**Data structure**  
Adding elements to associative array

**Data structure – Printing an associative array**

*selectTest.php*



```
1. <?php //teacher to see scores for 1 topic on website for 1 student
2.     require "connection.php";
3.     $conn = connection::getConn();
4.     session_start();
5.     $id = $_SESSION["studentID"];
6.     $plainID = substr($id, 0);
7.     $sql = "SELECT * FROM `student` WHERE `studentID`=".$plainID;
8.     $result = $conn->query($sql);
9.     if ($result->num_rows > 0) {
10.         while($row = $result->fetch_assoc()) {
11.             $firstName = $row["firstName"];
12.         }
13.     }
```

```

14. $topic = $_POST["topicID"];
15. $_SESSION["topic"] = $topic;
16. $student = "s".$_SESSION["studentID"];
17. ?>
18. <html>
19.     <head>
20.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
21.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
22.         t" type="text/css">
23.         <style>
24.             body {font-family: "Raleway", Arial, sans-serif}
25.             .w3-row img {margin-bottom: -8px}
26.         </style>
27.     </head>
28.     <body style="background-color:#add8e6;">
29.         <header>
30.             <h1><b><center>Student Progress Tracker for <?php echo $firstName;?></c
31.             enter></b></h1>
32.         </header>
33.     </body>
34. </html>
35. <?php
36.     echo "Please click ".<a href="homepage.php">here</a>". to go back";
37.     echo "<br>";
38.     echo "<table border=1 frame=box rules=all>";
39.     //select and output the questions and scores, where topic ID is selected
40.     $sql = "SELECT * FROM scoreBook INNER JOIN question ON scoreBook.questionID =
41.     question.questionsID and question.topicID='$topic' and scoreBook.studentID='{ $stude
42.     nt}'";
43.     $result = $conn->query($sql);
44.     if ($result->num_rows > 0) {
45.         echo "Please click ".<a href="selectStudentTestPDF.php" target="_blank">he
46.         re</a>". to view as a pdf";
47.         echo "<br>";
48.         echo "Note: MCQ and Short Answer questions are all worth 1 mark, long quest
49.         ions are worth 2";
50.         echo "<tr><td width=50%>" . "<b>Question </b>".</td><td width=10%>".<b>
51.         Score</b>".</td><td width=20%>".<b>Reason</b>".</td></tr>";
52.         while($row = $result->fetch_assoc()) {
53.             echo "<tr><td width=50%>" . $row['questions'] . "</td><td width=10%>" .
54.             $row['score'] . "</td><td width=20%>" . $row['Reason'] . "</td></tr>";
55.         }
56.         echo "</table>";
57.     } else {
58.         echo "Student has not completed test in this topic";
59.     }
60. ?>

```

**SQL – Student Performance**  
on a particular topic

**Coding Style – Good use of global variables**

**Coding Style – Minimal use of global variables**

sort.php

```

1. <?php //sort out what type of question and difficulty the question will be
2.     session_start();
3.     if($_SESSION['userScoreData'][0]){ //ensures that user logged in
4.         if ($_SESSION['userScoreData'][1]<3){ //checks question count, if less than
5.             3 then MCQ
6.             $type = "m";
7.             if (sizeof($_SESSION['userScoreData'][9])>2){ //checking done more than
8.                 2 questions
9.                 $size = sizeof($_SESSION['userScoreData'][9]);
10.                if (($_SESSION['userScoreData'][9][$size-
11.                1]) == ($_SESSION['userScoreData'][9][$size-
12.                2])){ //checking if previous and previous one before that are equal
13.                    if ($_SESSION['userScoreData'][9][$size-
14.                    2]==1){ //if equal then both should be same number. check if 1, so therefore correc
15.                    t both

```

**Coding Styles – Annotation used effectively when required**

**High-Level Systems Overview-**  
Determining difficulty of the question

**Algorithm Design – If question number is**

### Objective 25

Question style vary

### Objective 34

Increase difficulty if 3 in row correct

### Objective 35

Decrease difficult if 3 in row incorrect

```

10.         if ($_SESSION['userScoreData'][8]==2){ //move up depending
on previous level
11.             $_SESSION['userScoreData'][8]=3;
12.         }
13.         if ($_SESSION['userScoreData'][8]==1){ //move up depending
on previous level
14.             $_SESSION['userScoreData'][8]=2;
15.         }
16.     }else{
17.         if ($_SESSION['userScoreData'][8]==2){ //move down
depending on previous level
18.             $_SESSION['userScoreData'][8]=1;
19.         }
20.         if ($_SESSION['userScoreData'][8]==3){ //move down
depending on previous level
21.             $_SESSION['userScoreData'][8]=2;
22.         }
23.     }
24. }
25. }
26. $locate = "individualPages/".$_SESSION['userScoreData'][0].".php";
27. }
28. elseif ($_SESSION['userScoreData'][1]>2 and $_SESSION['userScoreData'][1]<7
){ //checks question count, if less than 7 but more than 2 then short answer
29.     $type = "s";
30.     $size = sizeof($_SESSION['userScoreData'][9]);
31.     if (($_SESSION['userScoreData'][9][$size-
1]) == ($_SESSION['userScoreData'][9][$size-
2])){ //checking if previous and previous one before that are equal
32.         if ($_SESSION['userScoreData'][9][$size-
2]==1){ //if equal then both should be same number. check if 1, so therefore
correct both
33.             if ($_SESSION['userScoreData'][8]==2){ //move up depending on
previous level
34.                 $_SESSION['userScoreData'][8]=3;
35.             }
36.             if ($_SESSION['userScoreData'][8]==1){ //move up depending on
previous level
37.                 $_SESSION['userScoreData'][8]=2;
38.             }
39.         }else{
40.             if ($_SESSION['userScoreData'][8]==2){ //move down depending on
previous level
41.                 $_SESSION['userScoreData'][8]=1;
42.             }
43.             if ($_SESSION['userScoreData'][8]==3){ //move down depending on
previous level
44.                 $_SESSION['userScoreData'][8]=2;
45.             }
46.         }
47.     }
48.     $locate = "individualPages/".$_SESSION['userScoreData'][0].".php";
49. }
50. elseif ($_SESSION['userScoreData'][1]>6 and $_SESSION['userScoreData'][1]<13
){ //checks question count, if less than 13 but more than 7 then Long
51.     $type = "l";
52.     $size = sizeof($_SESSION['userScoreData'][9]);
53.     if (($_SESSION['userScoreData'][9][$size-
1]) == ($_SESSION['userScoreData'][9][$size-
2]) AND ($_SESSION['userScoreData'][9][$size-
3]) == ($_SESSION['userScoreData'][9][$size-
4]) AND ($_SESSION['userScoreData'][9][$size-
1]) == ($_SESSION['userScoreData'][9][$size-
4])){ //checking if previous and previous one before that are equal to the previous one before that and before that since long questions come in pairs

```

#### Algorithm

**Design - if**  
question  
number is  
more than  
2 and less  
than 7

#### Algorithm

**Design -**  
if  
question  
number is  
more than  
6 and less  
than 13

```

54.         if ($_SESSION['userScoreData'][9][$size-
2]==1){ //if equal then all should be same number. check if 1, so therefore correct
both
55.             if ($_SESSION['userScoreData'][8]==2){ //move up depending on
previous level
56.                 $_SESSION['userScoreData'][8]=3;
57.             }
58.             if ($_SESSION['userScoreData'][8]==1){ //move up depending on
previous level
59.                 $_SESSION['userScoreData'][8]=2;
60.             }
61.         }else{
62.             if ($_SESSION['userScoreData'][8]==2){ //move down depending on
previous level
63.                 $_SESSION['userScoreData'][8]=1;
64.             }
65.             if ($_SESSION['userScoreData'][8]==3){ //move down depending on
previous level
66.                 $_SESSION['userScoreData'][8]=2;
67.             }
68.         }
69.     }
70.     $locate = "individualPages/".$_SESSION['userScoreData'][0].".php";
71. }
72. else{
73.     $locate = "feedback.php";
74. }
75. $final = "Location: ".$locate."?type=".$type;
76. header($final);//takes them to page and type depending on question number
77. }else{
78.     echo "Wrong access";
79. }
80. ?>

```

**Coding Style – File paths parametrised**

**Algorithm Design –**  
else go to  
feedback  
page

**Coding Style – Good exception handling**

student.php

**Technique - Inheritance**

**Technique**  
Class

```

1. <?php
2. //child class of user class
3. //oop - inheritance
4. require "user.php";
5. class student extends user {
6.     private $firstName;
7.     private $surname;
8.     private $dob;
9.     private $school;
10.    private $username;
11.    private $pw;
12.    function construct ($firstName,$surname,$school,$dob,$username,$password)
13.    {
14.        parent:: construct($username,$password);
15.        $this->firstName = $firstName;
16.        $this->surname = $surname;
17.        $this->dob= $dob;
18.        $this->school=$school;
19.    }
20.    function getFirstName() {
21.        return $this->firstName;
22.    }
23.    function getSurname() {
24.        return $this->surname;
25.    }
26.    function getDOB() {
27.        return $this->dob;

```

**Coding Style Loosely**  
coupled modules

**Coding Style - Modules**

**Coding Style – Cohesive modules**

**Cod  
ing  
Styl  
e –  
Mo  
dul  
aris  
atio  
n of  
cod  
e**

```

27.     }

28.     function getSchool() {
29.         return $this->school;
30.     }
31.     function setFirstName($firstName) {
32.         $this->firstName = $firstName;
33.     }
34.     function setSurname($surname) {
35.         $this->surname=$surname;
36.     }
37.     function setDOB($dob) {
38.         $this->dob = $dob;
39.     }
40.     function setSchool($school) {
41.         $this->school = $school;
42.     }
43.
44. }
45. ?>

```

teacher.php

Technique - Inheritance

Technique  
Class

```

1. <?php
2. //child class of user class
3. //oop - inheritance
4. require "user.php";
5. class teacher extends user {
6.     private $firstName;
7.     private $surname;
8.     private $email;
9.     private $id;
10.    function construct ($username,$password,$firstName,$surname,$email) {
11.        parent:: construct($username,$password);
12.        $this->firstName = $firstName;
13.        $this->surname = $surname;
14.        $this->email=$email;
15.    }
16.    function getFirstName() {
17.        return $this->firstName;
18.    }
19.    function getSurname() {
20.        return $this->surname;
21.    }
22.    function getEmail() {
23.        return $this->email;
24.    }
25. }
26. ?>

```

topic.php

```

1. <html>
2. <head>
3. </head>
4. <body>
5. <!--
6. - form so that user can select the topic they want to check the progress of -->
7. <p>Please select the topic you want to view the student's progress of
8. <form action="selectTest.php" method="post">
9.     <select name="topicID">
10.         <option value="1">Addition</option>
11.         <option value="2">Subtraction</option>

```

```

11.         <option value="3">Numbers</option>
12.         <option value="4">Multiplication</option>
13.         <option value="5">Division</option>
14.         <option value="6">Mixed Operations</option>
15.         <option value="7">Sequences</option>
16.         <option value="8">Decimal</option>
17.         <option value="9">Ratios</option>
18.         <option value="10">Percentage</option>
19.         <option value="11">Time</option>
20.         <option value="12">Round and Estimate</option>
21.         <option value="13">Area of shapes</option>
22.         <option value="14">Perimeter of shapes</option>
23.         <option value="15">Probability</option>
24.         <option value="16">Factorisation</option>
25.         <option value="17">Substitution</option>
26.         <option value="18">Conversion of measurements</option>
27.         <option value="19">Rates and Speed</option>
28.     </select>
29.     <input type="submit">
30. </form>
31. </body>
32. </html>

```

user.php

**Technique**  
Class

```

1. <?php
2.     //parent class of student, parent and teacher class
3.     //oop - inheritance
4.     class user {
5.         private $username;
6.         private $password;
7.
8.         function construct ($username,$password) {
9.             $this->username = $username;
10.            $this->password = $password;
11.        }
12.        function getUsername() {
13.            return $this->username;
14.        }
15.        function getPassword() {
16.            return $this->password;
17.        }
18.        function setPassword($password) {
19.            $this->password = $password;
20.        }
21.        function setUsername($username) {
22.            $this->username = $username;
23.        }
24.    }
25. ?>

```

**Coding Style – Modularisation of code**

**Coding Style –**  
Loosely coupled  
modules

addition.php

## Coding Styles – Annotation used effectively when required

```

1. <?php //addition topic question generator
2.     session_start();
3.     //requiring all my classes that I will need
4.     require "connection.php";
5.     require "assignment.php";
6.     require "number.php";
7.     require "string.php";
8.     require "titlePage.php";
9.     //creating an instance of object
10.    $conn = connection::getConn();
11.    $s = new assignment;
12.    $num = new number;
13.    $string = new stringQuestion;

14.    // $type is given through a get value sent from the sort page which determines
    the type of question
15.    $type = $_GET['type'];
16.    // $question is calling a function in the class assignment, which has topic ID
    1, and user difficulty of $_SESSION['userScoreData'][8]
17.    $question=$s->question(1,$type,$_SESSION['userScoreData'][8]);
18.    // $id is the ID of the question that was selected
19.    $id= $s->id($question);
20.    if ($type == "m"){

21.        //gets two random numbers, one between 1 and 50, and one between 10 and 100
        and replaces them in $question where it says n1 and n2
22.        //the second calculation is between 10 and 100 so that for the options in
        the mcq, none will be less than 0
23.        $number1 = $num->getRandomNumber(1,50);
24.        $number2=$num->getRandomNumber(10,100);

25.        $updated1 = $string->getString("n1",$number1,$question);
26.        $updated2 = $string ->getString("n2",$number2,$updated1);
27.        //calculates the answer
28.        $answer = ($num->addition($number1,$number2));
29.        //provides a solution
30.        $sol=$number1." + ".$number2;
31.        echo "<h3 align=center>".$updated2." </h3> ";
32.        //calculates the options
33.        $optOne = $num->optionOne($answer);
34.        $optTwo = $num->optionTwo($answer);
35.        //session variable storing the important information about the question -
        the final question with the numbers, answer, solution and the two other options in
        the mcq
36.        $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
37.        //this file randomizes the order of all the options
38.        require "mcqOptionOrder.php";
39.        //this file displays it in an mcq format
40.        require "mcqDisplay.php";
41.    }elseif ($type == "s"){
42.        //gets three random numbers, all between 20 and 100 and replaces them in
        $question where it says n1, n2 and n3
43.        $number1 = $num->getRandomNumber(20,100);
44.        //to give the question some variation, different objects are used and one
        is randomly selected
45.        $objects =array("apple","banana","kiwi","orange","strawberry","pineapple");
46.        $randIndex = array_rand($objects);
47.        $object = $objects[$randIndex];
48.        //the variables of n1,n2,n3 and so1 in the question are replaced with the
        numbers and object chosen
49.        $updated1 = $string->getString("n1",$number1,$question);
50.        $updated2 = str_replace("so1", "$object", $updated1);
51.        $number2=$num->getRandomNumber(20,100);
52.        $number3=$num->getRandomNumber(20,100);
53.        $updated3 = $string ->getString("n2",$number2,$updated2);
54.        $updated4 = $string ->getString("n3",$number3,$updated3);

```

Technique- Encapsulation

**Algorithm Design –**  
instantiate,  
call and get

**Coding Style**  
Meaningful  
Identifier  
Names

Coding Style – Self Documenting Code

**String  
Replace  
Function**

Coding Style – Consistent style throughout

**Algorithm Design –**  
if type =  
multiple  
choice

**Algorithm Design**  
if type =  
short



Roshni

answer question

```

55.     echo "<br>";
56. echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>". $updated4. "</h4> ";
57.     //answer and solution dependant on the ID of the question
58.     if ($id==7 or $id==88 or $id==87){
59.         $answer = ($num->addition($number3,($num->addition($number1,$number2)))
    );
60.         $sol = "$number1." + ".$number2." + ".$number3";
61.     }
62.     if ($id==10){
63.         $answer = ($num->addition($number1,($num->addition($number1,$number1)))
    );
64.         $sol = "$number1." + ".$number1." + ".$number1";
65.     }
66.     //session variable stores the important information about the question - so
    the question, answer and solution
67.     $_SESSION['information']=array($updated4,$answer,$sol);
68.     //file used to display the question in short answer question format
69.     require "generalShortNumberForm.php";
70. }elseif($type == "1"){
71.     //gets three random numbers, all between 10 and 100 and replaces them in
    $question where it says n1, n2 and n3 and replaces these in the question
72.     $number1 = $num->getRandomNumber(10,100);
73.     $number2=$num->getRandomNumber(10,100);
74.     $number3=$num->getRandomNumber(10,100);
75.     $updated1 = $string->getString("n1",$number1,$question);
76.     $updated2 = $string->getString("n2",$number2,$updated1);
77.     $updated3 = $string->getString("n3",$number3,$updated2);
78.     echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>". $updated3. "</h4> ";
79.     //calculates answer and solution for each part of the question
80.     $answerOne = ($num->addition($number1,$number2));
81.     $solA = $number1." + ".$number2;
82.     $answerTwo = ($num->addition($answerOne,$number3));
83.     $solB=$number1." + ".$number2." + ".$number3;
84.     //session variable stores the important information of the question, answer
    to part a, solution to part a, answer to part b and solution to part b
85.     $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo,$solB)
    ;
86.     //file that displays the question in long answer question format
87.     require "generalNumberLongForm.php";
88. }
89. ?>

```

**Algorithm Design**  
if type =  
long  
answer  
question

area.php

```

1. <?php //area topic question generator
2.     session_start();
3.     //requiring all my classes that I will need
4.     require "connection.php";
5.     require "assignment.php";
6.     require "number.php";
7.     require "string.php";
8.     require "titlePage.php";
9.     //creating an instance of an object
10.    $conn = connection::getConn();
11.    $s = new assignment;
12.    $num = new number;
13.    $string = new stringQuestion;
14.    // $type is given through a get value sent from the sort page which determines
    the type of question
15.    $type = $_GET['type'];
16.    // $question is calling a function in the class assignment, which has topic ID
    1, and user difficulty of $_SESSION['userScoreData'][8]
17.    $question=$s->question(13,$type,$_SESSION['userScoreData'][8]);

```

```

18. // $id is the ID of the question that was selected
19. $id= $s->id($question);
20. //gets three random numbers, all between 10 and 100 and replaces them in
    $question where it says n1, n2 and n3 and replaces these in the question
21. $number1 = $num->getRandomNumber(10,100);
22. $number2=$num->getRandomNumber(10,100);
23. $number3=$num->getRandomNumber(10,100);
24. $updated1 = $string->getString("n1",$number1,$question);
25. $updated2 = $string ->getString("n2",$number2,$updated1);
26. $updated3 = $string ->getString("n3",$number3,$updated2);
27.
28. if ($type == "m"){
29.     //calculates answer, solution and options depending on the ID
    (As it is for different shapes)
30.     if ($id==26){
31.         $answer = $num->multiply($number1,$number1);
32.         if ($number1 !=2){$optOne = $num->optionOneTwoNumbers($number1,$number1
    );} //to make sure that none of the options are the same as the answer
33.         else {$optOne = $num->optionOneTwoNumbers($number1,$number1+1);}
34.         $optTwo = (int)($num->optionOneTwoNumbers($number1,(rand(2,10))));
35.         $sol=$number1. " x ". $number1;
36.     }
37.     if ($id==27){
38.         $answer = ( $num->multiply(0.5,($num->multiply($number1,$number2))));
39.         if ($answer != (int)($answer)){ //to make sure that the numbers
    generated do not result in a decimal number
40.             while ($answer != (int)($answer)){
41.                 $number1 = $num->getRandomNumber(10,100);
42.                 $number2=$num->getRandomNumber(10,100);
43.                 $answer = ( $num->multiply(0.5,($num->multiply($number1,$number
    2))));
44.             }
45.             $updated1 = $string->getString("n1",$number1,$question);
46.             $updated3 = $string ->getString("n2",$number2,$updated1);
47.         }
48.         $optOne = (int)( $num->multiply(0.5,($num->multiply($number1,($number2+
    6))));
49.         $optTwo = (int)( $num->multiply(0.5,($num->multiply($number1,$number2+3
    ))));
50.         $sol="0.5 x ".$number1. " x ". "$number2";
51.     }
52.     if ($id==29){
53.         $answer = ($num->multiply(0.5,($num->multiply(($num->addition($number2,
    $number3)),($number1))));
54.         $optOne = (int)( $num->multiply(0.5,($num->multiply(($num->addition($nu
    mber1,$number3)),($number2))));
55.         $optTwo = (int)($num->multiply(0.5,($num->multiply(($num->addition($num
    ber2,$number3)),($number1)))+1);
56.         $sol="0.5 x ( ".$number2." + ".$number3." ) x ".$number1."";
57.     }
58.     echo "<br>";
59.     echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>".$updated3." </h4> ";
60.     //session variable stores necessary information for question - question,
    answer, sol and the options
61.     $_SESSION['information']=array($updated3,$answer,$sol,$optOne,$optTwo);
62.     require "mcqOptionOrder.php";
63.     require "mcqDisplay.php";
64. }elseif ($type == "s"){
65.     //calculates answer and solution depending on ID number
66.     if ($id==51){
67.         $answer = $num->multiply($number1,$number1);
68.         $sol=$number1. " x ". $number1;
69.     }
70.     if ($id==52){

```

```

71.         $answer = ( $num->multiply(0.5,($num->multiply($number1,$number2))));
72.         $sol="0.5 x ".$number1." x ".$number2;
73.     }
74.     if ($id==53){
75.         $answer = ( $num->multiply(0.5,($num->multiply(($num->addition($number2
, $number3)),($number1))));
76.         $sol="(0.5 x ( ".$number2." + ".$number3." ) x ".$number1." )";
77.     }
78.     echo "<br>";
79. echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated3."</h4> ";
80.     $_SESSION['information']=array($updated3,$answer,$sol);
81.     require "generalShortNumberForm.php";
82.     }elseif($type == "1"){
83. //calculates answer for part a, sol for part a, answer for part b, sol for part b
depending on
84.     if ($id == 72){
85.         $answerOne=$num->multiply($number1,$number1);
86.         $solA=$number1." X ".$number1;
87.         $answerTwo = ( $num->multiply($number1,($num->multiply($number1,2))));
88.         $solB= "(".$number1." X 2) X ".$number1."";
89.     }
90.     if($id == 91){
91.         $answerOne = $num->multiply($number1,$number1);
92.         $solA=$number1." X ".$number1;
93. $answerTwo = $num->subtraction(($num->multiply($number2,$number2)),$ans werOne);
94. $solB = "First work out the area of the big square - ".$number1." X ".$ number1."
Then subtract answer one from that area";
95.     }
96.     if ($id ==92){
97.         $answerOne = $num->multiply($number1,$number2);
98.         $solA=$number1." X ".$number2;
99.         $answerTwo = $num->multiply($answerOne,$number3);
100.        $solB = "Area of room is ".$answerOne." and so you multiply this
with how much it costs per unit squared which is ".$number3;
101.    }
102.    echo "<br>";
103.    echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated3."</h4> ";
104.    $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo
,$solB);
105.    require "generalNumberLongForm.php";
106.    }
107.    ?>

```

assignment.php

Technique - Class

```

1. <?php //oop - class
2.     class assignment{
3.         private $s;
4.         private $question;
5.         function getQuestion(){
6.             return $this->question;
7.         }
8.         function __construct () {
9.             //aggregation sql object
10.            require "sql.php";
11.            $this->s = new sql;
12.        }
13.        function question($topicID,$type,$level=2){
14.            return $this->s->getQuestionFromDB($topicID,$type,$level);
15.        }
16.        function id($question){

```

Technique - Composition

Coding Style – Modules with appropriate interfaces

High-Level Systems Overview – Getting a question from the database

```

17.         return $this->s->getQuestionIDFromDB($question);
18.     }
19. }
20. ?>

```

*backToSort.php*

```

1. <?php
2. //redirects student back to the sort page
3. ?>
4. <html>
5.     <body>
6.         <form style="text-align:center" name="form1" action = "../sort.php">
7.             <input type = "Submit" Name = "Submit1" VALUE = "Next Question">
8.         </form>
9.     </body>
10. </html>

```

*connection.php*

Technique - Class

```

1. <?php
2.     class connection //this class allows pages to connect to the db
3.     {
4.         public static function getConn()
5.         {
6.             $servername = "localhost";
7.             $username = "al2020-13";
8.             $password = "36*M5cxz";
9.             $dbname = "al2020-13";
10.            // Create connection
11.            $conn = new mysqli($servername, $username, $password, $dbname);
12.            if ($conn->connect_error) {
13.                die("Connection failed: " . $conn->connect_error);
14.            }
15.            return $conn;
16.        }
17.    }
18. ?>

```

*conversion.php*

```

1. <?php //conversion topic question generator
2.     function genUnit(){
3.         //function to generate a random unit
4.         $unit=array("mm","cm","m");
5.         $randIndex = array_rand($unit);
6.         return $unit[$randIndex];
7.     }
8.     session_start();
9.     //requiring all my classes that I will need
10.    require "connection.php";
11.    require "assignment.php";
12.    require "number.php";
13.    require "string.php";
14.    require "titlePage.php";
15.    //creating an instance of an object
16.    $conn = connection::getConn();
17.    $s = new assignment;
18.    $num = new number;
19.    $string = new stringQuestion;

```

```

20. // $type is given through a get value sent from the sort page which determines
    the type of question
21. $type = $_GET['type'];
22. // $question is calling a function in the class assignment, which has topic ID
    18, and user difficulty of $_SESSION['userScoreData'][8]
23. $question=$s->question(18,$type,$_SESSION['userScoreData'][8]);
24. // $id is the ID of the question that was selected
25. $id= $s->id($question);
26. // gets a random number between the range
27. $number1 = $num->getRandomNumber(1000000,999000000);
28. // gets a unit
29. $unit1 = genUnit();
30. $unit2 = "km";
31. // replaces question with the variables calculated
32. $updated1 = $string->getString("n1",$number1,$question);
33. $updated2 = $string->getString("u2", $unit2, $updated1);
34. $updated3 = $string->getString("u1", "$unit1", $updated2);
35. if ($type == "m"){
36.     echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;>".$updated3."</h4> ";
37.     // calculates answer and sol based on unit
38.     if ($unit1=="cm"){
39.         $answer = $num->division($number1,100000);
40.         $sol= $number1." / 100000";
41.     }
42.     if ($unit1=="mm"){
43.         $answer = $num->division($number1,1000000);
44.         $sol= $number1." / 1000000";
45.     }
46.     if ($unit1=="m"){
47.         $answer = $num->division($number1,1000);
48.         $sol= $number1." / 1000";
49.     }
50.     $optOne = $num->multiply($answer,10);
51.     $optTwo = $num->multiply($answer,100);
52.     $_SESSION['information']=array($updated3,$answer,$sol,$optOne,$optTwo);
53.     require "mcqOptionOrder.php";
54.     require "mcqDisplay.php";
55. }elseif ($type == "s"){
56. echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;>".$updated3."</h4> ";
57.     // calculates answer and sol based on units
58.     if ($unit1=="cm"){
59.         $answer = $num->division($number1,100000);
60.         $sol= $number1." / 100000";
61.     }
62.     if ($unit1=="mm"){
63.         $answer = $num->division($number1,1000000);
64.         $sol= $number1." / 1000000";
65.     }
66.     if ($unit1=="m"){
67.         $answer = $num->division($number1,1000);
68.         $sol= $number1." / 1000";
69.     }
70.     $_SESSION['information']=array($updated3,$answer,$sol);
71.     require "generalShortStringForm.php";
72. }elseif($type == "l"){
73. // generates two random numbers between the two ranges and updated the
    question
74.     $number2=$num->getRandomNumber(1000000,999000000);
75.     $number3=$num->getRandomNumber(1000000,999000000);
76.     $updated4 = $string->getString("n2",$number2,$updated3);
77.     $updated3 = $string->getString("n3",$number3,$updated4);
78. echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;>".$updated3."</h4> ";
79.     // calculates the answer a, sol a, answer b, sol b depending on unit

```

**Coding Style – Use of constants**

```

80.         if ($unit1=="cm"){
81.             $answerOne =$num->division($number1,100000);
82.             $numTwoConv =$num->division($number2,100000);
83.             $numberThreeConv = $num->division($number3,100000);
84.             $answerTwo = $num->multiply($numTwoConv,$numberThreeConv);
85.             $solA=$number1." / 100000";
86.             $solB="(".$number2." / 100000) X (".$number3." / 100000 )";
87.         }
88.         if ($unit1=="mm"){
89.             $answerOne = $num->division($number1,1000000);
90.             $numTwoConv = $num->division($number2,1000000);
91.             $numberThreeConv = $num->division($number3,1000000);
92.             $answerTwo = $num->multiply($numTwoConv,$numberThreeConv);
93.             $solA=$number1." / 1000000";
94.             $solB="(".$number2." / 1000000) X (".$number3." / 1000000 )";
95.         }
96.         if ($unit1=="m"){
97.             $answerOne = $num->division($number1,1000);
98.             $numTwoConv = $num->division($number2,1000);
99.             $numberThreeConv = $num->division($number3,1000);
100.             $answerTwo = $num->multiply($numTwoConv,$numberThreeConv);
101.             $solA=$number1." / 1000";
102.             $solB="(".$number2." / 1000) X (".$number3." / 1000 )";
103.         }
104.         $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo
,$solB);
105.         require "generalStringLongForm.php";
106.     }
107.     ?>

```

decimal.php

```

1.  <?php //decimal topic question generator
2.      function genDenominator(){
3.          //child friendly denominator - one chosen at random
4.          $array = array(10,50,100);
5.          shuffle ($array);
6.          return $array[0]; 7.
    }
8.      session_start();
9.      require "connection.php";
10.     require "assignment.php";
11.     require "number.php";
12.     require "string.php";
13.     require "titlePage.php";
14.     $conn = connection::getConn();
15.     $s = new assignment;
16.     $num = new number;
17.     $string = new stringQuestion;
18.     //$type is given through a get value sent from the sort page which determines the
    type of question
19.     $type = $_GET['type'];
20.     //$question is calling a function in the class assignment, which has topic ID 8,
    and user difficulty of $_SESSION['userScoreData'][8]
21.     $question=$s->question(8,$type,$_SESSION['userScoreData'][8]);
22.     //$id is the ID of the question that was selected
23.     $id= $s->id($question);
24.     if ($type == "m"){
25.         //generates a fraction that is easy to convert to decimal
26.         $number1 = $num->getRandomNumber(1,50);
27.         $number2 = genDenominator();
28.         $updated1 = $string->getString("n1",$number1,$question);
29.         $updated2 = $string ->getString("n2",$number2,$updated1);
30.         $frac1 = $number1."/".$number2;
31.         $updated3 = $string->getString("f1", "$frac1", $updated2);

```

```

32.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3." </h4> ";
33.         //calculates answer, and options
34.         $answer = $num->division($number1,$number2);
35.         $optOne = $num->multiply($answer,10);
36.         $optTwo = $num->division($answer,10);
37.         //calculates solution based on denominator
38.         if ($number2 == 10){
39.             $sol = "Move the decimal point of ".$number1." one place to the left"
;
40.         }
41.         if ($number2 == 50){
42.             $sol = "Divide ".$number1." by 2";
43.         }
44.         if ($number2 == 100){
45.             $sol = "Move the decimal point of ".$number1." two places to the left
";
46.         }
47.         $_SESSION['information']=array($updated3,$answer,$sol,$optOne,$optTwo);
48.         require "mcqOptionOrder.php";
49.         require "mcqDisplay.php";
50.     }elseif ($type == "s"){
51.         //calculates random numbers between 20 and 150
52.         $number1 = $num->getRandomNumber(20,150);
53.         $number2=$num->getRandomNumber(20,150);
54.         if ($id == 95){
55.             $answerOne = $num->division($number2,($num->addition($number1,$number2)
));
56.             if (strlen(substr(strrchr($answerOne, "."), 1))!=1){
57.                 while ((strlen(substr(strrchr($answerOne, "."), 1))!=1)){
58.                     $number1 = $num->getRandomNumber(20,150);
59.                     $number2=$num->getRandomNumber(20,150);
60.                     $answerOne =$num->division($number2,($num->addition($number1,$n
umber2)));
61.                 }
62.             }
63.             $answer = $num->division($number2,($num->addition($number1,$number2)));
64.             $sol = $number2. " is the total number of people in red. So, you divide
this with the total number of people in the pub.";
65.         }else{
66.             $answer = $num->division(($num->addition($number1,$number2)),100);
67.             $sol= "(".$number2." + ".$number1.")" / 100 ";
68.         }
69.         $updated1 = $string->getString("n1",$number1,$question);
70.         $updated2 = $string->getString("n2",$number2,$updated1);
71.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated2." </h4> ";
72.         $_SESSION['information']=array($updated2,$answer,$sol);
73.         require "generalShortStringForm.php";
74.     }elseif($type == "l"){
75.         //generates random numbers between the ranges and replaces it in the
$question
76.         $number1 = $num->getRandomNumber(1000,1000000);
77.         $number1 = $num->division($number1,100);
78.         $number2=$num->getRandomNumber(1000,1000000);
79.         $number2 = $num->division($number2,100);
80.         $updated1 = $string->getString("n1",$number1,$question);
81.         $updated2 = $string->getString("n2",$number2,$updated1);
82.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated2." </h4> ";
83.         //calculates answer a,sol a, answer b, sol b according to ID
84.         if ($id==96){
85.             $answerOne = ($num->addition(($num->addition($number1,$number2)),($num-
>addition($number1,$number2)));

```



```

86.         $solA = $number1." + ".$number2 ." + ".$number1." + ".$number2;
87.         $answerTwo = $num->multiply($number1,$number2);
88.         $solB = $number1." * ".$number2;
89.     }elseif ($id == 97){
90.         if ($number1>$number2){
91.             $answerOne = $number1;
92.             $solA = "Compare the two numbers. The first number is bigger";
93.         }else{
94.             $answerOne = $number2;
95.             $solA = "Compare the two numbers. The second number is bigger";
96.         }
97.         $answerTwo = $num->addition($number1,$number2);
98.         $solB = $number1." + ".$number2;
99.     }else{
100.         $answerOne = $num->addition($number1,$number2);
101.         $solA = $number1." + ".$number2;
102.         $answerTwo = $num->addition($answerOne,$number1);
103.         $solB = $number1." + ".$number2." + ".$number1;
104.     }
105.     $_SESSION['information']=array($updated2,$answerOne,$solA,$answerTwo
,$solB);
106.     require "generalStringLongForm.php";
107. }
108. ?>

```

division.php

```

1. <?php //division topic question generator
2. session_start();
3. require "connection.php";
4. require "assignment.php";
5. require "number.php";
6. require "string.php";
7. require "titlePage.php";
8. $conn = connection::getConn();
9. $s = new assignment;
10. $num = new number;
11. $string = new stringQuestion;
12. // $type is given through a get value sent from the sort page which determines
the type of question
13. $type = $_GET['type'];
14. // $question is calling a function in the class assignment, which has topic ID
5, and user difficulty of $_SESSION['userScoreData'][8]
15. $question=$s->question(5,$type,$_SESSION['userScoreData'][8]);
16. // $id is the ID of the question that was selected
17. $id= $s->id($question);
18. if ($type == "m"){
19.     $number1 = $num->getRandomNumber(2,100);
20.     $number2=$num->getRandomNumber(2,100);
21.     if (($num->intDiv($number1 , $number2))==false){ //make sure answer
is integer
22.         while (($num->intDiv($number1 , $number2))==false) { //if not
integer, then regenerates numbers until answer is an integer
23.             $number1 = $num->getRandomNumber(2,100);
24.             $number2=$num->getRandomNumber(2,100);
25.         }
26.     }
27.     $updated1 = $string->getString("n1",$number1,$question);
28.     $updated2 = $string->getString("n2",$number2,$updated1);
29.     $answer = $num->division($number1,$number2);
30.     $sol=$number1." / ".$number2;
31.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated2." </h4> ";
32.     echo "<br>";
33.     //calculates the options - makes sure both are positive

```

**Coding Style**  
Appropriate  
indentation

**Technique –**  
Optimisation  
of numbers

```

34.     $optOne = $num->optionOne($answer);
35.     $optTwo = $num->optionTwo($answer);
36.     $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
37.     require "mcqOptionOrder.php";
38.     require "mcqDisplay.php";
39. }elseif ($type == "s"){
40.     $number1 = $num->getRandomNumber(20,10000);
41.     $number2=$num->getRandomNumber(20,100900);
42.     if (($num->division($number2,$number1))<1){//makes sure when divide it is
greater than 1
43.         while(($num->division($number2,$number1))<1){ //regenerates until great
er than 1
44.             if (($num->intDiv($number2 , $number1))==false){ //makes sure it
divides to an integer
45.                 while (($num->intDiv($number2 , $number1))==false) { //if not,
regenerates numbers until both integer and greater than 1 (since it is a nested if)
46.                     $number1 = $num->getRandomNumber(20,10000);
47.                     $number2=$num->getRandomNumber(20,10000);
48.                 }
49.             }
50.         }
51.     }elseif ($type == "l"){
52.         if (($num->intDiv($number2 , $number1))==false){ //makes sure integer
result
53.             while (($num->intDiv($number2 , $number1))==false) { //regenerates
until positive
54.                 $number1 = $num->getRandomNumber(20,10000);
55.                 $number2=$num->getRandomNumber(20,10000);
56.             }
57.         }
58.     }
59.     $updated1 = $string->getString("n2",$number1,$question);
60.     $updated2 = $string->getString("n1",$number2,$updated1);
61.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $updated2." </h4> ";
62.     $answer = $num->division($number2,$number1);
63.     $sol = "$number2." / "."$number1";
64.     $_SESSION['information']=array($updated2,$answer,$sol);
65.     require "generalShortNumberForm.php";
66. }elseif($type == "l"){
67.     //randomly generates 3 random numbers
68.     $number1 = $num->getRandomNumber(10,100000);
69.     $number2=$num->getRandomNumber(10,1000);
70.     $number3=$num->getRandomNumber(3,100);
71.     if (($num->division(($num->division($number1,$number2)), $number3))<2){
//if division of (1/2)/3 less than 2, then regenerates until greater than 2
72.         while (($num->division(($num->division($number1,$number2)), $number3))<
2){
73.             if( (($num->intDiv($number1 , $number2))==false) or ($num->intDiv((
$number1 / $number2) , $number3)==false)){ //if division of (1/2)/3 not ineger,
regenerates until satisfies this condition
74.                 while( (($num->intDiv($number1 , $number2))==false) or ($num->i
ntDiv(($number1 / $number2) , $number3)==false)) {
75.                     $number1 = $num->getRandomNumber(10,100000);
76.                     $number2=$num->getRandomNumber(10,1000);
77.                     $number3=$num->getRandomNumber(3,100);
78.                 }
79.             }
80.         }
81.     } else{
82.         if ( ( (($num->intDiv($number1 , $number2))==false) or ($num->intDiv((n
umber1 / $number2) , $number3)==false))) { //if division of (1/2)/3 not ineger, reg
enarates until satisfies this condition

```

```

83.         while ( (($num->intDiv($number1 , $number2))==false) or ($num->intD
iv(($number1 / $number2) , $number3)==false)) { //if division of (1/2)/3 not ineger,
regenerates until satisfies this condition
84.             $number1 = $num->getRandomNumber(10,100000);
85.             $number2=$num->getRandomNumber(10,1000);
86.             $number3=$num->getRandomNumber(3,100);
87.         }
88.     }
89. }
90. $updated1 = $string->getString("n1",$number1,$question);
91. $updated2 = $string ->getString("n2",$number2,$updated1);
92. $updated3 = $string ->getString("n3",$number3,$updated2);
93. echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3." </h4> ";
94. $answerOne = $num->division($number1,$number2);
95. $solA= $number1." / ".$number2;
96. $answerTwo = $num->division(($num->division($number1,$number2)), $number3);
97. $solB = ($number1 / $number2)." / ".$number3;
98. $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo,$solB)
;
99.     require "generalNumberLongForm.php";
100. }
101. ?>

```

factorise.php

```

1. <?php //factorise topic question generator
2. session_start();
3. require "connection.php";
4. require "assignment.php";
5. require "number.php";
6. require "string.php";
7. require "titlePage.php";
8. $conn = connection::getConn();
9. $s = new assignment;
10. $num = new number;
11. $string = new stringQuestion;
12. // $type is given through a get value sent from the sort page which determines
the type of question
13. $type = $_GET['type'];
14. // $question is calling a function in the class assignment, which has topic ID
16, and user difficulty of $_SESSION['userScoreData'][8]
15. $questionDB=$s->question(16,$type,$_SESSION['userScoreData'][8]);
16. //generates unique numbers according to difficulty level
17. if ($_SESSION['userScoreData'][8]==2){
18.     34.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.

```

**Algorithm Design**  
instantiate , call and get

```

$m = $num->getRandomNumber(-10,10);
$n=$num->getRandomNumber(-10,10);
if ($m == $n){
    while ($m == $n){
        $m = $num->getRandomNumber(-
10,10);
        $n=$num->getRandomNumber(-
10,10);
    }
}

```

```

} else if
($_SESSION['userScoreData'][8]==1){
    $m = $num->getRandomNumber(2,10);
    $n=$num->getRandomNumber(2,10);
    if ($m == $n){
        while ($m == $n){
            $m = $num-

```

**Technique Optimisation**  
of numbers

Roshni

```
> A ween -10
get I and 10
t g
R o
a r
n i
d Algorithm Design – If
o difficulty is 1, generate
m t numbers between 2 and 10
N h
u m
m D
b e
e s
r i
( g
2 n
, –
1 l
0 f
) d
; i
$ n f
= $ i
n u f
m f
- i
> c
get u
t l
R t
a y
n i
d s
o 2
m ,
N g
u e
m n
b e
r r
a
( t
2 e
, n
1 r
0 a
) t
; e
} n
} u
m
b
e
r
s
b
e
t
```

```

35.     }else if ($_SESSION['userScoreData'][8]==3){
36.         $m = $num->getRandomNumber(-20,20);
37.         $n=$num->getRandomNumber(-20,20);
38.         if ($m == $n){
39.             while ($m == $n){
40.                 $m = $num->getRandomNumber(-20,20);
41.                 $n=$num->getRandomNumber(-20,20);
42.             }
43.         }

44.     }
45.     //calculates product and sum in order to write question
46.     $b = ($num->addition($n,$m));
47.     $c = $num->multiply($n,$m);
48.     //writes question based on signs
49.     if($b<=0 and $c>0){
50.         $question = "x^2 ".$b."x + ".$c."=0";
51.     }else if ($b>=0 and $c<0){
52.         $question = "x^2 + ".$b."x ".$c;
53.     }else if ($b<=0 and $c<0){
54.         $question = "x^2 ".$b."x ".$c;
55.     }else{
56.         $question = "x^2 + ".$b."x + ".$c;
57.     }
58.     $longQuestion = $questionDB. " ".$question;
59.     if ($type == "m"){
60.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>".$longQuestion." </h4> ";

61.         //calculates the solutions depending on the signs of each of the roots
62.         if (abs($n)==$n){
63.             $ansn = $num->multiply($n,-1);
64.         }
65.         else if ((abs($n)!=-$n)){
66.             $ansn = abs($n);
67.         }
68.         if (abs($m)==$m){
69.             $ansm = $num->multiply(-1,$m);
70.         }
71.         else if ((abs($m)!=-$m)){
72.             $ansm = abs($m);
73.         }
74.         $answer = "x=".$ansn." and x=".$ansm;
75.         $sol ="Find the common factors that multiply to make ".$c." but add to make ".$b." which would be ".$ansm." and ".$ansn;
76.         $optOne = "x=".(($ansn*-1))." and x=".(($c);
77.         $optTwo = "x=".$n." and x=".(($ansm*-1);
78.         $_SESSION['information']=array($longQuestion,$answer,$sol,$optOne,$optTwo);

79.         require "mcqOptionOrder.php";
80.         require "mcqDisplay.php";
81.     }elseif ($type == "s"){
82.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>".$longQuestion." </h4> ";
83.         //calculates answer depending on signs of roots
84.         //2 ways as brackets can be written in either order
85.         if ($m>0 and $n>0){
86.             $answerOne = "(x+ ".$n.") (x+ ".$m.")";
87.             $answerTwo = "(x+ ".$m.") (x+ ".$n.")";
88.         } else if ($m<0 and $n>0){
89.             $answerOne = "(x+ ".$n.") (x- ".$m.")";
90.             $answerTwo = "(x- ".$m.") (x+ ".$n.")";
91.         }else if ($m>0 and $n<0){
92.             $answerOne = "(x- ".$n.") (x+ ".$m.")";
93.             $answerTwo = "(x+ ".$m.") (x- ".$n.")";
94.         }elseif ($m==0){
95.             $answerOne = "(x- ".$n.") (x)";
96.             $answerTwo = "(x) (x- ".$n.")";

```

**Algorithm Design – If**  
difficulty is 3, generate  
numbers between -20  
and 20

**Algorithm Design – Write**  
the quadratic equation and  
the question

**Algorithm**  
**Design –**  
if the  
question  
type is  
multiple  
choice

**Algorithm**  
**Design –**  
if the  
question  
type is  
short  
answer

```

97.         }elseif ($n==0){
98.             $answerOne = "(x".$m.") (x)";
99.             $answerTwo = "(x)(x".$m.")";
100.         }else{
101.             $answerOne = "(x".$n.") (x".$m.")";
102.             $answerTwo = "(x".$m.") (x".$n.")";
103.         }
104.         $_SESSION['question']=$question;
105.         $_SESSION['sol']="Find the common factors that multiply to make ".$c
        ." but add to make ".$b." which would be ".$m." and ".$n." and then write the quadra
        tic in a factorise form" ;
106.         $_SESSION['answerOne']=$answerOne;
107.         $_SESSION['answerTwo']=$answerTwo;
108.         require "factoriseShortQuestionForm.php";
109.     }elseif($type == "1"){
110.         //calculates answer depending on signs of roots
111.         //2 ways as brackets can be written in either order
112.         if ($m>0 and $n>0){
113.             $answerOne = "(x+$.n.) (x+$.m.)";
114.             $answerTwo = "(x+$.m.) (x+$.n.)";
115.         } else if ($m<0 and $n>0){
116.             $answerOne = "(x+$.n.) (x".$m.")";
117.             $answerTwo = "(x".$m.") (x+$.n.)";
118.         }else if ($m>0 and $n<0){
119.             $answerOne = "(x".$n.") (x+$.m.)";
120.             $answerTwo = "(x+$.m.) (x".$n.)";
121.         }else{
122.             $answerOne = "(x".$n.") (x".$m.)";
123.             $answerTwo = "(x".$m.") (x".$n.)";
124.         }
125.         //calculates solutions based on signs
126.         if (abs($n)==$n){
127.             $ansn = $num->multiply($n,-1);
128.         }
129.         else if ((abs($n)!= $n)){
130.             $ansn = abs($n);
131.         }
132.         if (abs($m)==$m){
133.             $ansm = $num->multiply(-1,$m);
134.         }
135.         else if ((abs($m)!= $m)){
136.             $ansm = abs($m);
137.         }
138.         //can be in either order
139.         $answerA = "X= ".$ansn." and X= ".$ansm;
140.         $answerB = "X= ".$ansm." and X= ".$ansn;
141.         $updated = $string->getString("q1",$question,$questionDB);
142.         echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;'> ".$updated." </h4> ";
143.         $_SESSION['question']=$updated;
144.         $_SESSION['solA']="Find the common factors that multiply to make "
        .$c." but add to make ".$b." which would be ".$ansm." and ".$ansn." and hence use
        that knowledge to factorise the quadratic";
145.         $_SESSION['answerOneA']=$answerOne;
146.         $_SESSION['answerOneB']=$answerTwo;
147.         $_SESSION['solB']="Find the common factors that multiply to make "
        .$c." but add to make ".$b." which would be ".$ansm." and ".$ansn;
148.         $_SESSION['answerTwoA']=$answerA;
149.         $_SESSION['answerTwoB']=$answerB;
150.         require "factoriseLongQuestionForm.php";
151.     }
152.     ?>

```

**Algorithm Design –**  
if the question type is long answer question

*factoriseLongQuestionCheck.php*

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php //check the long answer question for factorisation (can be multiple ans)
14.     session_start();
15.     //question number increases by 2 (since 2 parts - a and b)
16.     $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+2;
17.     $quest = "Question was : ".$_SESSION['question'];
18.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>".$_quest."</h4> ";
19.     $reason = $_POST['sol'];
20.     echo "<br>";
21.     if (isset($_POST['Submit1'])) {
22.         //checks if wrote write answer for part a
23.         $selected_radio = $_POST['answerOne'];
24.         if ($selected_radio == $_SESSION['answerOneA'] or $selected_radio == $_SESSION['answerOneB']) { //if did, output as correct, displays green check box, increases total correct by 1 and adds 1 to array of scores
25.             $answerStatusOne = "correct";
26.             $_SESSION['userScoreData'][2]=$_SESSION['userScoreData'][2]+1;
27.             array_push($_SESSION['userScoreData'][9],1);
28.             $displayOne = "✔";
29.             $score = 1;
30.         }
31.         else {//if not, output as wrong, displays red cross box, increases total wrong by 1 and adds 0 to array of scores
32.             $answerStatusOne = "wrong";
33.             $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;
34.             array_push($_SESSION['userScoreData'][9],0);
35.             $displayOne = "✗";
36.             $score = 0;
37.         }
38.         $selected_radio_Two = $_POST['answerTwo'];
39.         if (($selected_radio_Two == $_SESSION['answerTwoA'] or $selected_radio_Two == $_SESSION['answerTwoB'])){//if did, output as correct, displays green check box , increases total correct by 1 and adds 1 to array of scores
40.             $answerStatusTwo = "correct";
41.             $_SESSION['userScoreData'][2]=$_SESSION['userScoreData'][2]+1;
42.             array_push($_SESSION['userScoreData'][9],1);
43.             $displayTwo = "✔";
44.             $score = $score + 1;
45.         }
46.         else {//if not, output as wrong, displays red corss box, increases total wrong by 1 and adds 0 to array of scores
47.             $answerStatusTwo = "wrong";
48.             $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;
49.             array_push($_SESSION['userScoreData'][9],0);
50.             $displayTwo = "✗";
51.         }
52.     }
53.     $answerChoA = "Your response for part a was: ".$selected_radio;

```

```

54.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $answerCho. " </h4> ";
55.     $answerChoARes = "This was : ". $answerStatusOne. " ";
56.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $answerChoARes. " </h4> ";
57.     require "connection.php";
58.     $conn = connection::getConn();
59.     $num = $_SESSION['userScoreData'][4];
60.     $question = $_SESSION["question"];
61.     //insert question into db
62.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('$question', $
um)";
63.     if ($conn->query($sql) == TRUE) {
64.     } else {
65.         echo "Connection error";
66.     }
67.     //gets question ID of what just inserted
68.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
69.     $result = $conn->query($sql);
70.     if ($conn->query($sql) == TRUE) {
71.         if ($result->num_rows > 0) {
72.             while($row = $result->fetch_assoc()) {
73.                 $qid = $row["questionsID"];
74.             } }else {
75.                 echo "Connection error";
76.             }
77.             $sid = $_SESSION['userData'][2];
78.             //inserts into score book the students score for this question
79.             $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`) V
ALUES ('$sid', '$qid', '$score', '$reason')";
80.             if ($conn->query($sql2) ==TRUE) {
81.             } else {
82.                 echo "Connection error";
83.             }
84.     }
85.     <html><center><span><?php echo $displayOne?></span></center></html>
86.     <?php
87.         if ($answerStatusOne=="wrong"){
88.             //if answer one wrong, displays correct answer and sol
89.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct answer was : ". $_SESSION['answerOneA']. " </h4> ";
90.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct method is: ". $_SESSION['solA']. " </h4> ";
91.         }
92.         echo "<br>";
93.         $answerChoB = "Your response for part b was: ". $selected_radio_Two;
94.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $answerChoB. " </h4> ";
95.         $answerChobRes = "This was : ". $answerStatusTwo. " ";
96.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $answerChoBRes. " </h4> ";
97.     }
98.     <html><center><span><?php echo $displayTwo?></span></center></html>
99.     <?php
100.         if ($answerStatusTwo=="wrong"){
101.             //if answer two wrong, displays correct answer and sol
102.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct answer was : ". $_SESSION['answerTwoA']. " </h4> ";
103.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct method is: ". $_SESSION['solB']. " </h4> ";
104.         }
105.         require "backToSort.php";
106.     }

```

SQL – Insert score for a question into database (part 1)



*factoriseLongQuestionForm.php*

```

1. <!DOCTYPE html>
2. <html> <!--Form for student to answer long answer question for factorisation-->
3.     <body style="background-color:#add8e6;">
4.         <div style="text-align: center;">
5.             <div style="display: inline-block; text-align: left;">
6.                 <form name = "form1" method = "post" action = "factoriseLongQuestionCh
eck.php">
7.                     <br>
8.                     <p>Part A</p>
9.                     <input type="text" name="answerOne" required/>
10.                    <p></p>
11.                    <p>Part B</p>
12.                    <input type="text" name="answerTwo" required/>
13.                    <p></p>
14.                    <p>Reasoning for answers</p>
15.                    <input type="text" name="sol" />
16.                    <p></p>
17.                    <input type = "Submit" Name = "Submit1" VALUE = "Submit"/>
18.                </form>
19.            </div>
20.        </div>
21.    </body>
22. </html>

```

**Coding Style – Defensive Programming**

*factoriseShortQuestionCheck.php*

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php //check student response for factorise short question as multiple ans allowed
14.     session_start();
15.     $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+1;
16.     $quest = "The question was : ".$_SESSION['information'][0];
17.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $quest." </h4> ";
18.     $reason = $_POST['sol'];
19.     if (isset($_POST['Submit1'])) {
20.         $selected_radio = $_POST['answer'];
21.         if ($selected_radio == $_SESSION['answerOne'] or $selected_radio == $_SESSI
ON['answerTwo'] ) { //checks answer written was correct, if so displays correct,
green tick box, increases total correct by 1 and adds 1 to array of scores
22.             $answerStatus = "correct";
23.             $_SESSION['userScoreData'][2]=$_SESSION['userScoreData'][2]+1;
24.             array_push($_SESSION['userScoreData'][9],1);
25.             $display = "☑";
26.             $score = 1;
27.         }
28.         else { //displays wrong, red cross box, increases total wrong by 1 and adds
0 to array of scores
29.             $answerStatus = "wrong";
30.             $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;

```

**Data Structure – Add to an array**

```

31.         array_push($_SESSION['userScoreData'][9],0);
32.         $display = "X";
33.         $score = 0;
34.     }
35.
36. }
37.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "Your response was ".$selected_radio." </h4> ";
38.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "This was : ".$answerStatus." </h4> ";
39.     require "connection.php";
40.     $conn = connection::getConn();
41.     $num = $_SESSION['userScoreData'][4];
42.     $question = "Factorise the following ".$_SESSION["question"];
43.     //inserts into db the question
44.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('$question', $n
um)";
45.     if ($conn->query($sql) == TRUE) {
46.     } else {
47.         echo "Connection error";
48.     }
49.     //gets the questionid of the question just inserted
50.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
51.     $result = $conn->query($sql);
52.     if ($conn->query($sql) == TRUE) {
53.         if ($result->num_rows > 0) {
54.             while($row = $result->fetch_assoc()) {
55.                 $qid = $row["questionsID"];
56.             } }else {
57.                 echo "Connection Error ";
58.             }
59.             $sid = $_SESSION['userData'][2];
60.             //inserts student score for this question
61.             $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`) V
ALUES ('$sid', '$qid', '$score', '$reason')";
62.             if ($conn->query($sql2) ==TRUE) {
63.             } else {
64.                 echo "Connection error";
65.             }
66. ?>
67. <html><center><span><?php echo $display?></span></center></html>
68. <?php
69.     if ($answerStatus=="wrong"){//if wrong, displays answer and solution
70.         echo "<br>";
71.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct answer was : ".$_SESSION['answerOne']." </h4> ";
72.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". "The correct method is : ".$_SESSION['sol']." </h4> ";
73.     }
74.     require "backToSort.php";
75. ?>

```

#### factoriseShortQuestionForm.php

```

1. <!DOCTYPE html>
2. <html> <!--form for student to do factorisation short answer questions -->
3.     <head>
4.     </head>
5.     <body style="background-color:#add8e6;">
6.         <div style="text-align: center;">
7.             <div style="display: inline-block; text-align: left;">
8.                 <form name = "form1" method = "post" action = "factoriseShortQuestionC
heck.php">

```

```

9.         <input type="text" name="answer" />
10.        <p></p>
11.        <p>Reasoning for answer</p>
12.        <input type="text" name="sol" />
13.        <p></p>
14.        <input type = "Submit" Name = "Submit1" VALUE = "Submit">
15.    </form>
16. </div>
17. </div>
18. </body>
19. </html>

```

generalMCQForm.php

```

1. <!DOCTYPE html>
2. <html> <!-- form used for MCQ questions where specific format/type notreq-->
3.     <head>
4.     </head>
5.     <body style="background-color:#add8e6;">
6.         <div style="text-align: center;">
7.             <div style="display: inline-block; text-align: left;">
8.                 <form name = "form1" method = "post" action = "factoriseShortQuestionC
heck.php">
9.                     <input type="text" name="answer" />
10.                    <p></p>
11.                    <p>Reasoning for answer</p>
12.                    <input type="text" name="sol" />
13.                    <p></p>
14.                    <input type = "Submit" Name = "Submit1" VALUE = "Submit">
15.                </form>
16.            </div>
17.        </div>
18.    </body>
19. </html>

```

generalNumberLongForm.php

```

1. <html> <!--form used for long questions where both answers are numbers-->
2.     <body style="background-color:#add8e6;">
3.         <div style="text-align: center;">
4.             <div style="display: inline-block; text-align: left;">
5.                 <form name = "form1" method = "post" action = "longQuestionCheck.php">
6.                     <br>
7.                     <p>Part A</p>
8.                     <input type="number" name="answerOne" required />
9.                     <p></p>
10.                    <p>Part B</p>
11.                    <input type="number" name="answerTwo" required/>
12.                <p></p>
13.                <p>Reasoning for answers</p>
14.                <input type="text" name="sol" />
15.                <p></p>
16.                <input type = "Submit" Name = "Submit1" VALUE = "Submit"/>
17.            </form>
18.        </div>
19.    </div>
20. </body>
21. </html>

```

### Objective 27

Type what answer is

### Objective 28

Question split into two parts

### Objective 33

Option to provide a reason for answers written

### Coding Style

Defensive Programming

*generalShortNumberForm.php*

```

1. <html> <!--form used when answer for short ans question in a number -->
2.   <body style="background-color:#add8e6;">
3.     <div style="text-align: center;">
4.       <div style="display: inline-block; text-align: left;">
5.         <form name ="form1" method ="post" action ="shortQuestionCheck.php"
6.       >
7.         <br>
8.         <input type="number" name="answer" required />
9.         <p></p>
10.        <p>Reasoning for answer</p>
11.        <input type="text" name="sol" />
12.        <p></p>
13.        <input type = "Submit" Name = "Submit1" VALUE = "Submit"/>
14.      </div>
15.    </div>
16.  </body>
17. </html>

```

**Objective 27**

Type what  
answer is

**Objective 33**

Option to provide  
a reason for  
answer written

*generalShortStringForm.php*

```

1. <html> <!--form used when answer for short ans question is a string-->
2.   <body style="background-color:#add8e6;">
3.     <div style="text-align: center;">
4.       <div style="display: inline-block; text-align: left;">
5.         <form name ="form1" method ="post" action ="shortQuestionCheck.php"
6.       >
7.         <br>
8.         <input type="text" name="answer" required/>
9.         <p></p>
10.        <p>Reasoning for answer</p>
11.        <input type="text" name="sol" />
12.        <p></p>
13.        <input type = "Submit" Name = "Submit1" VALUE = "Submit"/>
14.      </div>
15.    </div>
16.  </body>
17. </html>

```

*generalStringLongForm.php*

```

1. <html> <!--form used when both answers for long ans question are strings-->
2.   <body style="background-color:#add8e6;">
3.     <div style="text-align: center;">
4.       <div style="display: inline-block; text-align: left;">
5.         <form name ="form1" method ="post" action ="longQuestionCheck.php">
6.       >
7.         <br>
8.         <p>Part A</p>
9.         <input type="text" name="answerOne" required/>
10.        <p></p>
11.        <p>Part B</p>
12.        <input type="text" name="answerTwo" required/>
13.        <p></p>
14.        <p>Reasoning for answers</p>
15.        <input type="text" name="sol" />
16.        <p></p>
17.        <input type = "Submit" Name = "Submit1" VALUE = "Submit"/>
18.      </div>

```

**Coding Style –  
Defensive  
Programming**

```

19.         </div>
20.     </body>
21. </html>

```

longQuestionCheck.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php //page to check if answers to long ans questions are correct
14.     session_start();
15.     echo "<br>";
16.     $quest = "The question was : ".$SESSION['information'][0];
17.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$quest."</h4> ";
18.     $SESSION['userScoreData'][1]=$SESSION['userScoreData'][1]+2;
19.     echo "<br>";
20.     if (isset($_POST['Submit1'])) {
21.         $selected_radio = $_POST['answerOne'];

22.         if ($selected_radio == $SESSION['information'][1]) { //checks if answer
written is correct. if so, displays green tick, display correct, increase total
correct by 1 and push 1 into array of scores
23.             $answerStatusOne = "correct";
24.             $SESSION['userScoreData'][2]=$SESSION['userScoreData'][2]+1;
25.             array_push($SESSION['userScoreData'][9],1);
26.             $displayOne = "✔";
27.             $score = 1;
28.         }
29.         else {//else, display red cross, display incorrect, increase total wrong by
1 and push 0 into array of scores
30.             $answerStatusOne = "wrong";
31.             $SESSION['userScoreData'][3]=$SESSION['userScoreData'][3]+1;
32.             array_push($SESSION['userScoreData'][9],0);
33.             $displayOne = "✗";
34.             $score = 0;
35.         }
36.         $selected_radio_Two = $_POST['answerTwo'];
37.         if ($selected_radio_Two == $SESSION['information'][3]) {//checks if answer written
is correct. if so, displays green tick, display correct, increase total correct by
1 and push 1 into array of scores
38.             $answerStatusTwo = "correct";
39.             $SESSION['userScoreData'][2]=$SESSION['userScoreData'][2]+1;
40.             array_push($SESSION['userScoreData'][9],1);
41.             $displayTwo = "✔";
42.             $score = $score + 1;
43.         }
44.         else {//else, display red cross, display incorrect, increase total wrong by
1 and push 0 into array of scores
45.             $answerStatusTwo = "wrong";
46.             $SESSION['userScoreData'][3]=$SESSION['userScoreData'][3]+1;
47.             array_push($SESSION['userScoreData'][9],0);
48.             $displayTwo = "✗";
49.         }
50.     }

```

**Algorithm Design –**  
Increment question  
number by 2 and print  
the question

**Algorithm  
Design – If  
part a  
answer is  
correct**

**Algorithm  
Design –  
If part a  
answer is  
wrong**

**Algorithm  
Design –  
If part b  
answer is  
correct**

**Algorithm  
Design – If  
part b  
answer is  
wrong**

```

51.     $answerChoA = "Your response for part a was: ".$selected_radio;
52. echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$answerChoA." </h4> ";
53.     $answerChoARes= "This was : ".$answerStatusOne." ";
54. echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$answerChoARes." </h4> ";
55. ?>
56. <html><center><span><?php echo $displayOne?></span></center></html>
57. <?php
58.     if ($answerStatusOne=="wrong"){ //if wrong, displays correct answer and sol
59.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". "The correct answer was : ".$_SESSION['information'][1]." </h4> ";
60.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". "The correct method is: ".$_SESSION['information'][2]." </h4> ";
61.     }
62.     }
63.     echo "<br>";
64.     echo "<br>";
65.     echo "<br>";
66.     $answerChoB = "Your response for part b was: ".$selected_radio_Two;
67. echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$answerChoB." </h4> ";
68.     $answerChoBRes= "This was : ".$answerStatusTwo." ";
69. echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$answerChoBRes." </h4> ";
70. ?>
71. <html><center><span><?php echo $displayTwo?></span></center></html>
72. <?php
73.     if ($answerStatusTwo=="wrong"){ //if wrong, displays correct answer and sol
74.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". "The correct answer was : ".$_SESSION['information'][3]." </h4> ";
75.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". "The correct method is: ".$_SESSION['information'][4]." </h4> ";
76.     }
77.     require "connection.php";
78.     $conn = connection::getConn();
79.     $num = $_SESSION['userScoreData'][4];
80.     $question = $_SESSION['information'][0];
81.     //inserts question into db
82. $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('$question', $num)";
83.     if ($conn->query($sql) == TRUE) {
84.     } else {
85.         echo "Connection error";
86.     }
87.     //gets questionid of this question just added to db
88.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
89.     $result = $conn->query($sql);
90.     if ($conn->query($sql) == TRUE) {
91.         if ($result->num_rows > 0) {
92.             while($row = $result->fetch_assoc()) {
93.                 $qid = $row["questionsID"];
94.             }
95.         }
96.     } else {
97.         echo "Connection error";
98.     }
99.     $sid = $_SESSION['userData'][2];
100.     //records students score for that question in scorebook
101.     $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`) VALUES ('$sid', '$qid', '$score')";
102.     if ($conn->query($sql2) == TRUE) {
103.     } else {
104.         echo "Connection error";
105.     }
106.     require "backToSort.php";

```

**Algorithm Design –**  
Output for part a if correct or not, if wrong then display correct answer and method

**Algorithm Design –**  
Output for part b if correct or not, if wrong then display correct answer and method

**Algorithm Design –**  
Add question and topicID into question table

**Algorithm Design -**  
Insert studentID, questionID, score and reason into the scorebook table

107. ?&gt;

mcqCheck.php

```

1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.     <style>
6.       body {font-family: "Raleway", Arial, sans-serif}
7.       .w3-row img {margin-bottom: -8px}
8.     </style>
9.   </head>
10.  <body style="background-color:#add8e6;">

11.  </body>
12. </html>

13. <?php //page to check if mcq ans is correct
14.   session_start();
15.   $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+1; //increases
    question number by 1
16.   echo "<br>";
17.   $quest = "The question was : ".$_SESSION['information'][0];
18.   echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$_quest." </h4> ";
19.   $reason = $_POST['sol'];
20.   $answerStatus = "wrong";
21.   $options = "correct";
22.   if (isset($_POST['Submit1'])) {
23.     $selected_radio = $_POST['Opt'];
24.     if ($selected_radio == 'ans') { //checks if answer chosen was correct
25.       $answerStatus = "correct"; //if so, displays correct, green tick box,
        adds 1 to total correct and pushed 1 into array of scores
26.       $_SESSION['userScoreData'][2]=$_SESSION['userScoreData'][2]+1;
27.       array_push($_SESSION['userScoreData'][9],1);
28.       $display = "✔";
29.       $score = 1;
30.     }
31.     else if ($selected_radio == 'opte') { //displays wrong, red cross box, adds
        1 to total wrong and pushes 0 into array of scores
32.       $options = 'wrong';
33.       $answerChosen = $_SESSION['information'][3];
34.       $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;
35.       array_push($_SESSION['userScoreData'][9],0);
36.       $display = "✗";
37.       $score = 0;
38.     }
39.     else if ($selected_radio == 'opten') { //displays wrong, red cross box, adds
        1 to total wrong and pushes 0 into array of scores
40.       $options = 'wrong';
41.       $answerChosen = $_SESSION['information'][4];
42.       $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;
43.       array_push($_SESSION['userScoreData'][9],0);
44.       $display = "✗";
45.       $score = 0;
46.     }
47.   }
48.   $answerCho = "The answer you chose was : ".$answerStatus;
49.   echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$_answerCho." </h4> ";
50.   ?>
51. <html><center><span><?php echo $display?></span></center></html>
52. <?php

```

**Algorithm****Design –**

Increment question number & display the question

**Algorithm Design -**

If answer selected is correct

**Algorithm Design**

– if answer selected is wrong

**Objective 24**

Once student completes a question, there should be a comment

**Objective 29**

If student answer wrong, there should be mini explanation for correct answer and method

```

53.     if ($answerStatus=="wrong"){ //if wrong displays correct answer and correct
method
54.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The answer you chose was: ".$answerChosen." </h4> ";
55.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The correct answer was : ".$_SESSION['information'][1]." </h4> ";
56.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The correct method is: ".$_SESSION['information'][2]." </h4> ";
57.     }
58.     require "connection.php";
59.     $conn = connection::getConn();
60.     $num = $_SESSION['userScoreData'][4];
61.     //adds question to db
62.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('".$_SESSION['in
formation'][0]."', $num)";
63.     if ($conn->query($sql) == TRUE) {
64.     } else {
65.         echo "Connection error";
66.     }
67.     $question = $_SESSION['information'][0];

68.     //get the question ID of the question ID just added to the db
69.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
70.     $result = $conn->query($sql);
71.     if ($conn->query($sql) == TRUE) {
72.         if ($result->num_rows > 0) {
73.             while($row = $result->fetch_assoc()) {
74.                 $qid = $row["questionsID"];
75.             }
76.         }
77.     } else {
78.         echo "Connection error";
79.     }
80.     $sid = $_SESSION['userData'][2];
81.     //insert students attempt at the question into scorebook table
82.     $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`)
VALUES ('$sid', '$qid', '$score', '$reason')";
83.     if ($conn->query($sql2) ==TRUE) {
84.     } else {
85.         echo "Connection error";
86.     }
87.     require "backToSort.php";
88. ?>

```

SQL – Insert question  
student got into database

**Algorithm  
Design –**  
Insert  
question  
and  
topicID  
into  
question  
table

**Algorithm  
Design -**  
Insert  
studentID,  
questionID,  
score and  
reason into  
the  
scorebook  
table

mcqDisplay.php

**Coding Style – Well designed User Interface**



**Objective 26**

Student select  
an answer

**Objective 33**

Student provide  
reason for  
answer chosen

```

1. <html> <!--Page to display MCQ question and options to user-->
2.   <body style="background-color:#add8e6;">
3.     <div style="text-align: center;">
4.       <div style="display: inline-block; text-align: left;">
5.         <form name = "form1" method = "post" action = "mcqCheck.php">
6.           <input type = 'Radio' Name = 'Opt' value= '<?php echo $fstatus;
7.             ?>' required><?php if ($first==$one){echo $answer;} elseif($first==$two){echo $opt0
8.               ne;} elseif($first==$three) {echo $optTwo;} ?><br>
9.             <input type = 'Radio' Name = 'Opt' value= '<?php echo $sstatus;?
10.              >' required ><?php if ($second==$one){echo $answer;} elseif($second==$two){echo $op
11.                tOne;} elseif($second==$three) {echo $optTwo;} ?><br>
12.              <input type = 'Radio' Name = 'Opt' value= '<?php echo $tstatus;?
13.                >' required ><?php if ($third==$one){echo $answer;} elseif($third==$two){echo $opt0
14.                  ne;} elseif($third==$three) {echo $optTwo;} ; ?><br>
15.                <p>Reasoning for answer chosen</p>
16.                <input type="text" name="sol" />
17.                <p></p>
18.                <input type = "Submit" Name = "Submit1" VALUE = "Submit">
19.              </form>
20.            </div>
21.          </div>
22.        </body>
23.      </html>

```

```

15.         </div>
16.     </body>
17. </html>

```

*mcqOptionOrder.php*

```

1. <?php //page to generate answer and options in random order
2.     $one = "<Input type = 'Radio' Name = 'Opt' value= 'ans' ><?php echo $answer;?>";
3.     $two = "<Input type = 'Radio' Name = 'Opt' value= 'opte' ><?php echo $optOne;?>"
4.     ;
5.     $three = "<Input type = 'Radio' Name = 'Opt' value= 'opten' ><?php echo $optTwo;
6.     ?>";
7.     $outputRandomisation = rand(1,3);
8.     if ($outputRandomisation == 1){
9.         $first = $one;
10.        $fstatus = 'ans';
11.        $second = $two;
12.        $sstatus = 'opte';
13.        $third = $three;
14.        $tstatus = 'opten';
15.    }
16.    else if ($outputRandomisation == 2){
17.        $first = $two;
18.        $fstatus = 'opte';
19.        $second = $one;
20.        $sstatus = 'ans';
21.        $third = $three;
22.        $tstatus = 'opten';
23.    }
24.    else if ($outputRandomisation == 3){
25.        $first = $two;
26.        $fstatus = 'opte';
27.        $second = $three;
28.        $sstatus = 'opten';
29.        $third = $one;
30.        $tstatus = 'ans';
31.    }
32.    ?>

```

*mixedOperation.php*

```

1. <?php //mixed operation topics question generator
2.     //functions to calculate answer
3.     session_start();
4.     require "connection.php";
5.     require "assignment.php";
6.     require "number.php";
7.     require "string.php";
8.     require "titlePage.php";
9.     $conn = connection::getConn();
10.    $s = new assignment;
11.    $num = new number;
12.    $string = new stringQuestion;
13.    // $type is given through a get value sent from the sort page which determines
14.    the type of question
15.    $type = $_GET['type'];
16.    // $question is calling a function in the class assignment, which has topic ID
17.    6, and user difficulty of $_SESSION['userScoreData'][8]
18.    $question=$s->question(6,$type,$_SESSION['userScoreData'][8]);
19.    if ($type == "m"){
20.        //generates numbers based on difficulty and understanding
21.        if ($_SESSION['userScoreData'][8]==1){
22.            $number1 = $num->getRandomNumber(2,10);

```

```

21.         $number2=$num->getRandomNumber(2,10);
22.     }elseif ($_SESSION['userScoreData'][8]==2){
23.         $number1 = $num->getRandomNumber(2,100);
24.         $number2=$num->getRandomNumber(2,100);
25.     }elseif ($_SESSION['userScoreData'][8]==3){
26.         $number1 = $num->getRandomNumber(2,200);
27.         $number2=$num->getRandomNumber(2,200);
28.     }
29.     //randomly picks operation and calcs answer, sol and updates the question
30.     $opChoice = rand(1, 4);
31.     if ($opChoice == 1){
32.         $op = "+";
33.         $answer = $num->addition($number1,$number2);
34.         $updated1 = $string->getString("o1", "$op", $question);
35.         $sol=$number1." + ".$number2;
36.     }elseif ($opChoice == 2){
37.         $op = "-"; //makes sure no negative number is produced
38.         if ($number1 < $number2){
39.             $temp = $number2;
40.             $number2 = $number1;
41.             $number1 = $temp;
42.         }
43.         $answer = $num->subtraction($number1,$number2);
44.         $sol=$number1." - ".$number2;
45.         $updated1 = $string->getString("o1", "$op", $question);
46.     }elseif ($opChoice == 3){
47.         $op = "*";
48.         $answer = $num->multiply($number1,$number2);
49.         $updated1 = $string->getString("o1", "$op", $question);
50.         $sol=$number1." X ".$number2;
51.     }elseif ($opChoice == 4){
52.         $op = "/";
53.         if (($num->intDiv($number1 , $number2))==false){ //makes sure integer
sol is produced
54.             while (($num->intDiv($number1 , $number2))==false) {
55.                 if ($_SESSION['userScoreData'][8]==1){
56.                     $number1 = $num->getRandomNumber(2,10);
57.                     $number2=$num->getRandomNumber(2,10);
58.                 }elseif ($_SESSION['userScoreData'][8]==2){
59.                     $number1 = $num->getRandomNumber(2,100);
60.                     $number2=$num->getRandomNumber(2,100);
61.                 }elseif ($_SESSION['userScoreData'][8]==3){
62.                     $number1 = $num->getRandomNumber(2,200);
63.                     $number2=$num->getRandomNumber(2,200);
64.                 }
65.             }
66.         }
67.         $answer = $num->division($number1,$number2);
68.         $updated1 = $string->getString("o1", "$op", $question);
69.         $sol=$number1." / ".$number2;
70.     }
71.     $updated2 = $string->getString("n1",$number1,$updated1);
72.     $updated3 = $string->getString("n2",$number2,$updated2);
73.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3." </h4> ";
74.     $optOne = ($answer+(rand(1,10)));
75.     $optTwo = ($answer-(rand(1,10)));
76.     $_SESSION['information']=array($updated3,$answer,$sol,$optOne,$optTwo);
77.     require "mcqOptionOrder.php";
78.     require "mcqDisplay.php";
79. }elseif ($type == "s"){
80.     //generates numbers based on user understanding
81.     if ($_SESSION['userScoreData'][8]==1){
82.         $number1 = $num->getRandomNumber(20,500);
83.         $number2=$num->getRandomNumber(20,500);
84.         $number3 = $num->getRandomNumber(20,500);

```

```

85.         }elseif ($_SESSION['userScoreData'][8]==2){
86.             $number1 = $num->getRandomNumber(20,750);
87.             $number2=$num->getRandomNumber(20,750);
88.             $number3 = $num->getRandomNumber(20,750);
89.         }elseif ($_SESSION['userScoreData'][8]==3){
90.             $number1 = $num->getRandomNumber(20,1000);
91.             $number2=$num->getRandomNumber(20,1000);
92.             $number3 = $num->getRandomNumber(20,1000);
93.         }
94.         if (($num->intDivMult($number2 , $number1, $number3))==false){ //makes sure
answer in integer else regenerates until integer
95.             while(($num->intDivMult($number2 , $number1, $number3))==false) {
96.                 if ($_SESSION['userScoreData'][8]==1){
97.                     $number1 = $num->getRandomNumber(20,500);
98.                     $number2=$num->getRandomNumber(20,500);
99.                     $number3 = $num->getRandomNumber(20,500);
100.                 }elseif ($_SESSION['userScoreData'][8]==2){
101.                     $number1 = $num->getRandomNumber(20,750);
102.                     $number2=$num->getRandomNumber(20,750);
103.                     $number3 = $num->getRandomNumber(20,750);
104.                 }elseif ($_SESSION['userScoreData'][8]==3){
105.                     $number1 = $num->getRandomNumber(20,1000);
106.                     $number2=$num->getRandomNumber(20,1000);
107.                     $number3 = $num->getRandomNumber(20,1000);
108.                 }
109.             }
110.         }
111.         $updated1 = $string->getString("n1",$number1,$question);
112.         $updated2 = $string ->getString("n2",$number2,$updated1);
113.         $updated3 = $string ->getString("n3",$number3,$updated2);
114.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated3." </h4> ";
115.         $answer = $num->multiply(($num->division($number2,$number1)),($number
3));
116.         $sol= "(".$number2." / ".$number1.") x ".$number3";
117.         $_SESSION['information']=array($updated3,$answer,$sol);
118.         require "generalShortNumberForm.php";
119.     }elseif($type == "1"){
120.         //generates numbers based on user understanding
121.         if ($_SESSION['userScoreData'][8]==1){
122.             $number1 = $num->getRandomNumber(10,75);
123.             $number2=$num->getRandomNumber(10,75);
124.             $number3=$num->getRandomNumber(10,75);
125.             $number4 = $num->getRandomNumber(10,75);
126.             $number5=$num->getRandomNumber(10,75);
127.             $number6=$num->getRandomNumber(10,75);
128.         }elseif ($_SESSION['userScoreData'][8]==2){
129.             $number1 = $num->getRandomNumber(10,100);
130.             $number2=$num->getRandomNumber(10,100);
131.             $number3=$num->getRandomNumber(10,100);
132.             $number4 = $num->getRandomNumber(10,100);
133.             $number5=$num->getRandomNumber(10,100);
134.             $number6=$num->getRandomNumber(10,100);
135.         }elseif ($_SESSION['userScoreData'][8]==3){
136.             $number1 = $num->getRandomNumber(10,200);
137.             $number2=$num->getRandomNumber(10,200);
138.             $number3=$num->getRandomNumber(10,200);
139.             $number4 = $num->getRandomNumber(10,200);
140.             $number5=$num->getRandomNumber(10,200);
141.             $number6=$num->getRandomNumber(10,200);
142.         }
143.         if (((($num->intDiv($number5 , $number6))==false) or ($num->multiply(
$number1 , $number2) < $num->addition( ($num->multiply($number3 , $number4) ) , ($n
um->division($number5,$number6))))){ //if answer not integer or positive then
regenerates until condition is satisfied
144.             while ((is_int($number5 / $number6))==false) {

```

```

145.         if ($_SESSION['userScoreData'][8]==1){
146.             $number1 = $num->getRandomNumber(10,75);
147.             $number2=$num->getRandomNumber(10,75);
148.             $number3=$num->getRandomNumber(10,75);
149.             $number4 = $num->getRandomNumber(10,75);
150.             $number5=$num->getRandomNumber(10,75);
151.             $number6=$num->getRandomNumber(10,75);
152.         }elseif ($_SESSION['userScoreData'][8]==2){
153.             $number1 = $num->getRandomNumber(10,100);
154.             $number2=$num->getRandomNumber(10,100);
155.             $number3=$num->getRandomNumber(10,100);
156.             $number4 = $num->getRandomNumber(10,100);
157.             $number5=$num->getRandomNumber(10,100);
158.             $number6=$num->getRandomNumber(10,100);
159.         }elseif ($_SESSION['userScoreData'][8]==3){
160.             $number1 = $num->getRandomNumber(10,200);
161.             $number2=$num->getRandomNumber(10,200);
162.             $number3=$num->getRandomNumber(10,200);
163.             $number4 = $num->getRandomNumber(10,200);
164.             $number5=$num->getRandomNumber(10,200);
165.             $number6=$num->getRandomNumber(10,200);
166.         }
167.     }
168. }
169. $updated1 = $string->getString("n1",$number1,$question);
170. $updated2 = $string ->getString("n2",$number2,$updated1);
171. $updated3 = $string ->getString("n3",$number3,$updated2);
172. $updated4 = $string->getString("n4",$number4,$updated3);
173. $updated5 = $string ->getString("n5",$number5,$updated4);
174. $updated6 = $string ->getString("n6",$number6,$updated5);
175. echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>". $updated6." </h4> ";
176. $answerOne = ($num->multiply($number3 , $number4));
177. $solA = $number3." x ".$number4;
178. $answerTwo = ($num->addition(($num->subtraction(($num->multiply($nu
    mber1 , $number2)), ($num->multiply($number3 , $number4)))), $num->division($number5
    , $number6)));
179. $solB = (".$number1." x ".$number2.") - ( ".$number3." x ".$number4
    .") + ( ".$number5." / ".$number6.)";
180. $_SESSION['information']=array($updated6,$answerOne,$solA,$answerTwo
    , $solB);
181.     require "generalNumberLongForm.php";
182. }
183. ?>

```

*multiply.php*

```

1. <?php //multiplication question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    // $type is given through a get value sent from the sort page which determines
    the type of question
13.    $type = $_GET['type'];
14.    // $question is calling a function in the class assignment, which has topic ID
    4, and user difficulty of $_SESSION['userScoreData'][8]
15.    $question=$s->question(4,$type,$_SESSION['userScoreData'][8]);
16.    $id= $s->id($question);

```

**Technique- Encapsulation**

```

17.     if ($type == "m"){
18.         //generates numbers based on user understanding
19.         if ($_SESSION['userScoreData'][8]==1){
20.             $number1 = $num->getRandomNumber(5,10);
21.             $number2=$num->getRandomNumber(5,10);
22.         }elseif ($_SESSION['userScoreData'][8]==2){
23.             $number1 = $num->getRandomNumber(5,20);
24.             $number2=$num->getRandomNumber(5,20);
25.         }elseif ($_SESSION['userScoreData'][8]==3){
26.             $number1 = $num->getRandomNumber(5,50);
27.             $number2=$num->getRandomNumber(5,50);
28.         }
29.         $updated1 = $string->getString("n1",$number1,$question);
30.         $updated2 = $string->getString("n2",$number2,$updated1);
31.         $answer = $num->multiply($number1,$number2);
32.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". $updated2." </h4> ";
33.         $sol=$number1." X ".$number2;
34.         $optOne = $num->optionOne($answer);
35.         $optTwo = $num->optionTwo($answer);
36.         $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
37.         require "mcqOptionOrder.php";
38.         require "mcqDisplay.php";
39.     }elseif ($type == "s"){
40.         $number1 = $num->getRandomNumber(20,100);
41.         $number2=$num->getRandomNumber(20,100);
42.         $updated1 = $string->getString("n1",$number1,$question);
43.         $updated2 = $string->getString("n2",$number2,$updated1);
44.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". $updated2." </h4> ";
45.         $answer = $num->multiply($number1,$number2);
46.         $sol = "$number1" . " x " . "$number2";
47.         $_SESSION['information']=array($updated2,$answer,$sol);
48.         require "generalShortNumberForm.php";
49.     }elseif($type == "l"){
50.         $number1 = $num->getRandomNumber(2,100);
51.         $updated1 = $string->getString("n1",$number1,$question);
52.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". $updated1." </h4> ";
53.         $answerOne = $num->multiply($number1,7);
54.         $solA = $number1." x 7";
55.         $answerTwo = $num->multiply($answerOne,52);
56.         $solB = $number1." x 7 x 52";
57.         $_SESSION['information']=array($updated1,$answerOne,$solA,$answerTwo,$solB);
58.         ;
59.         require "generalNumberLongForm.php";
60.     }
61. }
62. ?>

```

number.php

Technique - Class

**Objective 23**

Numbers used are random

**Objective 30**

Numbers should be whole numbers

**Objective 31**

Correct answer is whole number

**Objective 32**

Each option in MCQ is whole number

```

1. <?php
2. //class to generate random number
3. class number {
4.     private $number;
5.     function getRandomNumber($lowerVal,$upperVal) {
6.         $this->number = (rand($lowerVal,$upperVal));
7.         return $this->number;
8.     }
9.     function getNumber(){
10.         return $this->number;
11.     }
12.     function multiply($number1,$number2){
13.         $this->ans = ($number1 * $number2);
14.         return $this->ans;

```

Generate Random Number

Coding Style – Modules with appropriate interfaces

Coding Style – Modularisation of code

```

15.     }
16.     function multiplyTwo($number1){//multiply a number by 2
17.         $this->ans = ($number1 * 2);
18.         return $this->ans;
19.     }
20.     function multiplyFour($number1){//multiply a number by 4
21.         $this->ans = ($number1 * 4);
22.         return $this->ans;
23.     }

24.     function division($number1,$number2){
25.         //function that calcs division
26.         $this->ans = ($number1 / $number2);
27.         return $this->ans;
28.     }
29.     function addition($number1,$number2){
30.         //function that calcs division
31.         $this->ans = ($number1 + $number2);
32.         return $this->ans;
33.     }
34.     function subtraction($number1,$number2){
35.         $this->ans = ($number1 - $number2);
36.         return $this->ans;
37.     }

38.     function intDiv($num1,$num2){ //makes sure is integer
39.         $this->inte = is_int($num1 / $num2);
40.         return $this->inte;
41.     }
42.     function intDivMult($num1,$num2,$num3){ //makes sure is integer
43.         $this->inte = is_int(($num1 / $num2)*$num3);
44.         return $this->inte;
45.     }
46.     function genUnit(){ //function to generate a random unit
47.         $unit=array("mm","cm","m");
48.         $randIndex = array_rand($unit);
49.         $this->unit = $unit[$randIndex];
50.         return $this->unit;
51.     }

52.     function optionOne($answer){ //option based on final answer
53.         $this->optOne = $answer+(rand(1,5));
54.         if ($this->optOne <0){
55.             while ($this->optOne <0){
56.                 $this->optOne = ($answer+(rand(1,5)));
57.             }
58.         }
59.         return $this->optOne;
60.     }
61.     function optionOneTwoNumbers($num1,$num2){
62.         return $this->optOne = $num1+$num2;
63.     }
64.     function optionTwo($answer){ //option based on final answer
65.         $this->optTwo = ($answer-(rand(1,5)));
66.         if ($this->optTwo <0){
67.             $cons = (rand(6,10));
68.             $this->optTwo = ($answer+$cons);
69.         }
70.         return $this->optTwo;
71.     }
72. }
73. ?>

```

**Coding Style – Loosely couple modules**

**Coding Style – Cohesive modules**

**Coding Style - Modules**

**Coding Style – Use of constants**

```
1. <html>
2.   <head>
3.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
```



```

4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet" type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php //checks to see if ans chosen for number topic is correct
14.     session_start();
15.     $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+1;
16.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>". "The question was: " .$_SESSION['quest'] . " </h4> ";
17.     echo "<br>";
18.     $reason = $_POST['sol'];
19.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>". "The sequence was: " .$_SESSION['list'][0] . " " .$_SESSION['list'][1] . " " .$_SESSION['list'][2] . " " .$_SESSION['list'][3] . " " . " </h4> ";
20.     echo "<br>";
21.     $ans = $_SESSION['answer'];
22.     if (isset($_POST['Submit1'])) {
23.         $selected_radio = $_POST['Opt'];
24.         if ($selected_radio == $ans) { //checks if answer chosen is the right answer. if it is, displays correct, green tick box, adds 1 to total correct, adds 1 to array of scores
25.             $answerStatus = "correct";
26.             $display = "✔";
27.             $score = 1;
28.             array_push($_SESSION['userScoreData'][9],1);
29.         }
30.         else { //if wrong displays incorrect, red cross box, adds 1 to total wrong, adds 0 to array of scores
31.             $answerStatus = "wrong";
32.             $display = "✗";
33.             $score = 0;
34.             array_push($_SESSION['userScoreData'][9],0);
35.         }
36.     }
37.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>". "The answer you chose was " . $answerStatus . " </h4> ";
38.     require "connection.php";
39.     $conn = connection::getConn();
40.     $num = $_SESSION['userScoreData'][4];
41.     $question = $_SESSION["quest"];
42.     //adds question to db
43.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('$question', $num)";
44.     if ($conn->query($sql) == TRUE) {
45.     } else {
46.         echo "Connection error";
47.     }
48.     //gets question ID of question just added
49.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
50.     $result = $conn->query($sql);
51.     if ($conn->query($sql) == TRUE) {
52.         if ($result->num_rows > 0) {
53.             while($row = $result->fetch_assoc()) {
54.                 $qid = $row["questionsID"];
55.             } } else {
56.                 echo "Error: " . $sql . "<br> " . $conn->error;
57.             }
58.             $sid = $_SESSION['userData'][2];
59.             //inserts students attempt of question in db

```

```

60. $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`) V
    ALUES ('$sid', '$qid', '$score', '$reason')";
61. if ($conn->query($sql2) == TRUE) {
62. } else {
63.     echo "Connection error";
64. }
65. ?>
66. <html><center><span><?php echo $display?></span></center></html>
67. <?php
68.     if ($answerStatus=="wrong"){
69.         //if wrong, displays correct answer and sol
70.         echo "<h4 align=center style='margin-left:30%;margin-
            right:30%;>". "The correct answer was : ".$_SESSION['answer']. " </h4> ";
71.         echo "<h4 align=center style='margin-left:30%;margin-
            right:30%;>". "The correct method is: ".$_SESSION['sol']. " </h4> ";
72.     }
73.     require "backToSort.php";
74. ?>

```

**SQL** – Insert score for a question into a database (part 2)

numberMCQForm.php

```

1. <html> <!--number topic mcq form for student -->
2. <body style="background-color:#add8e6;">
3. <div style="text-align: center;">
4. <div style="display: inline-block; text-align: left;">
5. <form name="form1" method="post" action="numberMCQCheck.php">
6. <input type="Radio" Name='Opt' value= '<?php echo $listOfNum
ber[0];?>' ><?php echo $listOfNumber[0]; ?><br>
7. <input type="Radio" Name='Opt' value= '<?php echo $listOfNum
ber[1];?>' ><?php echo $listOfNumber[1]; ?><br>
8. <input type="Radio" Name='Opt' value= '<?php echo $listOfNum
ber[2];?>' ><?php echo $listOfNumber[2];?><br>
9. <input type="Radio" Name='Opt' value= '<?php echo $listOfNum
ber[3];?>' ><?php echo $listOfNumber[3];?><br>
10. <p>
11. <p>Reasoning for answer chosen</p>
12. <input type="text" name="sol" />
13. <p></p>
14. <input type="Submit" Name="Submit1" VALUE="Submit">
15. </form>
16. </div>
17. </div>
18. </body>
19. </html>

```

**Coding Style**  
Appropriate Indentation

numberQ.php

```

1. <?php //number topic question generator
2. function completeNumbers($numbers){ //makes sure unique numbers in array
3.     $noRepeats = array_unique($numbers);
4.     while ((count($noRepeats)) != 4){
5.         $arr = $noRepeats;
6.         array_push($arr,rand(10,100));
7.         $noRepeats = array_unique($arr);
8.     }
9.     return $noRepeats;
10. }
11. function merge_sort($my_array){
12.     if(count($my_array) == 1 ){ return $my_array;}
13.     $mid = count($my_array) / 2;
14.     $left = array_slice($my_array, 0, $mid);
15.     $right = array_slice($my_array, $mid);
16.     $left = merge_sort($left);
17.     $right = merge_sort($right);

```

**Check unique numbers in the array**

**Technique**  
Recursive Algorithm

**Technique** – MergeSort

```

18.         return merge($left, $right);
19.     }
20.     function merge($left, $right){
21.         $res = array();
22.         while (count($left) > 0 && count($right) > 0){
23.             if($left[0] > $right[0]){
24.                 $res[] = $right[0];
25.                 $right = array_slice($right, 1);
26.             }else{
27.                 $res[] = $left[0];
28.                 $left = array_slice($left, 1);
29.             }
30.         }
31.         while (count($left) > 0){
32.             $res[] = $left[0];
33.             $left = array_slice($left, 1);
34.         }
35.         while (count($right) > 0){
36.             $res[] = $right[0];
37.             $right = array_slice($right, 1);
38.         }
39.         return $res;
40.     }
41.     function sortOut($listOfNumbers){//sorts into increasing order
42.         shuffle($listOfNumbers);
43.         return $listOfNumbers;
44.     }
45.     session_start();
46.     require "connection.php";
47.     require "assignment.php";
48.     require "number.php";
49.     require "string.php";
50.     require "titlePage.php";
51.     $conn = connection::getConn();
52.     $s = new assignment;
53.     $num = new number;
54.     $string = new stringQuestion;
55.     //$type is given through a get value sent from the sort page which determines the
type of question
56.     $type = $_GET['type'];
57.     //$question is calling a function in the class assignment, which has topic ID 3,
and user difficulty of $_SESSION['userScoreData'][8]
58.     $question=$s->question(3,$type,$_SESSION['userScoreData'][8]);
59.     $id= $s->id($question);
60.     if ($type == "m"){
61.         //create array, sort
62.         $numbers = array(rand(10,100),rand(10,100),rand(10,100),rand(10,100));
63.         $listOfNumbers = completeNumbers($numbers);
64.         $sort = merge_sort($listOfNumbers); // url ref - https://www.w3resource.com/ php-
exercises/searching-and-sorting-algorithm/searching-and-sorting-algorithm- exercise-
17.php
65.         $listOfNumber = sortOut($sort);
66.         if ($id==12){
67.             $lookForLargest = 1;
68.         }
69.         if ($id==13){
70.             $lookForLargest = 0;
71.         }
72.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$question." </h4> ";
73.         $_SESSION['quest'] = $question;
74.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$listOfNumber[0]." ".$listOfNumber[1]." ".$listOfNumber[2]." ".$listOf
Number[3]." </h4>";
75.         $_SESSION['list']=$listOfNumber;
76.         if ($lookForLargest ==1){

```

```

77.         $list = $listOfNumber;
78.         sort($list);
79.         $_SESSION['answer']=$list[3];
80.     }
81.     if ($lookForLargest ==0){
82.         $list = $listOfNumber;
83.         sort($list);
84.         $_SESSION['answer']=$list[0];
85.     }
86.     $_SESSION['sol']="The list in sorted order is: ".$list[0]. " ".$list[1]. " ".$
list[2]. " ".$list[3];
87.     require "numberMCQForm.php";
88. }elseif ($type == "s"){
89.     $f = new NumberFormatter("en", NumberFormatter::SPELLOUT); // url ref - http
s://stackoverflow.com/questions/1150088/php-express-number-in-words
90.     //chooses number based on user understanding and question type
91.     if ($id==1 or $id==117 or $id==118){
92.         if ($_SESSION['userScoreData'][8]==1){
93.             $number1 = $num->getRandomNumber(10,50);
94.         }elseif ($_SESSION['userScoreData'][8]==2){
95.             $number1 = $num->getRandomNumber(10,100);
96.         }elseif ($_SESSION['userScoreData'][8]==3){
97.             $number1 = $num->getRandomNumber(10,300);
98.         }
99.         $question = $string->getString("n1",$number1,$question);
100.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$question."</h4> ";
101.         $answer = $f->format($number1);
102.     }
103.     if ($id==9){
104.         $number1 = $num->getRandomNumber(1,10);
105.         $number2=$num->getRandomNumber(1,99);
106.         $sn1 = $f->format($number1);
107.         $sn2 = $f->format($number2);
108.         $updated1 = $string->getString("sn1",$sn1,$question);
109.         $question = $string->getString("sn2",$sn2,$updated1);
110.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$question."</h4> ";
111.         if ($number2>9){
112.             $answer = (int)($number1.$number2);
113.         }else if ($number2<10){
114.             $answer = (int)($number1."0".$number2);
115.         }
116.     }
117.     $sol = "Check carefully - look at each word and convert it to its number.";

118.     $_SESSION['information']=array($question,$answer,$sol);
119.     $typeOne = "text";
120.     $typeTwo = "number";
121.     require "numberShortForm.php";
122. }elseif($type == "l"){
123.     $f = new \NumberFormatter("en", NumberFormatter::SPELLOUT); // url ref - ht
tps://stackoverflow.com/questions/1150088/php-express-number-in-words
124.     //chooses number based on user understanding and question type
125.     if ($_SESSION['userScoreData'][8]==1){
126.         $number1 = $num->getRandomNumber(10,50);
127.         $number2 = $num->getRandomNumber(10,50);
128.     }elseif ($_SESSION['userScoreData'][8]==2){
129.         $number1 = $num->getRandomNumber(10,100);
130.         $number2 = $num->getRandomNumber(10,100);
131.     }elseif ($_SESSION['userScoreData'][8]==3){
132.         $number1 = $num->getRandomNumber(10,300);
133.         $number2 = $num->getRandomNumber(10,300);
134.     }
135.     $updated1 = $string->getString("n1",$number1,$question);
136.     $updated2 = $string->getString("n2",$number2,$updated1);

```

Number Formatter Class

```

137.         echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;'>".$updated2." </h4> ";
138.         $answerOne = $f->format($number1);
139.         $solA="Check the spelling!";
140.         $answerTwo = $f->format($number2);
141.         $solB="Check the spelling!";
142.         $_SESSION['information']=array($updated2,$answerOne,$solA,$answerTwo,$solB)
    ;
143.         require "generalStringLongForm.php";
144.     }
145. ?>

```

numberShortForm.php

```

1. <!DOCTYPE html>
2. <html> <!--form if question is short answer for number topic-->
3.     <head>
4.     </head>
5.     <body>
6.         <div style="text-align: center;">
7.             <div style="display: inline-block; text-align: left;">
8.                 <form name = "form1" method = "post" action = "shortQuestionCheck.php"
    >
9.                     <input type="<?php if($id==1 or $id==117 or $id==118){echo $typ
    eOne;}else{echo $typeTwo;}?>" name="answer" />
10.                    <p>
11.                        <p>Reasoning for answer</p>
12.                        <input type="text" name="sol" />
13.                    <p></p>
14.                    <input type = "Submit" Name = "Submit1" VALUE = "Submit">
15.                </form>
16.            </div>
17.        </div>
18.    </body>
19. </html>

```

percentage.php

Coding Style – Appropriate Indentation

```

1. <?php //percentage topic question generator
2.     //makes sure percentage is factor of 2 or 5
3.     function genRandomNumberOne(){
4.         $num = (rand(2, 99));
5.         if (($num%2!=0)or ($num%5!=0)){
6.             while (($num%2!=0)or ($num%5!=0)){
7.                 $num = (rand(2, 99));
8.             }
9.         }
10.        return $num;
11.    }
12.    //makes sure number generated is a factor of 5
13.    function genRandomNumberTwo(){
14.        $num = (rand(2, 99));
15.        if (($num%5!=0)){
16.            while (($num%5!=0)){
17.                $num = (rand(2, 99));
18.            }
19.        }
20.        return $num;
21.    }
22.    session_start();
23.    require "connection.php";
24.    require "assignment.php";
25.    require "number.php";
26.    require "string.php";

```

Algorithm Design –  
genRandomNumberOne function

Algorithm Design –  
genRandomNumberTwo function

Algorithm  
Design  
instantiate,  
call, get

```

27.     require "titlePage.php";
28.     $conn = connection::getConn();
29.     $s = new assignment;
30.     $num = new number;
31.     $string = new stringQuestion;
32.     //$type is given through a get value sent from the sort page which determines
    the type of question
33.     $type = $_GET['type'];
34.     //$question is calling a function in the class assignment, which has topic ID
    10, and user difficulty of $_SESSION['userScoreData'][8]
35.     $question=$s->question(10,$type,$_SESSION['userScoreData'][8]);
36.     $id= $s->id($question);
37.     if ($type == "m"){
38.         $number1 = genRandomNumberOne();
39.         $number2 = genRandomNumberTwo();
40.         $updated1 = $string->getString("n1",$number1,$question);
41.         $updated2 = $string ->getString("p1",$number2,$updated1);
42.     echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;>". $updated2." </h4> ";
43.         $answer = $num->multiply($number1 , ($num->division($number2,100));
44.         $sol = $number1 ." X (". $number2." / 100)";
45.         $optOne = $num->optionOne($answer);
46.         $optTwo = $num->optionTwo($answer);
47.         $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
48.         require "mcqOptionOrder.php";
49.         require "mcqDisplay.php";
50.     }elseif ($type == "s"){
51.         //calculates smart numbers that student can use to calculate answers, by putting
        all suitable numbers into an associative array
52.         $percentStart = 5; //url ref - https://stackoverflow.com/questions/57781077
        /how-to-calculate-optimum-numbers-to-use-in-a-percentage-problem-in-php
53.         $percentEnd = 75;
54.         $numberStart = 5;
55.         $numberEnd = 200;
56.         $eps = 1.E-9;
57.         $results=[];
58.         for($percent = $percentStart; $percent <= $percentEnd; $percent++){
59.             for($number = $numberStart; $number <= $numberEnd; $number++){
60.                 $percentFromNumber = $percent * $number/100;
61.                 if(fmod($percentFromNumber,1) < $eps){
62.                     $results[] = [
63.                         'percent' => $percent,
64.                         'number' => $number,
65.                         '%number' => $percentFromNumber 66.
66.                     ];
67.                 }
68.             }
69.         }
70.         //gets random element from associative array and gets all its elements
71.         $num = rand(1,660);

72.         $number1 = ($results[$num]['percent']);
73.         $number2 = ($results[$num]['number']);
74.         $answer = ($results[$num]['%number']);
75.         $updated1 = $string->getString("n1",$number1,$question);
76.         $updated2 = $string ->getString("n2",$number2,$updated1);
77.     echo "<h4 align=center style='margin-left:30%;margin-
        right:30%;>". $updated2." </h4> ";
78.         $sol="(".$number2." / 100) X ".$number1;
79.         $_SESSION['information']=array($updated2,$answer,$sol);
80.         require "generalShortNumberForm.php";
81.     }elseif($type == "l"){

82.         $percentStart = 5; //url ref - https://stackoverflow.com/questions/57781077
        /how-to-calculate-optimum-numbers-to-use-in-a-percentage-problem-in-php
83.         $percentEnd = 75;
84.         $numberStart = 5;
85.         $numberEnd = 200;

```

**Algorithm Design** – If question type is multiple choice

**Data Structure**  
Associative arrays

**Algorithm Design** – If question type is short answer question

**Data Structure**  
Print specific element in associative array

**Technique** –  
Optimisation of numbers

**Coding Style** – Use of constants

```

86.     $eps = 1.E-9;
87.     $results=[];
88.     for($percent = $percentStart; $percent <= $percentEnd; $percent++){
89.         for($number = $numberStart; $number <= $numberEnd; $number++){
90.             $percentFromNumber = $percent * $number/100;
91.             if(fmod($percentFromNumber,1) < $eps){
92.                 if (ceil(($number-
($number*($percent/100))*0.75))==floor(($number-($number*($percent/100))*0.75)){
93.                     $results[] = [
94.                         'percent' => $percent,
95.                         'number' => $number,
96.                         '%number' => $percentFromNumber
97.                     ];
98.                 }
99.             }
100.        }
101.    }
102.    $num = rand(1,185);
103.    $number1 = ($results[$num]['percent']);
104.    $number2 = ($results[$num]['number']);
105.    $answer = ($results[$num]['%number']);
106.    $updated1 = $string->getString("n1",$number1,$question);
107.    $updated2 = $string ->getString("n2",$number2,$updated1);
108.    echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $updated2. " </h4> ";
109.    $answerTwo = $number2*(($number1/100));
110.    $answerOne = (($number2-$answerTwo)*0.75);
111.    $solB="(".$number2.* ( ".$number1." X 0.75))";
112.    $solA="(".$number2." - ".$answerTwo.") X 0.75";
113.    $_SESSION['information']=array($updated2,$answerOne,$solA,$answerTwo
,$solB);
114.    require "generalNumberLongForm.php";
115.    }
116.    ?>

```

**Algorithm Design** – If question type is long answer question

*perimeter.php*

```

1. <?php //perimeter topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(14,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    $number1 = $num->getRandomNumber(10,100);
16.    $number2=$num->getRandomNumber(10,100);
17.    $updated1 = $string->getString("n1",$number1,$question);
18.    $updated2 = $string ->getString("n2",$number2,$updated1);
19.    if ($type == "m"){
20.        //calculates numbers, answers and solutions based on ID
21.        $number3=$num->getRandomNumber(10,100);
22.        $updated3 = $string ->getString("n3",$number3,$updated2);
23.        echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $updated3. " </h4> ";
24.        if ($id==30){
25.            $answer = $num->addition(($num->multiplyTwo($number1)) , ($num->multipl
yTwo($number2)));

```



```

26.         $optOne = $num->addition($num->addition($number1,$number2), $number1);
27. $optTwo = $num->addition($number1,$num->addition($number2,$num->additi
    n($number1,$num->addition($number1,$number2)));
28.         $sol="(".$number1." X 2) + (".$number2." X 2)";
29.     }
30.     if ($id==31){
31.         $answer = $num->addition($number1,$num->addition( $number2 , $number3));
32. $optOne = $num->addition($number1,$num->addition($number2, $num->additi
    on($number3,$number1)));
33.         $optTwo = $num->multiply(0.5,$num->multiply($number1 , $number2));
34.         $sol= $number1." + ".$number2." + ".$number3;
35.     }
36.     if ($id==32){
37.         $answer = $num->multiply($number1,4);
38.         $optOne = $num->multiply($number1,3);
39.         $optTwo = $num->multiply($number1,5);
40.         $sol= $number1." + ".$number1." + ".$number1." + ".$number1;
41.     }
42.     $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
43.     require "mcqOptionOrder.php";
44.     require "mcqDisplay.php";
45. }elseif ($type == "s"){
46.     //calculates numbers, answers and solutions based on ID
47.     $number3=$num->getRandomNumber(10,100);
48.     $updated3 = $string ->getString("n3",$number3,$updated2);
49. echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;'>".$updated3." </h4> ";
50.     if ($id==54){
51.         $answer = $num->addition(($num->multiplyTwo($number1)) , ($num->multipl
    yTwo($number2)));
52.         $sol="(".$number1." x 2) + (".$number2." x 2)";
53.     }
54.     if ($id==55){
55.         $answer = $num->addition($number1,$num->addition( $number2 , $number3))
    ;
56.         $sol=$number1." + ".$number2." + ".$number3;
57.     }
58.     if ($id==56){
59.         $answer = ($num->multiplyFour($number1));
60.         $sol=$number1." + ".$number1." + ".$number1." + ".$number1;
61.     }
62.     $_SESSION['information']=array($updated3,$answer,$sol);
63.     require "generalShortNumberForm.php";
64. }elseif($type == "l"){
65.     //calculates answers and solutions based on ID
66. echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;'>".$updated2." </h4> ";
67. $answerOne = $num->addition(($num->multiplyTwo($number1)) , ($num->multiply
    Two($number2)));
68.     $solA="(".$number1."X 2) + (".$number2."x 2)";
69.     $answerTwo = ($num->multiplyFour($number2));
70.     $solB="(".$number2."X 4)";
71.     $_SESSION['information']=array($updated2,$answerOne,$solA,$answerTwo,$solB)
    ;
72.     require "generalNumberLongForm.php";
73. }
74. ?>

```

probability.php

```

1. <?php //probability topic question generator
2.     session_start();
3.     require "connection.php";

```



```

4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(15,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    if ($type == "m"){
16.        $number1 = $num->getRandomNumber(1,6);
17.        $updated1 = $string->getString("n1",$number1,$question);
18.        //generates answer, sol and options based on ID
19.        if ($id==33){
20.            $answer = "1/6";
21.            $num = rand(2,4);
22.            $optOne = "$num"/6";
23.            $nume = rand(1,2);
24.            $optTwo = "$nume"/3";
25.            $sol=$number1." is one of the 6 number on the die";
26.        }
27.        if ($id==34){
28.            $answer = "1/4";
29.            $optOne = "2/4";
30.            $optTwo = "3/4";
31.            $sol="If you draw a space diagram, on the top it has the column heads a
nd tails for coin 1 and down the side it has the rows heads and tails for coin 2. Y
ou will notice that combination tail tail will only be one of the four possible out
comes.";
32.        }
33.        if ($id==130){
34.            $number2 = $num->getRandomNumber(3,10);
35.            $updated1 = $string->getString("n2",$number1,$updated1);
36.            $answer = 0.5**$number2;
37.            $optOne = 0.5*$number2;
38.            $optTwo = 0.5**($number2-1);
39.            $sol="The chance of rollowing a head is 0.5. There are ".$number2." tim
es, so it is 0.5 to the power of ".$number2;
40.        }
41.        echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated1." </h4> ";
42.        $_SESSION['information']=array($updated1,$answer,$sol,$optOne,$optTwo);
43.        require "mcqOptionOrder.php";
44.        require "mcqDisplay.php";
45.    }elseif ($type == "s"){
46.
47.
48.        //generates answer, sol and additional numbers based on ID
49.        if ($id==57){
50.            $number1 = $num->getRandomNumber(2,50);
51.            $number2=$num->getRandomNumber(2,50);
52.            $number3=$num->getRandomNumber(2,50);
53.            $updated1 = $string->getString("n1",$number1,$question);
54.            $updated2 = $string ->getString("n2",$number2,$updated1);
55.            $updated3 = $string ->getString("n3",$number3,$updated2);
56.            $question = $updated3;
57.            echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated3." </h4> ";
58.            $answer = $num->division($number1 , $num->addition($number1,$num->addit
ion($number2,$number3))) ;
59.            $sol = "The total number of balls is " .($number1 + $number2 + $number3)
." and there are ".$number1." red ones, which makes it ".$number1."/".($number1 + $
number2 + $number3);
60.

```

```

61.         }elseif ($id==58){
62.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$question.\" </h4> ";
63.             $answer = 5;
64.             $sol="2 and 6, 3 and 5, 4 and 4, 5 and 3, 6 and 2";
65.         }elseif ($id==59){
66.             echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$question.\" </h4> ";
67.             $answer = 0.5;
68.             $sol = "The only two possible outcomes are heads or tails, so a 50 50 c
hance.";
69.         }
70.         $_SESSION['information']=array($question,$answer,$sol);
71.         require "generalShortStringForm.php";
72.     }elseif($type == "1"){
73.         $number1 = $num->getRandomNumber(10,100);
74.         $number2=$num->getRandomNumber(10,100);
75.         $number3=$num->getRandomNumber(10,100);
76.         if (($num->addition($number2,$number3))>$number1){ //makes sure denominator bigger
than denominator. if not, regenerates numbers
77.             while (($num->addition($number2,$number3))>$number1) {
78.                 $number1 = $num->getRandomNumber(10,100);
79.                 $number2=$num->getRandomNumber(10,100);
80.                 $number3=$num->getRandomNumber(10,100); 81.
            }}
82.         $updated1 = $string->getString("n1",$number1,$question);
83.         $updated2 = $string ->getString("n2",$number2,$updated1);
84.         $updated3 = $string ->getString("n3",$number3,$updated2);
85.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$updated3.\" </h4> ";
86.         $answerOne = $number2."/\".$number1;
87.         $solA="Total number is \".$number1.\" and \".$number2.\" are selected, hence th e
solution is: \".$number2."/\".$number1;
88.         $answerTwo = $number3."/\".$number1;
89.         $solB="Total number is \".$number1.\" and \".$number3.\" are selected, hence th e
solution is: \".$number3."/\".$number1;
90.         $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo,$solB)
;
91.         require "probabilityLongForm.php";
92.     }
93. }>

```

probabilityLongForm.php

```

1. <!DOCTYPE html>
2. <html> <!--form if topic is probability for long question -->
3.     <head>
4.     </head>
5.     <body style="background-color:#add8e6;">
6.         <div style="text-align: center;">
7.             <div style="display: inline-block; text-align: left;">
8.                 <form name ="form1" method ="post" action ="longQuestionCheck.php">
9.                     <br>
10.                    <p>Part A</p>
11.                    <input type="text" name="answerOne" placeholder="Write in fract
ion form i.e. 1/3"/>
12.                    <p>
13.                    <p>Reasoning for part A</p>
14.                    <input type="text" name="solOne" />
15.                    <p></p>
16.                    <p>Part B</p>
17.                    <input type="text" name="answerTwo" placeholder="Write in fract
ion form i.e. 1/3"/>
18.                    <p>

```

```

19.         <p>Reasoning for part B</p>
20.         <input type="text" name="solTwo" />
21.         <p></p>
22.         <input type = "Submit" Name = "Submit1" VALUE = "Submit">
23.     </form>
24. </div>
25. </div>
26. </body>
27. </html>

```

rates.php

```

1. <?php //rates topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(19,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    //generates numbers based on user understanding
16.    if ($_SESSION['userScoreData'][8]==1){
17.        $number1 = $num->getRandomNumber(5,100);
18.        $number2=$num->getRandomNumber(5,100);
19.        $number3=$num->getRandomNumber(5,100);
20.    }elseif ($_SESSION['userScoreData'][8]==2){
21.        $number1 = $num->getRandomNumber(50,200);
22.        $number2=$num->getRandomNumber(50,200);
23.        $number3=$num->getRandomNumber(50,200);
24.    }elseif ($_SESSION['userScoreData'][8]==3){
25.        $number1 = $num->getRandomNumber(100,300);
26.        $number2=$num->getRandomNumber(100,300);
27.        $number3=$num->getRandomNumber(100,300);
28.    }
29.    if ($num->division($number2,$number1) == 1){//makes sure 'rate' is not one in o
30.        while ($num->division($number2,$number1) == 1){
31.            if ($_SESSION['userScoreData'][8]==1){
32.                $number1 = $num->getRandomNumber(5,100);
33.                $number2=$num->getRandomNumber(5,100);
34.                $number3=$num->getRandomNumber(5,100);
35.            }elseif ($_SESSION['userScoreData'][8]==2){
36.                $number1 = $num->getRandomNumber(50,200);
37.                $number2=$num->getRandomNumber(50,200);
38.                $number3=$num->getRandomNumber(50,200);
39.            }elseif ($_SESSION['userScoreData'][8]==3){
40.                $number1 = $num->getRandomNumber(100,300);
41.                $number2=$num->getRandomNumber(100,300);
42.                $number3=$num->getRandomNumber(100,300);
43.            }
44.        }
45.    }
46.    if ($num->intDivMult($number2,$number1,$number3)==false); { //makes sure gcf
47.        while ($num->intDivMult($number2,$number1,$number3)==false) {
48.            if ($_SESSION['userScoreData'][8]==1){
49.                $number1 = $num->getRandomNumber(5,100);
50.                $number2=$num->getRandomNumber(5,100);
51.                $number3=$num->getRandomNumber(5,100);

```

```

52.         }elseif ($_SESSION['userScoreData'][8]==2){
53.             $number1 = $num->getRandomNumber(50,200);
54.             $number2=$num->getRandomNumber(50,200);
55.             $number3=$num->getRandomNumber(50,200);
56.         }elseif ($_SESSION['userScoreData'][8]==3){
57.             $number1 = $num->getRandomNumber(100,300);
58.             $number2=$num->getRandomNumber(100,300);
59.             $number3=$num->getRandomNumber(100,300);
60.         }
61.         if ($num->division($number2,$number1) == 1){
62.             while ($num->division($number2,$number1) == 1){
63.                 if ($_SESSION['userScoreData'][8]==1){
64.                     $number1 = $num->getRandomNumber(5,100);
65.                     $number2=$num->getRandomNumber(5,100);
66.                     $number3=$num->getRandomNumber(5,100);
67.                 }elseif ($_SESSION['userScoreData'][8]==2){
68.                     $number1 = $num->getRandomNumber(50,200);
69.                     $number2=$num->getRandomNumber(50,200);
70.                     $number3=$num->getRandomNumber(50,200);
71.                 }elseif ($_SESSION['userScoreData'][8]==3){
72.                     $number1 = $num->getRandomNumber(100,300);
73.                     $number2=$num->getRandomNumber(100,300);
74.                     $number3=$num->getRandomNumber(100,300);
75.                 }
76.             }
77.         }
78.     }
79. }
80. }
81. $updated1 = $string->getString("n1",$number1,$question);
82. $updated2 = $string ->getString("n2",$number2,$updated1);
83. $updated3 = $string ->getString("n3",$number3,$updated2);
84. if ($type == "m"){
85.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3."</h4> ";
86.     $answer = $num->multiply(($num->division($number2,$number1)), $number3);
87.     $sol="(".$number2." / ".$number1.") x ".$number3;
88.     $optOne = $num->optionOne($answer);
89.     $optTwo = $num->optionTwo($answer);
90.     $_SESSION['information']=array($updated3,$answer,$sol,$optOne,$optTwo);
91.     require "mcqOptionOrder.php";
92.     require "mcqDisplay.php";
93. }elseif ($type == "s"){
94.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3."</h4> ";
95.     $answer = $num->multiply(($num->division($number2,$number1)), $number3);
96.     $sol= "(".$number2." / ".$number1.") x ".$number3;
97.     $_SESSION['information']=array($updated3,$answer,$sol);
98.     require "generalShortNumberForm.php";
99. }elseif($type == "l"){
100.     $mult = rand(2,5);
101.     $number4 = $num->multiply($number3, $mult);
102.     $updated4 = $string->getString("n4",$number4,$updated3);
103.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated4."</h4> ";
104.     $answerOne = $num->multiply(($num->division($number2,$number1)), $num
ber3);
105.     $solA = "(".$number2." / ".$number1.") x ".$number3;
106.     $answerTwo = $num->multiply($answerOne, $mult);
107.     $solB = "The common factor between $number3 and $number4 is $mult so
you do (".$number2." / ".$number1.") x ".$number3." x ".$mult;
108.     $_SESSION['information']= array($updated4,$answerOne,$solA,$answerTw
o,$solB);
109.     require "generalNumberLongForm.php";
110. }
111. ?>

```

ratio.php

```
1. <?php //ratio topic question generator
2. function gcd($a, $b){ //url ref - https://www.geeksforgeeks.org/c-program-
   find-gcd-hcf-two-numbers/
3.     // Everything divides 0
4.     if ($a == 0) {
5.         return $b;
6.     }if ($b == 0){
7.         return $a;
8.     }
9.     // base case
10.    if($a == $b){
11.        return $a ;
12.    }
13.    // a is greater
14.    if($a > $b) {
15.        return gcd( $a-$b , $b ) ;
16.    }
17.    return gcd( $a , $b-$a ) ;
18. }
19. session_start();
20. require "connection.php";
21. require "assignment.php";
22. require "number.php";
23. require "string.php";
24. require "titlePage.php";
25. $conn = connection::getConn();
26. $s = new assignment;
27. $num = new number;
28. $string = new stringQuestion;
29. $type = $_GET['type'];

30. $question=$s->question(9,$type,$_SESSION['userScoreData'][8]);
31. $id= $s->id($question);
32. echo "hi";
33. if ($type == "m"){
34.     $number1 = $num->getRandomNumber(1,50);
35.     $number2=$num->getRandomNumber(1,100);
36.     $updated1 = $string->getString("n1",$number1,$question);
37.     $updated2 = $string ->getString("n2",$number2,$updated1);
38. echo "<h4 align=center style='margin-left:30%;margin-
   right:30%;>".$updated2."</h4> ";
39. echo "<br>";
40. //calcs options, answer and sol based on user input
41. if ($id==15){
42.     $optOne = $num1." to ".$num2;
43.     $answer = $num2." : ".$num1;
44.     $optTwo = $num1." : ".$num2;
45. $sol = "Three of the four options were ".$num1."/ or to or : ".$num3 ." which
   means there were they same. Because, / means to and to means : and / mean s : . Hence
   the only different ratio was ".$num2"." : ".$num1";
46. }
47. if ($id==16){
48.     $optOne = $num1." to ".$num2;
49.     $optTwo = $num2." : ".numberToString($number1+$number2);
50.     $answer = $num2." : ".$num1;
51. $sol = "The question is asking for girls to boys, there are ".$num2." girls and
   ".$num1." boys, and hence the ratio is: ".$num2"." : ".$num1";
52. }
53. if ($id==17){
54.     $gcd = gcd($number1,$number2);
55.     $optOne = $gcd." to ".$number2/($gcd);
56.     $optTwo = ($gcd * (rand(2,4))).":".($gcd * (rand(8,11)));
57.     $ran = (rand(4,20));
58.     $answer = (($number1/$gcd)*$ran).":".((($number2/$gcd)*$ran);
```

Technique- Encapsulation

```

59.         $sol = "The common factor between the question and answer was: ".$gcd"
60.     };
61.     }
62.     $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
63.     require "mcqOptionOrder.php";
64.     require "mcqDisplay.php";
65. }elseif ($type == "s"){
66.     //calcs number based on understadning
67.     if ($_SESSION['userScoreData'][8]==1){
68.         $number1 = $num->getRandomNumber(5,100);
69.         $number2=$num->getRandomNumber(5,100);
70.         $number3=$num->getRandomNumber(5,100);
71.     }elseif ($_SESSION['userScoreData'][8]==2){
72.         $number1 = $num->getRandomNumber(50,200);
73.         $number2=$num->getRandomNumber(50,200);
74.         $number3=$num->getRandomNumber(50,200);
75.     }elseif ($_SESSION['userScoreData'][8]==3){
76.         $number1 = $num->getRandomNumber(100,300);
77.         $number2=$num->getRandomNumber(100,300);
78.         $number3=$num->getRandomNumber(100,300);
79.     }
80.     //makes sure number 1 is bigger
81.     if (($number2+$number3)>$number1){
82.         while (($number2+$number3)>$number1) {
83.             if ($_SESSION['userScoreData'][8]==1){
84.                 $number1 = $num->getRandomNumber(5,100);
85.                 $number2=$num->getRandomNumber(5,100);
86.                 $number3=$num->getRandomNumber(5,100);
87.             }elseif ($_SESSION['userScoreData'][8]==2){
88.                 $number1 = $num->getRandomNumber(50,200);
89.                 $number2=$num->getRandomNumber(50,200);
90.                 $number3=$num->getRandomNumber(50,200);
91.             }elseif ($_SESSION['userScoreData'][8]==3){
92.                 $number1 = $num->getRandomNumber(100,300);
93.                 $number2=$num->getRandomNumber(100,300);
94.                 $number3=$num->getRandomNumber(100,300);
95.             }
96.         }
97.     }
98.     if(is_int($number1/($number2+$number3))==false) {
99.         while (is_int($number1/($number2+$number3))==false){
100.             if ($_SESSION['userScoreData'][8]==1){
101.                 $number1 = $num->getRandomNumber(5,200);
102.                 $number2=$num->getRandomNumber(5,200);
103.                 $number3=$num->getRandomNumber(5,200);
104.             }elseif ($_SESSION['userScoreData'][8]==2){
105.                 $number1 = $num->getRandomNumber(50,400);
106.                 $number2=$num->getRandomNumber(50,400);
107.                 $number3=$num->getRandomNumber(50,400);
108.             }elseif ($_SESSION['userScoreData'][8]==3){
109.                 $number1 = $num->getRandomNumber(100,800);
110.                 $number2=$num->getRandomNumber(100,800);
111.                 $number3=$num->getRandomNumber(100,800);
112.             }
113.         }
114.     }
115.     $updated1 = $string->getString("n1",$number1,$question);
116.     $updated2 = $string ->getString("n2",$number2,$updated1);
117.     $updated3 = $string ->getString("n3",$number3,$updated2);
118.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;'>".$updated3."</h4> ";
119.     $answer = $num->multiply($num->division($number1,($num->addition($number2,$number3))),$number2);
120.     $sol="((".$number1." / ".$number2.") + ".$number3.") x (".$number2.")";

```

**Technique**  
Optimisation of  
numbers

```

121.         $_SESSION['information']=array($updated3,$answer,$sol);
122.         require "generalShortNumberForm.php";
123.     }elseif($type == "1"){
124.         //generates numbers based on difficulty
125.         if ($_SESSION['userScoreData'][8]==1){
126.             $number1 = $num->getRandomNumber(10,30);
127.             $number2=$num->getRandomNumber(1,5);
128.             $number3=$num->getRandomNumber(1,5);
129.         }elseif ($_SESSION['userScoreData'][8]==2){
130.             $number1 = $num->getRandomNumber(20,50);
131.             $number2=$num->getRandomNumber(1,10);
132.             $number3=$num->getRandomNumber(1,10);
133.         }elseif ($_SESSION['userScoreData'][8]==3){
134.             $number1 = $num->getRandomNumber(50,300);
135.             $number2=$num->getRandomNumber(1,25);
136.             $number3=$num->getRandomNumber(1,25);
137.         }
138.         $factor = $number1/$number2;
139.         $number4 = $factor * (rand(2,15));
140.         if ((is_int($factor))==false ){ //makes sure factor is even, else
regenerates numbers
141.             while ((is_int($factor))==false ) {
142.                 if ($_SESSION['userScoreData'][8]==1){
143.                     $number1 = $num->getRandomNumber(10,30);
144.                     $number2=$num->getRandomNumber(1,5);
145.                     $number3=$num->getRandomNumber(1,5);
146.                 }elseif ($_SESSION['userScoreData'][8]==2){
147.                     $number1 = $num->getRandomNumber(10,50);
148.                     $number2=$num->getRandomNumber(1,10);
149.                     $number3=$num->getRandomNumber(1,10);
150.                 }elseif ($_SESSION['userScoreData'][8]==3){
151.                     $number1 = $num->getRandomNumber(10,300);
152.                     $number2=$num->getRandomNumber(1,25);
153.                     $number3=$num->getRandomNumber(1,25);
154.                 }
155.                 $factor = $num->division($number1,$number2);
156.             }
157.         }
158.         $updated1 = $string->getString("n1",$number1,$question);
159.         $updated2 = $string ->getString("n2",$number2,$updated1);
160.         $updated3 = $string ->getString("n3",$number3,$updated2);
161.         $updated4 = $string ->getString("n4",$number4,$updated3);
162.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $updated4." </h4> ";
163.         $answerOne = $num->multiply($factor, $number3);
164.         $answerTwo = $num->division($number4,$factor);
165.         $solA=$factor." x ". $number3;
166.         $solB=$number4." / ".$factor;
167.         $_SESSION['information']=array($updated4,$answerOne,$solA,$answerTwo
,$solB);
168.         require "generalNumberLongForm.php";
169.     }
170.     ?>

```



roundAndEstimate.php

```
1. <?php //rounding and estimation topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(12,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    if ($type == "m"){
16.        $number1 = $num->getRandomNumber(50,1000);
17.        $number2=$num->getRandomNumber(1,100);
18.        $updated1 = $string->getString("n1",$number1,$question);
19.        $updated2 = $string->getString("n2",$number2,$updated1);
20.        $ansOne = 0;
21.        $ansTwo = 0;
22.        //generates answers, options and sol based on ID
23.        if ($id==24){
24.            echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated2."</h4> ";
25.            $ansOne = round($number1, -1); //0 whole number -1 tens -2 hundreds -
3 thousands etc.
26.            $ansTwo = round($number2, -1);
27.            if ($ansOne % 50 != 0 ){
28.                $ansOne = round($number1, -2);
29.            }
30.            if ($ansTwo % 50 != 0 ){
31.                $ansTwo = round($number2, -2);
32.            }
33.            echo "<br>";
34.            $answer = $ansOne." x ".$ansTwo;
35.            $optOne = ($ansOne+50)." x " .($ansTwo+50);
36.            $optTwo = ($ansOne-50)." x " .($ansTwo-50);
37.            $sol= "For the best approximation, you should do: ".$ansOne." x ".$
ansTwo";
38.        }
39.        if ($id==25){
40.            $number1 = $num->getRandomNumber(50,100);
41.            $updated1 = $string->getString("n1",$number1,$question);
42.            echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated1."</h4> ";
43.            $answer=round($number1, -1);;
44.            $optOne = $answer + 10;
45.            $optTwo = $answer - 10;
46.            $sol="Round to nearest 10. If 2nd digit greater than or equal to 5, rou
nd up else round down";
47.        }
48.        if ($id==143){
49.            $number1 = $num->getRandomNumber(100,10000);
50.            $updated1 = $string->getString("n1",$number1,$question);
51.            echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated1."</h4> ";
52.            $answer=round($number1, -2);;
53.            $optOne = $answer + 10;
54.            $optTwo = $answer - 10;
55.            $sol="Round to nearest 100. If 2nd digit greater than or equal to 5, ro
und up else round down";
56.        }
    }
```



```

57.     $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
58.     require "mcqOptionOrder.php";
59.     require "mcqDisplay.php";
60. }elseif ($type == "s"){
61.     if ($id==145){
62.         $number1 = $num->getRandomNumber(100,1000);
63.         $number2=$num->getRandomNumber(100,1000);
64.     }else{
65.         $number1 = $num->getRandomNumber(25,100);
66.         $number2=$num->getRandomNumber(25,100);
67.     }
68.     $updated1 = $string->getString("n1",$number1,$question);
69.     $updated2 = $string ->getString("n2",$number2,$updated1);
70.     echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>". $updated2." </h4> ";
71.     $ansOne = round($number1, -1); //0 whole number -1 tens -2 hundreds -
    3 thousands etc.
72.     $ansTwo = round($number2, -1);
73.     if ($ansOne % 50 != 0 ){
74.         $ansOne = round($number1, -2);
75.     }
76.     if ($ansTwo % 50 != 0 ){
77.         $ansTwo = round($number2, -2);
78.     }
79.     echo "<br>";
80.     $answer = $ansOne*$ansTwo;
81.     $sol=$ansOne." x ".$ansTwo;
82.     $_SESSION['information']=array($updated2,$answer,$sol);
83.     require "generalShortNumberForm.php";
84. }elseif($type == "l"){
85.     //gen numbers based on user understanding
86.     if ($_SESSION['userScoreData'][8]==1){
87.         $number1 = $num->getRandomNumber(5,75);
88.         $number2=$num->getRandomNumber(5,75);
89.         $number3=$num->getRandomNumber(5,75);
90.     }elseif ($_SESSION['userScoreData'][8]==2){
91.         $number1 = $num->getRandomNumber(50,100);
92.         $number2=$num->getRandomNumber(50,100);
93.         $number3=$num->getRandomNumber(50,100);
94.     }elseif ($_SESSION['userScoreData'][8]==3){
95.         $number1 = $num->getRandomNumber(50,1000);
96.         $number2=$num->getRandomNumber(50,1000);
97.         $number3=$num->getRandomNumber(50,1000);
98.     }
99.     $updated1 = $string->getString("n1",$number1,$question);
100.     $updated2 = $string ->getString("n2",$number2,$updated1);
101.     $updated3 = $string ->getString("n3",$number3,$updated2);
102.     // $frac1 = $num1."/".$num2;
103.     $updated = str_replace("n1", "$number1", $question); //url ref - ht
    tps://www.tutorialrepublic.com/faq/how-to-replace-a-word-inside-a-string-in-
    php.php
104.     $updated2 = str_replace("n2", "$number2", $updated);
105.     $question = str_replace("n3", "$number3", $updated2);
106.     echo "<h4 align=center style='margin-left:30%;margin-
    right:30%;>". $question." </h4> ";
107.     $ansOne = round($number1, -1); //0 whole number -1 tens -
    2 hundreds -3 thousands etc.
108.     $ansTwo = round($number2, -1);
109.     $ansThree = round($number3, -1);
110.     if ($ansOne % 50 != 0 ){
111.         $ansOne = round($number1, -2);
112.     }
113.     if ($ansTwo % 50 != 0 ){
114.         $ansTwo = round($number2, -2);
115.     }
116.     if ($ansThree % 50 != 0 ){

```

```

117.         $ansThree = round($number3, -2);
118.     }
119.     $answerOne = $ansOne * $ansTwo;
120.     $solA = $ansOne ." X " . $ansTwo;
121.     $answerTwo = ($ansOne * $ansTwo)+($ansOne * $ansThree);
122.     $solB = "(".$ansOne ." X " . $ansTwo.") + (".$ansOne ." X " . $ansThree
    e.");";
123.     $_SESSION['information']=array($question,$answerOne,$solA,$answerTwo
    , $solB);
124.     require "generalNumberLongForm.php";
125. }
126. ?>

```

*sequence.php*

```

1. <?php //sequence topic question generator
2.     //gen random no
3.     function genNum(){
4.         return rand(2,19);
5.     }
6.     //gen sequence
7.     function nexNumSeq(){
8.         $dif = genNum();
9.         $start = genNum();
10.        $sequence = array($start,($start+$dif),($start+$dif+$dif));
11.        $_SESSION['sequence']=$sequence;
12.        $_SESSION['answer']=$start+$dif+$dif+$dif;
13.        $_SESSION['dif']=$dif;
14.        $_SESSION['sol']=" $start"."+"."$dif"."+"."$dif"."+"."$dif";
15.        return $sequence;
16.    }
17.    session_start();
18.    require "connection.php";
19.    require "assignment.php";
20.    require "number.php";
21.    require "string.php";
22.    require "titlePage.php";
23.    $conn = connection::getConn();
24.    $s = new assignment;
25.    $num = new number;
26.    $string = new stringQuestion;
27.    $type = $_GET['type'];
28.    $question=$s->question(7,$type,$_SESSION['userScoreData'][8]);
29.    $id= $s->id($question);
30.    if ($type == "m"){
31.        $seq = nexNumSeq();
32.        echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". $question." </h4> ";
33.        echo "<h4 align=center style='margin-left:30%;margin-right:30%;>". $seq[0]." " . $seq[1]." " . $seq[2]." </h4> ";
34.        $optOne = ($_SESSION['answer']+rand(1,5));
35.        $optTwo = ($_SESSION['answer']-rand(1,5));
36.        $_SESSION['optionOne']=$optOne;
37.        $_SESSION['optionTwo']=$optTwo;
38.        $answer = $_SESSION['answer'];
39.        require "mcqOptionOrder.php";
40.        require "sequenceMCQForm.php";
41.    }elseif ($type == "s"){
42.        //gen numbers based on user understanding
43.        if ($_SESSION['userScoreData'][8]==1){
44.            $number1 = $num->getRandomNumber(5,75);
45.            $number2=$num->getRandomNumber(5,50);
46.        }elseif ($_SESSION['userScoreData'][8]==2){
47.            $number1 = $num->getRandomNumber(5,100);
48.            $number2=$num->getRandomNumber(5,75);

```

```

49.         }elseif ($_SESSION['userScoreData'][8]==3){
50.             $number1 = $num->getRandomNumber(20,500);
51.             $number2=$num->getRandomNumber(20,300);
52.         }
53.         if ($number1<$number2){ //makes sure num2 bigger than num1
54.             while ($number1<$number2){
55.                 if ($_SESSION['userScoreData'][8]==1){
56.                     $number1 = $num->getRandomNumber(5,75);
57.                     $number2=$num->getRandomNumber(5,50);
58.                 }elseif ($_SESSION['userScoreData'][8]==2){
59.                     $number1 = $num->getRandomNumber(5,100);
60.                     $number2=$num->getRandomNumber(5,75);
61.                 }elseif ($_SESSION['userScoreData'][8]==3){
62.                     $number1 = $num->getRandomNumber(20,500);
63.                     $number2=$num->getRandomNumber(20,300);
64.                 }
65.             }
66.         }
67.         $number3=$num->getRandomNumber(1,100);
68.         $dif = $num->subtraction($number1,$number2);
69.         $d2 = $num->addition($dif,$number3);
70.         $number3=$num->getRandomNumber(1,100);
71.         if (is_int(($number1-
$number2)/$number3)==false){ //makes sure answer is int, else regenerates
72.             while (is_int(($number1-$number2)/$number3)==false) {
73.                 if ($_SESSION['userScoreData'][8]==1){
74.                     $number1 = $num->getRandomNumber(5,75);
75.                     $number2=$num->getRandomNumber(5,50);
76.                 }elseif ($_SESSION['userScoreData'][8]==2){
77.                     $number1 = $num->getRandomNumber(5,100);
78.                     $number2=$num->getRandomNumber(5,75);
79.                 }elseif ($_SESSION['userScoreData'][8]==3){
80.                     $number1 = $num->getRandomNumber(20,500);
81.                     $number2=$num->getRandomNumber(20,300);
82.                 }
83.                 $number3=$num->getRandomNumber(1,100);
84.                 if ($number1<$number2){
85.                     while ($number1<$number2){
86.                         if ($_SESSION['userScoreData'][8]==1){
87.                             $number1 = $num->getRandomNumber(5,75);
88.                             $number2=$num->getRandomNumber(5,50);
89.                         }elseif ($_SESSION['userScoreData'][8]==2){
90.                             $number1 = $num->getRandomNumber(5,100);
91.                             $number2=$num->getRandomNumber(5,75);
92.                         }elseif ($_SESSION['userScoreData'][8]==3){
93.                             $number1 = $num->getRandomNumber(20,500);
94.                             $number2=$num->getRandomNumber(20,300);
95.                         }
96.                     }
97.                 }
98.             }
99.         }
100.         $updated1 = $string->getString("n1",$number1,$question);
101.         $updated2 = $string ->getString("n2",$number2,$updated1);
102.         $updated3 = $string ->getString("n3",$number3,$updated2);
103.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>". $updated3. "</h4> ";
104.         $answer = ($num->division($num->subtraction($number1,$number2),$numb
er3))+1;
105.         $sol = "The nth term sequence is: $number2 + $number3"."n = $number1
. So rearrange, and solve for n. n = ((". "$number1"." - ". "$number2"." ) / ". "$numb
er3"." )";
106.         $_SESSION['information']=array($updated3,$answer,$sol);
107.         require "generalShortNumberForm.php";
108.     }elseif($type == "1"){
109.         //gen numbers based on user understanding

```

```

110.         if ($_SESSION['userScoreData'][8]==1){
111.             $number1 = $num->getRandomNumber(5,75);
112.             $number2=$num->getRandomNumber(5,50);
113.         }elseif ($_SESSION['userScoreData'][8]==2){
114.             $number1 = $num->getRandomNumber(5,100);
115.             $number2=$num->getRandomNumber(5,75);
116.         }elseif ($_SESSION['userScoreData'][8]==3){
117.             $number1 = $num->getRandomNumber(20,500);
118.             $number2=$num->getRandomNumber(20,300);
119.         }
120.         if ($number1<$number2){
121.             while ($number1<$number2){
122.                 if ($_SESSION['userScoreData'][8]==1){
123.                     $number1 = $num->getRandomNumber(5,75);
124.                     $number2=$num->getRandomNumber(5,50);
125.                 }elseif ($_SESSION['userScoreData'][8]==2){
126.                     $number1 = $num->getRandomNumber(5,100);
127.                     $number2=$num->getRandomNumber(5,75);
128.                 }elseif ($_SESSION['userScoreData'][8]==3){
129.                     $number1 = $num->getRandomNumber(20,500);
130.                     $number2=$num->getRandomNumber(20,300);
131.                 }
132.             }
133.         }
134.         $number3=$num->getRandomNumber(1,100);
135.         $dif = $num->subtraction($number1,$number2);
136.         $d2 = $num->addition($dif,$number3);
137.         $number3=$num->getRandomNumber(1,100);
138.         if (is_int(($number1-$number2)/$number3)==false){
139.             while (is_int(($number1-$number2)/$number3)==false) {
140.                 if ($_SESSION['userScoreData'][8]==1){
141.                     $number1 = $num->getRandomNumber(5,75);
142.                     $number2=$num->getRandomNumber(5,50);
143.                 }elseif ($_SESSION['userScoreData'][8]==2){
144.                     $number1 = $num->getRandomNumber(5,100);
145.                     $number2=$num->getRandomNumber(5,75);
146.                 }elseif ($_SESSION['userScoreData'][8]==3){
147.                     $number1 = $num->getRandomNumber(20,500);
148.                     $number2=$num->getRandomNumber(20,300);
149.                 }
150.                 $number3=$num->getRandomNumber(1,100);
151.                 if ($number1<$number2){
152.                     while ($number1<$number2){
153.                         if ($_SESSION['userScoreData'][8]==1){
154.                             $number1 = $num->getRandomNumber(5,75);
155.                             $number2=$num->getRandomNumber(5,50);
156.                         }elseif ($_SESSION['userScoreData'][8]==2){
157.                             $number1 = $num->getRandomNumber(5,100);
158.                             $number2=$num->getRandomNumber(5,75);
159.                         }elseif ($_SESSION['userScoreData'][8]==3){
160.                             $number1 = $num->getRandomNumber(20,500);
161.                             $number2=$num->getRandomNumber(20,300);
162.                         }
163.                     }
164.                 }
165.             }
166.         }
167.         $updated1 = $string->getString("n1",$number1,$question);
168.         $updated2 = $string->getString("n2",$number2,$updated1);
169.         $updated3 = $string->getString("n3",$number3,$updated2);
170.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$updated3." </h4> ";
171.         $answerOne = ($num->division($num->subtraction($number1,$number2),$n
umber3))+1;

```

```

172.         $solA="The nth term sequence is: $number2 + $number3"."n = $number1.
    So rearrange, and solve for n. n = ((".$number1"." - ".$number2"." ) / ".$number3"." )";
173.         $answerTwo = $num->addition($number2,($num->multiplyTwo($number3)));
174.         $solB="Using the nth term sequence calculated above, use ".$number2.
    " + (".$number3."*2)";
175.         $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo
    ,$solB);
176.         require "generalNumberLongForm.php";
177.     }
178.     ?>

```

*sequenceMCQ.php*

```

1. <html> <!--check if mcq answer chosen for sequence topic is correct -->
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="stylesheet"
    type="text/css">
5.         <style>
6.             body {font-family: "Raleway", Arial, sans-serif}
7.             .w3-row img {margin-bottom: -8px}
8.         </style>
9.     </head>
10.    <body style="background-color:#add8e6;">
11.    </body>
12. </html>
13. <?php
14.     session_start();
15.     $question = "The sequence was: ".$_SESSION['sequence'][0]." ".$_SESSION['sequence']
    [1]." ".$_SESSION['sequence'][2]." What is the next number?";
16.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$question."</h4> ";
17.     $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+1;
18.     $reason = $_SESSION['information'][2];
19.     $answerStatus = "wrong";
20.     $options = "correct";
21.     if (isset($_POST['Submit1'])) {
22.         $selected_radio = $_POST['Opt'];
23.         if ($selected_radio == 'ans') { //checks if answer chosen is correct, then
    display correct, green tick, add 1 to array of scores
24.             $answerStatus = "correct";
25.             $display = "✔";
26.             array_push($_SESSION['userScoreData'][9],1);
27.             $score = 1;
28.         }
29.         else if ($selected_radio == 'opte') { //if answer chosen is wrong, then
    display incorrect, red cross, add 0 to array of scores
30.             $options = 'wrong';
31.             $answerChosen = $_SESSION['optionOne'];
32.             array_push($_SESSION['userScoreData'][9],0);
33.             $display = "✗";
34.             $score = 0;
35.         }
36.         else if ($selected_radio == 'opten') { //if answer chosen is wrong, then
    display incorrect, red cross, add 0 to array of scores
37.             $options = 'wrong';
38.             $answerChosen = $_SESSION['optionTwo'];
39.             array_push($_SESSION['userScoreData'][9],0);
40.             $display = "✗";
41.             $score = 0;
42.         }
43.     }

```

```

44.     echo "<br>";
45.     $answerCho = "The answer you chose was : ".$answerStatus;
46.     echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$answerCho."</h4> ";
47.
48.     require "connection.php";
49.     $conn = connection::getConn();
50.
51.     $num = $_SESSION['userScoreData'][4];
52.     //insert question into db
53.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('$question', $n
um)";
54.     if ($conn->query($sql) == TRUE) {
55.     } else {
56.         echo "Connection error";
57.     }
58.     //get questionid of this question
59.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
60.     $result = $conn->query($sql);
61.     if ($conn->query($sql) == TRUE) {
62.         if ($result->num_rows > 0) {
63.             while($row = $result->fetch_assoc()) {
64.                 $qid = $row["questionsID"];
65.             } }else {
66.                 echo "Connection error";
67.             }
68.             $sid = $_SESSION['userData'][2];
69.             //insert this score into db
70.             $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`) V
ALUES ('$sid', '$qid', '$score', '$reason')";
71.             if ($conn->query($sql2) ==TRUE) {
72.             } else {
73.                 echo "Connection error";
74.             }
75.     }
76.     <html><center><span><?php echo $display?></span></center></html>
77.     <?php
78.     if ($answerStatus=="wrong"){//if answer is wrong, displays important info
79.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The answer you chose was: ".$answerChosen."</h4> ";
80.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The correct answer was : ".$_SESSION['answer']."</h4> ";
81.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". "The correct method is: ".$_SESSION['sol']."</h4> ";
82.     }
83.     require "backToSort.php";
84.     ?>

```

## sequenceMCQForm.php

```

1. <html> <!--MCQ form for topic sequences-->
2.     <body style="background-color:#add8e6;">
3.         <div style="text-align: center;">
4.             <div style="display: inline-block; text-align: left;">
5.                 <form name = "form1" method = "post" action = "sequenceMCQCheck.php">
6.                     <input type = 'Radio' Name = 'Opt' value= '<?php echo $fstatus;?
7. >' ><?php if ($first==$one){echo $answer;} elseif($first==$two){echo $optOne;} else
8. if($first==$three) {echo $optTwo;} ?><br>
9. <input type = 'Radio' Name = 'Opt' value= '<?php echo $sstatus;?
10. >' ><?php if ($second==$one){echo $answer;} elseif($second==$two){echo $optOne;} el
11. seif($second==$three) {echo $optTwo;} ?><br>
12. <input type = 'Radio' Name = 'Opt' value= '<?php echo $tstatus;?
13. >' ><?php if ($third==$one){echo $answer;} elseif($third==$two){echo $optOne;} else
14. if($third==$three) {echo $optTwo;} ; ?><br>
15. <p>
16.     <input type = "Submit" Name = "Submit1" VALUE = "Submit">
17. </form>
18. </div>
19. </div>
20. </body>
21. </html>

```

## shortQuestionCheck.php

```

1. <html>
2.     <head>
3.         <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
4.         <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
5. t" type="text/css">
6.         <style>
7.             body {font-family: "Raleway", Arial, sans-serif}
8.             .w3-row img {margin-bottom: -8px}
9.         </style>
10.     </head>
11.     <body style="background-color:#add8e6;">
12.     </body>
13. </html>
14. <?php //check short answer questions
15. session_start();
16. $_SESSION['userScoreData'][1]=$_SESSION['userScoreData'][1]+1;
17. $quest = "The question was : ".$_SESSION['information'][0];
18. echo "<h4 align=center style='margin-left:30%;margin-
19. right:30%;>".$_quest."</h4> ";
20. $reason = $_POST['sol'];
21. if (isset($_POST['Submit1'])) {
22.     $selected_radio = $_POST['answer'];
23.     if ($selected_radio == $_SESSION['information'][1]) {
24.         $answerStatus = "correct";
25.
26.         $_SESSION['userScoreData'][2]=$_SESSION['userScoreData'][2]+1;
27.         $display = "✔";
28.         array_push($_SESSION['userScoreData'][9],1);
29.         $score = 1;
30.     }
31.     else {
32.         $answerStatus = "wrong";
33.         $_SESSION['userScoreData'][3]=$_SESSION['userScoreData'][3]+1;
34.         $display = "✘";
35.         array_push($_SESSION['userScoreData'][9],0);
36.         $score = 0;
37.     }
38. }

```

**Algorithm Design -**  
Increment question by 1  
and display the question

**Algorithm  
Design – If**  
the answer is  
correct

**Algorithm  
Design – If**  
the answer is  
wrong



```

36.     }
37.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".Your response was ".$selected_radio."</h4> ";
38.     echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".This was : ".$answerStatus."</h4> ";

39.
40.     require "connection.php";
41.     $conn = connection::getConn();
42.     $num = $_SESSION['userScoreData'][4];
43.     $sql = "INSERT INTO `question` (`questions`, `topicID`) VALUES ('".$_SESSION['information'][0]."', $num)";
44.     if ($conn->query($sql) == TRUE) {
45.     } else {
46.         echo "Connection error";
47.     }
48.     $question = $_SESSION['information'][0];
49.
50.     $sql = "SELECT questionsID FROM `question` WHERE questions = '$question'";
51.     $result = $conn->query($sql);
52.     if ($conn->query($sql) == TRUE) {
53.         if ($result->num_rows > 0) {
54.             while($row = $result->fetch_assoc()) {
55.                 $qid = $row["questionsID"];
56.             }}else {
57.                 echo "Connection error";
58.             }
59.             $sid = $_SESSION['userData'][2];
60. $sql2 = "INSERT INTO `scoreBook` (`studentID`, `questionID`, `score`, `Reason`)
        VALUES ('$sid', '$qid', '$score', '$reason')";
61.         if ($conn->query($sql2) ==TRUE) {
62.         } else {
63.             echo "Connection error";
64.         }
65.     ?>
66. <html><center><span><?php echo $display?></span></center></html>
67. <?php
68.     if ($answerStatus=="wrong"){
69.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".The correct answer was : ".$_SESSION['information'][1]."</h4> ";
70.         echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".The correct method is : ".$_SESSION['information'][2]."</h4> ";
71.     }
72.     require "backToSort.php";
73.     ?>

```

**Algorithm Design – Output if correct or not.**

**Algorithm Design – Add question and topicID into question table**

**Algorithm Design - Insert studentID, questionID, score and reason into the scorebook table**

sql.php

**Technique - Class**

```

1. <?php //oop - class
2. class sql {
3.     private $topicID;
4.     private $type;
5.     private $question;
6.     private $questionID;
7.     private $level;
8.     function getTopicID() {
9.         return $this->topicID;
10.    }
11.    function getType() {
12.        return $this->type;
13.    }
14.    function getQuestion(){
15.        return $this->question;
16.    }

```

17

18



```
turn $this->questionID;
```

**Algorithm Design** - if wrong then display correct answer and method

## Objective 22

Question generated for each question should be random

```
19.     }
20.     function getLevel() {
21.         return $this->level;
22.     }
23.     //function to get question from db
24.     function getQuestionFromDB($topicID,$type,$level=2){
25.         $conn = connection::getConn();
26.         $sql = "SELECT * FROM `questionBank` WHERE topicID='$topicID' AND type='$type' AND level='$level'";
27.         $result = $conn->query($sql);
28.         if ($result->num_rows > 0) {
29.             while($row = $result->fetch_assoc()) {
30.                 $this->question = $row["question"];
31.                 $this->questionID = $row["questionID"];
32.             }
33.         }
34.         return $this->question;
35.     }
36.     //function to get the ID of the question
37.     function getQuestionIDFromDB($question){
38.         $conn = connection::getConn();
39.         $sql = "SELECT * FROM `questionBank` WHERE question='$this->question'";
40.         $result = $conn->query($sql);
41.         if ($result->num_rows > 0) {
42.             while($row = $result->fetch_assoc()) {
43.                 $this->questionID = $row["questionID"];
44.             }
45.         }
46.         return $this->questionID;
47.     }
48. }
49.
50. }
51. ?>
```

Technique - Composition

SQL – Get question from database

Coding Style – Modules with appropriate interfaces

Data Structure SQL table

string.php

Technique - Class

```
1. <?php //oop - class
2. class stringQuestion {
3.     private $replace;
4.     private $newVal;
5.     private $location;
6.     private $question;
7.     //function to replace in string
8.     function getString($replace,$newVal,$location) {
9.         $this->question = str_replace("$replace", "$newVal", $location);
10.        return $this->question;
11.    }
12.    function getQuestion(){
13.        return $this->question;
14.    }
15. }
16. ?>
```

Coding Style – Cohesive modules

String Replace Function

Coding Style - Modules

sub.php

```
1. <?php //substitution topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(17,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    $general = "y=n1x+n2";
16.    //generates numbers based on user knowledge
17.    if ($_SESSION['userScoreData'][8]==1){
18.        $number1 = $num->getRandomNumber(2,10);
19.        $number2=$num->getRandomNumber(2,10);
20.    }elseif ($_SESSION['userScoreData'][8]==2){
21.        $number1 = $num->getRandomNumber(2,20);
22.        $number2=$num->getRandomNumber(2,20);
23.    }elseif ($_SESSION['userScoreData'][8]==3){
24.        $number1 = $num->getRandomNumber(10,30);
25.        $number2=$num->getRandomNumber(10,30);
26.    }
27.    $updated1 = $string->getString("n1",$number1,$general);
28.    $updated2 = $string ->getString("n2",$number2,$updated1);
29.    $sub = rand(2,20);
30.    if ($type == "m"){
31.        $ask = "When x = ".$sub.", what is the value of y?";
32.        $question=$updated2." ".$ask;
33.        $answer = $num->addition($num->multiply($number1,$sub),$number2);
34.        $optOne = $num->subtraction($num->multiply($number1,$sub),$number2);
35.        $optTwo = $num->addition($num->addition($num->multiply($number1,$sub),$number2),$number2);
36.        echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$question."</h4> ";
37.        $sol="((".$number1." X ".$sub.") + ".$number2.)";
38.        $_SESSION['information']=array($question,$answer,$sol,$optOne,$optTwo);
39.        require "mcqOptionOrder.php";
40.        require "mcqDisplay.php";
41.    }elseif ($type == "s"){
42.        $ask = "When x = ".$sub.", what is the value of y?";
43.        $question=$updated2." ".$ask;
44.        $answer = $num->addition($num->multiply($number1,$sub),$number2);
45.        echo "<h4 align=center style='margin-left:30%;margin-right:30%;>".$question."</h4> ";
46.        $sol="((".$number1." X ".$sub.") + ".$number2.)";
47.        $_SESSION['information']=array($question,$answer,$sol);
48.        require "generalShortNumberForm.php";
49.    }elseif($type == "l"){
50.        $suby = $num->getRandomNumber(2,20);
51.        if (floor(($suby-$number2)/$number1)!=ceil(($suby-$number2)/$number1)){ //if not integer, regenerates until finds a y value that is integer
52.            while (floor(($suby-$number2)/$number1)!=ceil(($suby-$number2)/$number1)){
53.                $suby = $num->getRandomNumber(2,20);
54.            }
55.        }
56.        $ask = "A. When x = ".$sub.", what is the value of y? B. When y = ".$suby." what is the value of x";
```

```

57.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$question.\" </h4> ";
58.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$updated2.\" </h4> ";
59.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$ask.\" </h4> ";
60.         $updated3=$question.\" ".$updated2.\" ".$ask;
61.         $answerOne = $num->addition($num->multiply($number1,$sub),$number2);
62.         $solA="((".$number1.\" X ".$sub.\" ) + ".$number2.\" )";
63.         $answerTwo = $num->division($num->subtraction($suby,$number2),$number1);
64.         $solB="((".$suby.\" - ".$number2.\" ) / ".$number1.\" )";
65.         $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo,$solB)
;
66.         require "generalNumberLongForm.php";
67.     }
68. ?>

```

*subtraction.php*

```

1. <?php //subtraction topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(2,$type,$_SESSION['userScoreData'][8]);
14.    $id= $s->id($question);
15.    if ($type == "m"){
16.        //generates numbers based on user knowledge
17.        if ($_SESSION['userScoreData'][8]==1){
18.            $number1 = $num->getRandomNumber(2,10);
19.            $number2=$num->getRandomNumber(2,10);
20.        }elseif ($_SESSION['userScoreData'][8]==2){
21.            $number1 = $num->getRandomNumber(2,100);
22.            $number2=$num->getRandomNumber(2,100);
23.        }elseif ($_SESSION['userScoreData'][8]==3){
24.            $number1 = $num->getRandomNumber(10,300);
25.            $number2=$num->getRandomNumber(10,300);
26.        }
27.
28.        if ($number1 < $number2){ //makes sure answer is not negative
29.            $temp = $number2;
30.            $number2 = $number1;
31.            $number1 = $temp;
32.        }
33.        $updated1 = $string->getString("n1",$number1,$question);
34.        $updated2 = $string->getString("n2",$number2,$updated1);
35.        echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>\".$updated2.\" </h4> ";
36.        $answer = $num->subtraction($number1,$number2);
37.        $sol=$number1.\" - ".$number2;
38.        $optOne = $num->optionOne($answer);
39.        $optTwo = $num->optionTwo($answer);
40.        $_SESSION['information']=array($updated2,$answer,$sol,$optOne,$optTwo);
41.        require "mcqOptionOrder.php";
42.        require "mcqDisplay.php";
43.    }elseif ($type == "s"){
44.        if ($id==36 or $id==158){
45.            //generates numbers based on user knowledge
46.            do{

```

Technique – Optimisation of numbers

```

46.         if ($_SESSION['userScoreData'][8]==1){
47.             $number1 = $num->getRandomNumber(2,10);
48.             $number2=$num->getRandomNumber(2,10);
49.         }elseif ($_SESSION['userScoreData'][8]==2){
50.             $number1 = $num->getRandomNumber(2,100);
51.             $number2=$num->getRandomNumber(2,100);
52.         }elseif ($_SESSION['userScoreData'][8]==3){
53.             $number1 = $num->getRandomNumber(10,300);
54.             $number2=$num->getRandomNumber(10,300);
55.         }
56.     }while($number2>$number1);
57.     $answer = $num->subtraction($number1,$number2);
58.     $sol = "$number1"." - ".$number2";
59. }
60. if ($id==37){
61.     do{
62.         if ($_SESSION['userScoreData'][8]==1){
63.             $number1 = $num->getRandomNumber(2,10);
64.             $number2=$num->getRandomNumber(2,10);
65.         }elseif ($_SESSION['userScoreData'][8]==2){
66.             $number1 = $num->getRandomNumber(2,100);
67.             $number2=$num->getRandomNumber(2,100);
68.         }elseif ($_SESSION['userScoreData'][8]==3){
69.             $number1 = $num->getRandomNumber(10,300);
70.             $number2=$num->getRandomNumber(10,300);
71.         }
72.     }while($number2<$number1);
73.     $answer = $num->subtraction($number2,$number1);
74.     $sol = "$number2"." - ".$number1";
75. }
76. $updated1 = $string->getString("n1",$number1,$question);
77. $updated2 = $string ->getString("n2",$number2,$updated1);
78. echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated2."</h4> ";
79. $_SESSION['information']=array($updated2,$answer,$sol);
80. require "generalShortNumberForm.php";
81. }elseif($type == "1"){
82.     do{
83.         if ($_SESSION['userScoreData'][8]==1){
84.             $number1 = $num->getRandomNumber(2,10);
85.             $number2=$num->getRandomNumber(2,10);
86.             $number3 = $num->getRandomNumber(2,10);
87.             $number4=$num->getRandomNumber(2,10);
88.         }elseif ($_SESSION['userScoreData'][8]==2){
89.             $number1 = $num->getRandomNumber(2,100);
90.             $number2=$num->getRandomNumber(2,100);
91.             $number3 = $num->getRandomNumber(2,100);
92.             $number4=$num->getRandomNumber(2,100);
93.         }elseif ($_SESSION['userScoreData'][8]==3){
94.             $number1 = $num->getRandomNumber(10,300);
95.             $number2=$num->getRandomNumber(10,300);
96.             $number3 = $num->getRandomNumber(10,300);
97.             $number4=$num->getRandomNumber(10,300);
98.         }
99.     }while($number1<($number2+$number3+$number4));
100.    $updated1 = $string->getString("n1",$number1,$question);
101.    $updated2 = $string ->getString("n2",$number2,$updated1);
102.    $updated3 = $string ->getString("n3",$number3,$updated2);
103.    $updated4 = $string ->getString("n4",$number4,$updated3);
104.    echo "<h4 align=center style='margin-left:30%;margin-
right:30%;>".$updated4."</h4> ";
105.    $answerOne = $num->subtraction($num->subtraction($number1,$number2), $numbe
r3);
106.    $solA = $number1." - ".$number2." - ".$number3;
107.    $answerTwo = $num->subtraction($answerOne,$number4);
108.    $solB = $number1." - ".$number2." - ".$number3." - ".$number4;

```

```

109.     $_SESSION['information']=array($updated4,$answerOne,$solA,$answerTwo,$solB);
110.     require "generalNumberLongForm.php";
111. }
112. ?>

```

time.php

```

1. <?php //time topic question generator
2.     session_start();
3.     require "connection.php";
4.     require "assignment.php";
5.     require "number.php";
6.     require "string.php";
7.     require "titlePage.php";
8.     $conn = connection::getConn();
9.     $s = new assignment;
10.    $num = new number;
11.    $string = new stringQuestion;
12.    $type = $_GET['type'];
13.    $question=$s->question(11,$type,2);
14.    $id= $s->id($question);
15.    if ($type == "m"){
16.        $number1 = $num->getRandomNumber(1,60);
17.        $number2=$num->getRandomNumber(1,60);
18.        if ($id==22){
19.            $answer = 60*60*24;
20.        }
21.        elseif($id==23){
22.            if(floor((60/$number1)*$number2)!=ceil((60/$number1)*$number2)){ //make
s sure answer is integer
23.                while ((floor($num->multiply($num->division(60,$number1),$number2))
!=ceil($num->multiply($num->division(60,$number1),$number2))){
24.                    $number1 = $num->getRandomNumber(1,60);
25.                    $number2=$num->getRandomNumber(1,60);
26.                }
27.            }
28.            $answer=$num->multiply($num->division(60,$number1),$number2);
29.            $updated1=$string->getString("n1",$number1,$question);
30.            $question=$string ->getString("n2",$number2,$updated1);
31.        }
32.        echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>".$question."</h4> ";
33.        $sol= "(60 / ".$number1.") x ".$number2;
34.        $optOne = $num->optionOne($answer);
35.        $optTwo = $num->optionTwo($answer);
36.        $_SESSION['information']=array($question,$answer,$sol,$optOne,$optTwo);
37.        require "mcqOptionOrder.php";
38.        require "mcqDisplay.php";
39.    }elseif ($type == "s"){
40.        $hour = rand(10,24);
41.        $minute = rand(10,60);
42.        $date = '2012-09-1 '.$hour.':'.$minute.':00';
43.        $endDate = new DateTime($date);
44.        $startDate= new DateTime('2012-09-1 04:10:00');
45.        $sinceStart = $startDate->diff(new DateTime($date)); //calculates difference
in time
46.        $startPrint = $startDate->format('H:i:s');
47.        $endPrint = $endDate->format('H:i:s');
48.        $intervalStartAndEnd = $startDate->diff($endDate);
49.        $hStartAndEnd = $intervalStartAndEnd->format('%h hour(s)');
50.        $mStartAndEnd = $intervalStartAndEnd->format('%i minute(s)');
51.        $dStartAndEnd = ((int)($hStartAndEnd)*60)+(int)($mStartAndEnd);
52.        $updated1 = $string->getString("t1",$startPrint,$question);
53.        $updated2 = $string ->getString("t2",$endPrint,$updated1);

```

Coding  
Style –  
Managed  
casting of  
types

DateTime class

```

54.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". $updated2." </h4> ";
55.         $answer = $dStartAndEnd;
56.         $sol="First calculate the number of hours which is ".$hStartAndEnd." and fr
om that work out the remaining minutes which is: ".$mStartAndEnd.". Then, convert h
ours to minutes and add it up which gives the final answer";
57.         $_SESSION['information']=array($updated2,$answer,$sol);
58.         require "generalShortNumberForm.php";
59.     }elseif($type == "1"){
60.         $time1 = mt_rand(0,9).":".str_pad(mt_rand(0,59), 2, "0", STR_PAD_LEFT);
61.         $time2 = mt_rand(10,15).":".str_pad(mt_rand(0,59), 2, "0", STR_PAD_LEFT);
62.         $time3 =mt_rand(16,23).":".str_pad(mt_rand(0,59), 2, "0", STR_PAD_LEFT);
63.         $answerOne = (strtotime($time2)-strtotime($time1))/60;
64.         $answerTwo = (strtotime($time3)-strtotime($time2))/60;
65.         $solA = "First calculate number of hours, which is ".(floor($answerOne / 60
))." then calculate the minutes remaining which is ".($answerOne -
((floor($answerOne / 60))*60));
66.         $solB = "First calculate number of hours, which is ".(floor($answerTwo / 60
))." then calculate the minutes remaining which is ".($answerTwo -
((floor($answerTwo / 60))*60));
67.         $updated1 = $string->getString("t1",$time1,$question);
68.         $updated2 = $string ->getString("t2",$time2,$updated1);
69.         $updated3 = $string ->getString("t3",$time3,$updated2);
70.         echo "<h4 align=center style='margin-left:30%;margin-
right:30%;'>". $updated3." </h4> ";
71.         $_SESSION['information']=array($updated3,$answerOne,$solA,$answerTwo,$solB)
;
72.         require "generalNumberLongForm.php";
73.     }
74. ?>

```

titlePage.php

```

1. <html lang="en">
2.     <head> <!--display topic details and question details on top of each page -->
3.     <title><?php echo $_SESSION['userScoreData'][5];?></title>
4.     <meta charset="UTF-8">
5.     <meta name="viewport" content="width=device-width, initial-scale=1">
6.     <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
7.     <link href="https://fonts.googleapis.com/css?family=Raleway" rel="styleshee
t" type="text/css">
8.     <style>
9.         /* Header/Logo Title */
10.        .header {
11.            padding: 1px;
12.            color: black;
13.            font-style: italic;
14.            font-size: 10px;
15.        }
16.        body {font-family: "Raleway", Arial, sans-serif}
17.        .w3-row img {margin-bottom: -8px}
18.    </style>
19. </head>
20. <body>
21.     <header>
22.         <h4 style="text-
align:center;"><b><?php echo "TOPIC: ".$_SESSION['userScoreData'][5];?> </b><b> <?
php $_SESSION['userScoreData'][6]=$_SESSION['userScoreData'][6]+1; echo " QUESTI ON
NUMBER : ".$_SESSION['userScoreData'][6]?></b></h4>
23.     </header>
24. </body>
25. </html>

```