**WAD Trial Lab Test 1 (1.5 hours) - Set A [20 marks]**

**General Instructions:**

* You can refer to any offline resources already on your laptop, but you must disable all networking and Bluetooth connections during the test. You must not communicate with anyone via any means during the test.
* Just before the test, you will be given instructions by the invigilator as to how to obtain resource files required for the lab test and how to submit your solutions.
* No questions will be entertained during the test. If necessary, make your own assumptions.
* You are allowed to use only standard PHP classes and functions in your solutions – do not use any third party libraries.
* Use meaningful names for classes, methods, functions and variables, as well as indent your code correctly*.* Use 4 spaces for indentation. Otherwise, you may attract penalty of up to **20%** of your score for the corresponding question.
* You **MUST** include your name as author in the comments of all your submitted source files. Failure to do so WILL attract a penalty of up to **20%** of your score for the corresponding question.  
    
  For example, if your registered name is "KIM Jong Un" and email ID is kim.jongun.2019, include the following comment at the beginning of each source file you write.

<!--

Name: KIM Jong Un

Email: kim.jongun.2019

-->

* You may wish to comment out the parts in your code which cause errors. But commented code will not be marked.
* **Resources:** Click [here](https://smu.sg/2019-is113-triallt1-setA-resources)
* **Solutions:** Click [here](https://smu.sg/2019-is113-triallt1-setA-solutions)

DO NOT TURN OVER UNTIL YOU ARE TOLD TO DO SO

**Question 1 (Difficulty Level: \*/\*) [ 8 marks ]**

**Given:**

* q1-A.html
* q1-A.php
* q1-B.html
* q1-B.php

**Part A (5 marks) - Difficulty Level (\*)**

1. Update **q1-A.html** to include a form. The page displays as follows:

|  |
| --- |
|  |

**Note**: "2", "5", "6" and "2" (divisor) are values entered by the user.

Upon clicking the SUBMIT button, it submits to **q1-A.php** via HTTP POST.

1. Implement a function called **is\_divisible\_by($num, $n)**in **q1-A.php**. The function returns true if $num is divisible by $n, false otherwise.
2. Use the function **is\_divisible\_by($num, $n)** to produce the following result when the user submits the values shown above through **q1-A.html**:

|  |
| --- |
|  |

**HINT:** Explore the use of HTML unordered list **<ul>**

**Part B (3 marks) - Difficulty Level (\*)**

1. Update **q1-B.html** to include a form. The page displays as shown below. This time, one or more numbers are input together in **ONE form input field** where the numbers are separated by a **comma**:

|  |
| --- |
|  |

**Note**: "2,4,6,8,9" and "2" are values entered by the user.

Upon clicking the SUBMIT button, it submits to **q1-B.php** via HTTP GET.

1. **q1-B.php** uses the same function **is\_divisible\_by($num, $n)** that you implemented in **Part A**. Copy and paste the function from **q1-A.php** to **q1-B.php**.
2. Use the function **is\_divisible\_by($num, $n)** to produce the following result when the user submits the values shown above through **q1-B.html** :

|  |
| --- |
|  |

**HINT**: Explore the use of PHP **explode** function

**Question 2 (Difficulty Level: \*/\*\*) [ 6 marks ]**

**Given:**

* q2.php
* q2-display.php
* q2-one.php
* (Image Files) apple.jpg, orange.jpg, pear.jpg

**Part A (3 marks) - Difficulty Level (\*)**

1. Modify **q2.php** such that if the user clicks on the words ("Apple", "Orange" or "Pear"), the corresponding checkboxes will be selected/unselected.
2. Once the user selects the fruits and press on the SUBMIT button on page **q2.php**, the form submits to **q2-display.php**.
3. Modify **q2-display.php** such that appropriate images will be displayed in a tabular format (use border='1' for your table).
   * For example, if the user selects “Apple” in **q2.php**, then page **q2-display.php** must display the image file apple.jpg.

**Sample Output**

|  |  |
| --- | --- |
| **Action Taken**  **(q2.php)** | **Sample Output**  **(q2-display.php)** |
| **None selected** | **(Note:** Use HTML Heading-1**)** |
| **"Apple" selected** |  |
| **"Apple" and "Orange" selected** |  |
| **"Apple", "Orange"and "Pear" selected** |  |

**Part B (3 marks) - Difficulty Level (\*\*)**

1. In page **q2-one.php**, once the user selects the fruits and press on the SUBMIT button on page, the form submits back to itself (to **q2-one.php**).
2. Modify **q2-one.php** such that:
   * Appropriate images will be displayed in a tabular format (use border='1' for your table).
   * Appropriate message will be displayed at the top of the page.

**Sample Output**

|  |  |
| --- | --- |
| **Action Taken (q2-one.php)** | **Sample Output (q2-one.php)** |
| **None selected** |  |
| **"Apple" selected** |  |
| **"Apple" and "Orange" selected** |  |
| **"Apple", "Orange"and "Pear" selected** |  |

**Question 3 (Difficulty Level: \*\*/\*\*\*) [ 6 marks ]**

**Given:**

* q3.php

1. Page **q3.php** displays students’ course **timetable** in a **tabular format**.
2. **$students** is an (**indexed) array** of **associative arrays** where:
   * Each **associative array** represents a **student**. A student has a **name** and **one or more courses**.
   * Thus, each **associative array** contains TWO (2) key-value pairs where:

|  |  |
| --- | --- |
| **Key** | **Value Type** |
| name | String  (e.g. "Jong Un Kim") |
| courses | (Indexed) **array** of (indexed) arrays  (e.g. see below) |

* + A **course** is an (indexed) **array** containing FIVE (5) items.

|  |
| --- |
| ['CS102', 'Discrete Mathematics', 'TUE', '0830', 2] |

* + 1. 1st item is the **Course Code**
    2. 2nd item is the **Course Title**
    3. 3rd item is the **Day** on which the course is offered (e.g. MON, TUE, etc.)
    4. 4th item is the **Time** at which the course starts
       - 0830 means class starts at 08:30am
       - 1330 means class start at 1:30pm
    5. 5th item is the **Credit Units**
       - The value **1** refers to **1.5-hour** class, and
       - The value **2** refers to **3-hour** class
    6. Hence, the above course **CS102**:
       - Is offered on **TUE** and it is a **3-hour** class
       - Starts at **08:30am**
       - Ends at **11:30am**

**Part A (2 marks) - Difficulty (\*\*)**

1. Complete **q3.php** such that the name drop-down menu will display the distinct names of students.
2. **DO NOT HARD-CODE the names**.
3. Your implementation should make use of the **getStudentNames()** function given in the **q3.php** file. This function is left incomplete. Please complete this function.

When the page loads:

|  |
| --- |
|  |

When the user clicks on the drop-down menu:

|  |
| --- |
|  |

**Part B (4 marks) - Difficulty (\*\*\*)**

1. Complete **q3.php** such that it displays the selected student’s timetable showing all courses taken by the student.
2. When a stu**w Timetable**” Sdent’s name is selected in the drop-down menu and the “**Sho**UBMIT button is clicked, the form submits to itself (to **q3.php** page). It then:
   1. Displays the student’s timetable in an HTML table; and
   2. The student’s name should be "**selected**" in the drop-down menu.
3. When the page loads for the first time, the timetable displayed will be for the first student in the drop-down menu (follow the order of student names - as returned by the **getStudentNames()** function).
4. You can assume that **$students** always contains at least one student.
5. Marks will be deducted for overly repetitive code that could have been simplified, e.g., by using a loop, or by writing a function.

**HINT:** Explore the use of **align='center'** attribute of <td>

**Example 1: The page loads for the first time**

|  |
| --- |
|  |

**Example 2: After 'Donald Trump' is selected and 'Show Timetable' is clicked**

|  |
| --- |
|  |

**Example 3: After 'Hugo Chavez' is selected and 'Show Timetable' is clicked**

|  |
| --- |
|  |

**- END -**