United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Final Exam Year: 2025 Trimester: Fall

CSE 4165/CSE 465 Web Programming

Total Marks: 30 Duration: 2 Hours

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

1. Write a PHP program to design a web page that processes a student's CT and exam marks. The page should take input from an HTML form for three CT marks, a midterm mark, and a final exam mark. The program must select the best two CT marks out of the three, calculate their average, and then compute the total marks by adding the CT average with the midterm and final exam marks. Finally, display the status as 'Passed' if the total marks > 54 otherwise display 'Failed'. For better understanding, refer to the demo input and output. **[10]**

Sample Input (html)	Sample Output (php)
CT 1: 12 CT 2: 15 CT 3: 14 Midterm Marks: 22 Final Marks: 32 Calculate Total	Best two CT's total: 29 Midterm marks: 22 Final marks: 32 **Total Marks = 83** Status: Passed

2. Write a PHP program for a simple event-planning system. The program should take three inputs from the user: **number of items sold per day**, **number of days the sale runs**, and **target number of items to be sold**. First, calculate the **total items sold** by multiplying the items sold per day with the number of days. Classify the sales performance using the following conditions: **[10]**

- **Excellent:** total items sold 500 or more
- **Good:** total items sold 300 or more but less than 500
- **Average:** total items sold 150 or more but less than 300
- **Poor:** total items sold less than 150

The program should then compare the total items sold with the target number of items. Finally, display the total items sold, the performance category, the target value, and clearly state whether the sales are **above**, **below**, or **exactly at** the target, along with the difference.

Sample Input (taken using html form)			Sample Output (printed using PHP code)		
Items Sold	Number of	Target	Total Items	Performanc	Result
25	12	300	300	Good	Target met exactly (0
40	10	350	400	Excellent	Above target by 50
15	10	200	150	Average	Below target by 50
8	10	120	80	Poor	Below target by 40

3. Create a database named **campus_library** and a table named **book_loans**. The table tracks student borrowing habits and book conditions. **[10]**

LoanID	StudentName	BookTitle	DaysOverdue	PenaltyFee	Status
101	Abdul	Data Structures	0	0.00	Returned
102	Jabbar	Operating Systems	12	24.00	Overdue
103	Barkat	Discrete Math	5	10.00	Overdue
104	Rahim	Linear Algebra	2	4.00	Overdue
105	Karim	Data Structures	15	30.00	Lost
106	Fahim	Operating Systems	0	0.00	Returned

Answer the following queries (using php and mysql)

1. Show the total number of books for each Status (*Returned*, *Overdue*, *Lost*), but only include statuses that have more than 1 entry in the table.
2. For any student who has a Status of '*Overdue*' and DaysOverdue less than 7, change their Status to '*Grace Period*' and set their PenaltyFee to 0.
3. For students with a PenaltyFee greater than 20.00, increase their fee by 10% as a "*processing charge*," but only if the final fee does not exceed 50.00.
4. Display each BookTitle and the total PenaltyFee collected/owed for that book. Sort the results so that the book generating the most money appears at the top.