

Lab 4 – Simple CRUD Application with Login				
Section	Task	Comments	Marks	Out Of
Best Practices				
	All database queries are completed using prepared statements.			3
	All form inputs are sanitised and validated.			3
	All information is easy-to-read, responsive, and user-friendly.			2
Database Requirements				
	The table with all of the information for the attractions is named using the student's username, followed by 'attractions'. Example: 'username_attractions'.			1
	This table contains columns for a primary key, the name of the attraction, category, and description. Each column has an appropriate data type and length.			2
	This table also contains six additional columns that capture different aspects of each attraction. Each column has an appropriate data type and length.			2
Login Requirements				
	A login page exists, including a login form that queries the database for a provided username and checks the provided password against a stored password hash. Please add a user to your login script: "admin", "Password1!" .			2
	When the user logs in, a \$_SESSION is started that stores the username. Additionally, the session ID is automatically regenerated to prevent session fixation attacks.			2
	If the user is not logged in, they will be unable to access any of the administrative, add, edit, or delete pages. If they try to access these pages, they will either be redirected to the login page or met with an error message.			2
	A user who is logged in may access the administrative, add, edit, and delete pages.			1
	Users who are logged in may also log out. When logging out, the \$_SESSION and all related variables are destroyed.			2
Home Page (Index)				
	The home page lists all of the records and their associated information.			2
	Home page does NOT require the user to login.			1
	A list of links are provided for various attraction categories.			2
Additional Challenge	When the user clicks a category link, they are able to see all of the records for that specific category.			3
Add Page				
	The add page includes a form with fields for each of the columns in the attractions table.			2
	Each field is appropriate for the data type of each column.			3
	Upon successful submission of the form, the new record is inserted into the database and the user is given a message.			2
Edit Page				
	The user is provided a list of all existing records in the database, including a button to edit each specific entry.			2
	Upon clicking the edit link, the user is provided with a form, similar to the add page. This form is prepopulated with all of the existing values in the database.			3
	The user is able to change the values in the form and, upon successful validation, the record is updated and they are provided with a message.			2
Delete & Confirmation Pages				
	The user is provided a list of all existing records in the database, including a button to delete each specific entry.			2
	Before the deletion occurs, the user is prompted for confirmation.			2
	Upon successful deletion of the record, the user is given a message.			2
Deductions				
	Potential deductions may include not using semantic HTML in your output, not giving each document a unique title, directory structure and file naming errors, required files such as images and stylesheets missing from the server, variable naming conventions, program errors that are printed out to the user, or fatal errors that prevent the rest of the program from running.			-10
		Total	0	50