WEB DEVELOPMENT

ASSESSMENT 1: Set Exercises

Module Tutor: Lavanya Mohan

|  |  |
| --- | --- |
| Contribution towards overall module mark | 40% |
| Date set |  |
| Deadline |  |
| Marked work returned by |  |
|  |  |

# Assessment 1: Set Exercises

## The Brief

Your task is to develop solutions to a series of exercises that will test your knowledge of the fundamental web development techniques introduced through the course of the module. These exercises complement the content delivered in class each week and allow you to put into practice what you have learned. The exercises can be found in each chapter of the Web Development GitHub repository. The required exercises can be found under the heading Assessment Exercises. Each chapter contains extension problems or bonus exercises that can be completed for additional marks.

You should compile an organised repository of code over the course of the module. It is recommended that you create a new project for each exercise and save this to the corresponding exercises folder provided in each chapter. Each exercise should be appropriately named (e.g. *01-FixMe*) so they are easy to find. You should commit changes in your repository often (e.g. after completing each exercise), use descriptive messages for these commits, and ensure you are regularly pushing your code back to GitHub. Throughout the module there are staged deadlines where exercises from specific chapters must be completed by:

|  |  |  |
| --- | --- | --- |
| Deadline | Required Chapters | Deadline Date |
| 1 | Chapter 1: Introduction to HTML & CSS  Chapter 2: Links & Navigation  Chapter 3: Images |  |
| 2 | Chapter 4: Page Layout  Chapter 5: Embedded Content  Chapter 6: Javascript |  |

## Deliverables

The deliverables for this assignment are as follows:

● Your code in response to the set exercises. Code must be pushed to your GitHub repository before the corresponding deadlines listed above.

## Submission

For each submission you should ensure your code for the required chapter exercises has been submitted to your GitHub repository. Please adhere to the following method:

* Check your exercises are functioning as expected.
* Commit and push your code to your GitHub repository.

Only code pushed to your Github repository before the assessment deadline will be marked. Ensure you give yourself enough time before your final push. Guides on how to push your code to GitHub are provided in your repository and on Minerva.

## Marking Criteria

Each skills portfolio submission will be evaluated against the following criteria and be worth 50% of the final grade.

1. Technical Implementation (75%)
2. Design (15%)
3. Repository Presentation (10%)

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Weighting | Mark Range Description | Mark Range |

|  |  |  |  |
| --- | --- | --- | --- |
| Technical  Implementation | 75% | Limited solutions that demonstrate little evidence of the techniques introduced. Several required exercises are missing or incomplete. No comments included in the code and limited attention given to code presentation | 0 - 19 (Low Fail) |
| Solutions contain significant errors and may not function correctly. Some required exercises are missing or incomplete. Little to no commenting in the code or attention given to code presentation | 20 - 39  (Fail) |
| Basic solutions that demonstrate a limited understanding of the techniques tested. Some required exercises may be missing. Some features may not function. Commenting is limited and code presentation has scope for improvement | 40 - 49  (Third) |
| Fair solutions that show an understanding of the techniques introduced. Solutions are fully functional, although may not use the most efficient method. All required exercises are submitted. Code is commented to a fair degree. Code presentation is reasonable, although some room for improvement possible | 50 - 59  (2:2) |
| Overall good solutions that demonstrate a sound understanding of the techniques introduced. Solutions consider efficiency of code. Code is well commented and presented appropriately. All required exercises are submitted and the submission may | 60 - 69  (2:1) |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | include attempts at extension problems and bonus exercises |  |
| Overall very good code that provides efficient solutions to the coding challenges. Strong understanding of techniques is evident. Code is very well presented with detailed commenting. All required exercises are submitted and submission includes several additional exercises beyond the minimum requirements. These additional exercises may be more advanced in nature and are completed to a very good standard | 70 - 79  (First) |
|  |  | Excellent solutions that extend the exercises to demonstrate techniques outside the scope of the class. Code is presented and commented to a high standard. All required exercises are submitted as well as several additional exercises beyond the minimum requirements, These additional exercises are more advanced in nature and completed to an excellent standard | 80 - 89  (High First) |
|  |  | Beyond expectations for this level of study. | 90 - 100  (Outstanding) |
| Design | 15% | Very limited attention paid to the design. | 0 - 19 (Low Fail) |
| Poor design with little to no direction. | 20 - 39  (Fail) |
| A basic design with limited graphical elements and/or styling | 40 - 49  (Third) |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | A fair design that implements basic graphical elements and/or. Look and feel is only loosely considered. | 50 - 59  (2:2) |
|  |  | A good design that implements well considered graphical elements and/or styling. Look and feel is pleasing though there is room for refinement. | 60 - 69  (2:1) |
| A very good design. Graphical elements and/or styling have been implemented successfully to enhance visual appeal of the front end. | 70 - 79  (First) |
| A highly sophisticated design. Excellent use of graphical elements and styling techniques to enhance the visual appeal of the front end. | 80 - 89  (High First) |
|  |  | Beyond expectations for this level of study. | 90 - 100  (Outstanding) |
| Repository  Presentation | 10% | Limited repository organisation that does not adhere to the method specified. Exercises are poorly named and saved in wrong locations. Commit messages lack description and code is likely all pushed in a single submission | 0 - 19 (Low Fail) |
| Poor repository organisation with unclear exercise naming. Some exercises are saved in incorrect locations making them difficult to find. Commit messages lack description and code is infrequently pushed. | 20 - 39  (Fail) |
| Basic repository organisation. Some exercises may be difficult to find. Commit messages are basic and would benefit from further clarity.  Code is pushed on an irregular basis. | 40 - 49  (Third) |
|  |  | Fair repository organisation, though there may be some minor slips in presentation. Commit messages are good, though there is room for refinement. Code is mostly pushed on a per task basis | 50 - 59  (2:2) |
| Good repository organisation, though some exercises may need better naming conventions. Commit messages are good, though there is room for refinement. Code is pushed on a per task basis with only minor slips in attention | 60 - 69  (2:1) |
|  |  | Very good repository organisation, exercises are easy to find and clearly labelled. Commit messages are clear with code frequently pushed on a per task basis | 70 - 79  (First) |
| Excellent highly organised repository, with clear commit messages. Code is pushed very frequently. | 80 - 89  (High First) |
| Beyond expectations for this level of study. | 90 - 100  (Outstanding) |
|  |

## Intended Learning Outcomes (ILOs)

|  |  |
| --- | --- |
| ILO | Assessed |
| The application of HTML5 and CSS to create media-rich artefacts that are deployed online. |  |
| An adherence to coding conventions that ease the review, maintenance and debugging of web applications. | ✓ |
| An ability to deploy computational thinking to select and apply appropriate technical strategies for addressing a web development problem. | ✓ |
| An ability to discuss the technical implementation of a web project and reflect critically on the results. | ✓ |

Mark penalties may be applied to late submissions without prior approval of an extension. Please ensure that you prepare and submit your work in good time to allow for any issues that may arise.