|  |  |  |  |
| --- | --- | --- | --- |
| **INFO 2302: WEB TECHNOLOGIES**  **Activities for CSS: 02 GRID** | | | |
| NAME | Mohamad Imad Addin Bin Ja’far | | |
| MATRIC NO | 2127923 | SECTION | 6 |

Instruction: -

1. Please download the file CSS02.zip
2. Unpackage the content into a folder
3. Write the codes and remarks about what have changed in the web browser.

|  |  |  |
| --- | --- | --- |
| INSTRUCTIONS | CODE | REMARKS |
| Download grid.html and mygrid.css |  | You will be learning how to use GRID to do layout |
| External link | Create the link to mygrid.css in grid.html |  |
| \* Then tick the Grid display setting in the inspector view | Insert **the following** in the Parent  *display: grid;*  *grid-template-columns: 1fr 1fr 1fr 1fr;*  *gap: 20px;* | **grid-template-columns**: Divides the container into 4 equal columns (1fr for each).  **gap**: Adds 20px space between grid items, making layout look more organized and less crowded. |
|  | header{      grid-column: 1/span 4;      grid-row: 1;      }  nav{      grid-column: 1;      grid-row: 2;  }  article{      grid-column: 2/span 2;  }  aside{      grid-column: 4;  }  footer{      grid-column: span 4;      grid-row: 3;  } | header spans across all 4 columns (from column 1 to 4) and occupies the first row of the grid, stretching fully across the top.  nav is positioned in the second row and the first column. It aligns under header and takes up only the first column, pushing other content aside.  article spans across columns 2 and 3, giving it more width than nav and aligning beside it in the second row.  aside appears in the fourth column of the second row, aligning it to the far right, beside the article.  footer spans across all 4 columns and appears in the third row, positioning it at the bottom of the layout. This creates a full-width footer. |
|  |  |  |
|  |  |  |
|  | Create the link to mygrid2.css in grid.html | You will learn how to do layout using GRID |
|  | Insert **the following** in the Parent  display: grid;      gap: 20px;      grid-template-areas:      "head head head head"      "navi navi navi navi"      "content content content side"      "foot foot foot foot"      ;      grid-template-columns: 1fr 1fr 1fr 1fr; | There’s now a 20px space around each section  Each section is displayed horizontally next to each other |
|  | header{      grid-area: head;  }  article{      grid-area: content;  }  nav{      grid-area: navi;  }  aside{      grid-area: side ;  }  footer{      grid-area: foot;  } | Header now spans the full width on top  Article is now on the bottom spread horizontally  Nav is positioned below header justified to the left  No visible changes  Footer now spans across the full width at the bottom |
|  |  |  |