|  |  |  |  |
| --- | --- | --- | --- |
| Theme | Unit: Layouts | | |
| Duration | 3 week / ~15 days | | |
| Essential Question/s | 1. What are the benefits of creating a layout? | | |
| Daily  Aims | Day1: What are the key concepts in creating a layout?  Day2: How do we position elements using a normal flow and relative positioning?  Day3: How do we position elements by using absolute positioning and fixed positioning?  Day4: How do we address overlapping issues and floating elements?  Day5: Weekly summative assessment  Day6: How do we use floats and adjust floats?  Day7: How do we fix issues regarding parents of floated elements?  Day8: How do we create multi-column layouts with floats?  Day9: What is the difference between fixed width layouts and liquid layouts?  Day10: Weekly summative assessment  Day11: Fixed width layout practice  Day12: Fixed width layout practice  Day13: Liquid layout practice  Day14: Liquid layout practice  Day15: Final | | |
| Vocabulary | Domain  Specific | | * normal flow * relative positioning * absolute positioning * fixed positioning * floating elements * static * relative * overlapping * pixel * pixel border * resolution * stacking context |
| General  Academic | | * clear * adjust * offset |
| Key Student Learning Objectives  Students will be able to: | Content  Objective | | 1. Explore different ways to position elements using normal flow, relative positioning, absolute positioning and floats. 2. Discover how various devices have different screen sizes and resolutions, and how this affects the design process. 3. Learn the difference between fixed width and liquid layouts, and how they are created. 4. Find out how designers use grids to make their page designs look more professional. |
| Language  Objective | | 1. Differentiation between syntax and code. 2. Understand how to use domain specific vocabulary in various situations. |
| Sequence of Key Learning Activities | 1. Mini-lesson 2. Conceptual lecture 3. Student-centered activity 4. Code sharing | | |
| Rigor  (from ‘Checking for Rigor’) | * Students will be asked to write code in their own interpretation. * note book checks * checking for text annotations * Ask students to highlight syntax and code. | | |
| Unit flow: |  | | |
| Assessments | Formative Assessment  (Check for understanding) | * Oral Questioning * Misconception check * Exit Ticket | |
| Summative Assessment | * Weekly Assessment * Final project: | |
|  | Final project | * Final project: * 1) students will be given a wireframe layout and is expected to deconstruct all components. * 2) Once deconstructed, students will adjust the wireframe to their own interpretation and form a new layout. | |