



CSS Grid Layout

Specification overview. Implementation status and roadmap.

Sergio Villar Senin - Igalia
[BlinkOn 3](#) - 4-5 Nov 2014

About Igalia & me

- Igalia is an **Open Source consultancy** founded in 2001
- Maintainers of the WebKitGtk port
- Member of **Igalia Web Engines Team**
- **WebKit** reviewer
- **Blink** committer
- Current working on **CSS Grid Layout**



“allows authors to easily define complex,
responsive 2-dimensional layouts”

Tab Atkins Jr.
Google

Motivation

- **Grid based designs** were first done using tables and more recently floating divs
- Those approaches have issues and a lot of complexity
- Lots of CSS frameworks emerging to make things easier
- CSS Grid Layout is a **powerful and flexible standard** defined by the **W3C**
- Latest ED from 23 September 2014



Web Engines Adoption



almost complete (under experimental flag)

- <chrome://flags/#enable-experimental-web-platform-features>



almost complete (released in Safari nightlies)



very early stages (just some property parsing)



outdated, based on very old version of the specs

BTW, this is a grid!



#cssgrid #blinkon3 #igalia



LOG IN | SIGN ...

LONGFORM ▾

VIDEO ▾

REVIEWS ▾

TECH ▾

SCIENCE ▾

CULTURE ▾

DESIGN ▾

BUSINE Q

US & WORLD ▾ FORUMS ▾

This is the Fitbit Surge, a \$250 'superwatch' with built-in GPS

By Chris Welch

92

Pascal Blanche is building a stunning sci-fi universe

By Andrew Webster

18

I flew in the Red Bull Air Race and I didn't die

By Sean O'Kane

21

The Verge Playlist: Brian Eno

By Colin Lecher

21

Microsoft to launch its wearable fitness band in coming weeks

By Tom Warren

113

Tesla's autopilot isn't special (but it's still cool)

By Chris Ziegler

86

'Birdman' review: a surreal and spectacular look at a superhero coming undone

By Molly Osberg

20

mozilla

Descubra
que motiva a
nosa misión



Comprenda a
importancia da
súa privacidade
na rede



Descargar o
Firefox



Ensinar a Web



Cree un aplicativo
en segundos

Motivation

CSS Grid Layout Example

Chapters

- Down the Rabbit-Hole
- The Pool of Tears
- A Caucus-Race and a Long Tale
- The Rabbit Sends in a Little Bill
- Advice from a Caterpillar
- Pig and Pepper
- A Mad Tea-Party
- The Queen's Croquet-Ground
- The Mock Turtle's Story
- The Lobster Quadrille
- Who Stole the Tarts?
- Alice's Evidence

ALICE'S ADVENTURES IN WONDERLAND

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversations?'

So she was considering in her own mind (as well as she could, for the hot day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.

*"Begin at the beginning
and go on till you come
to the end; then stop."*

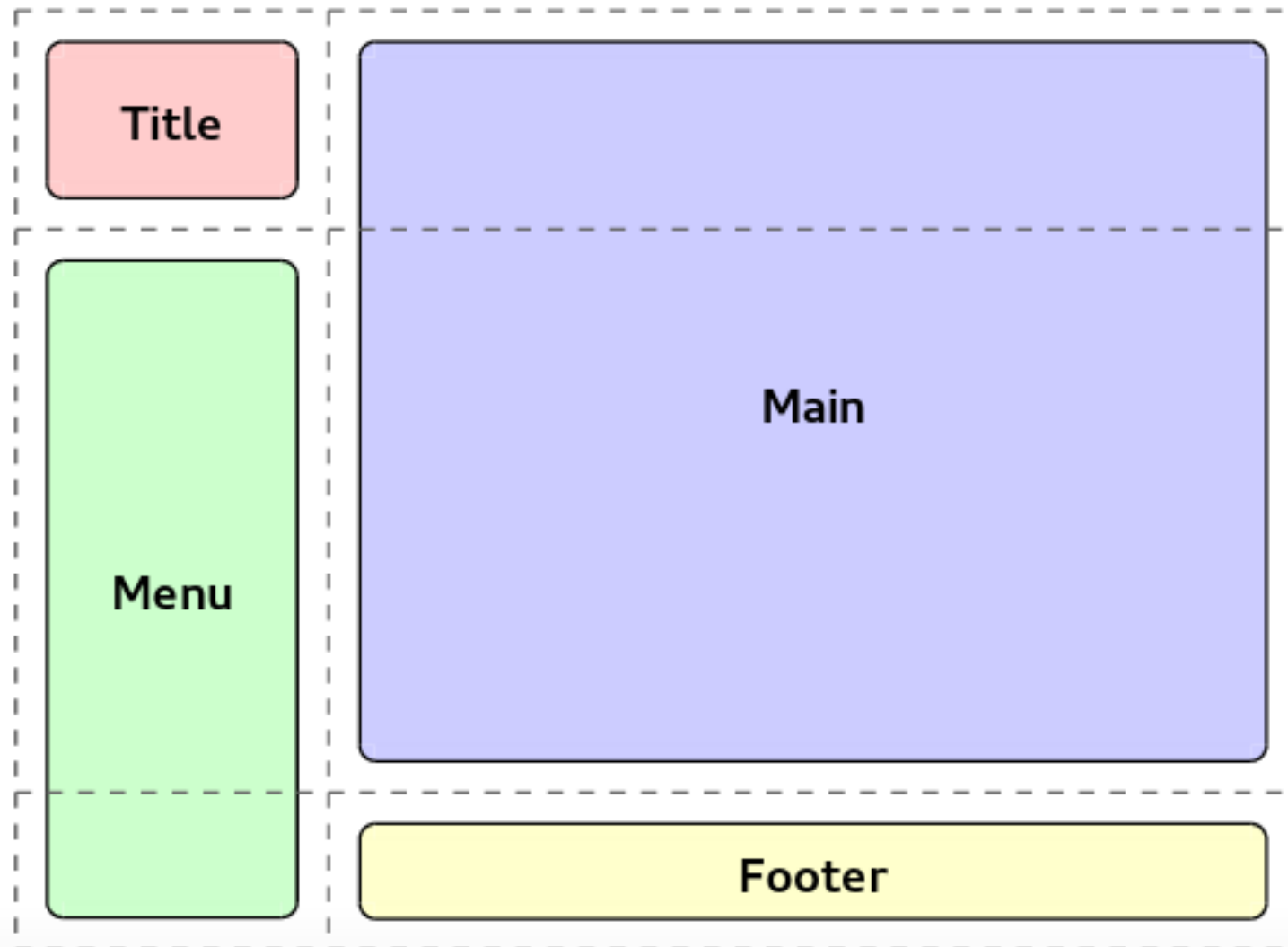
Lewis Carroll

From [Project Gutenberg](#)



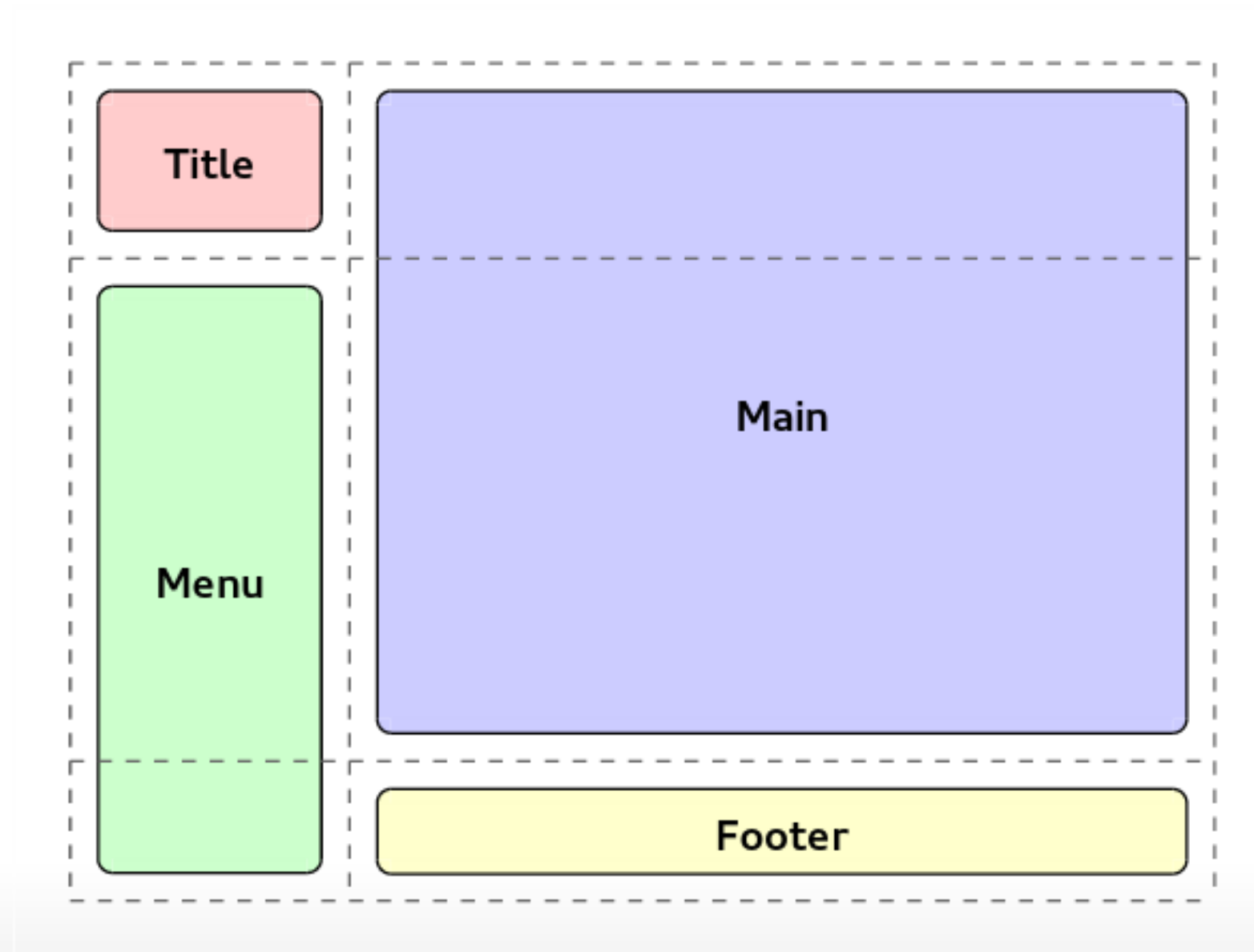
Introduction

CSS Grid Layout



Introduction

CSS Grid Layout

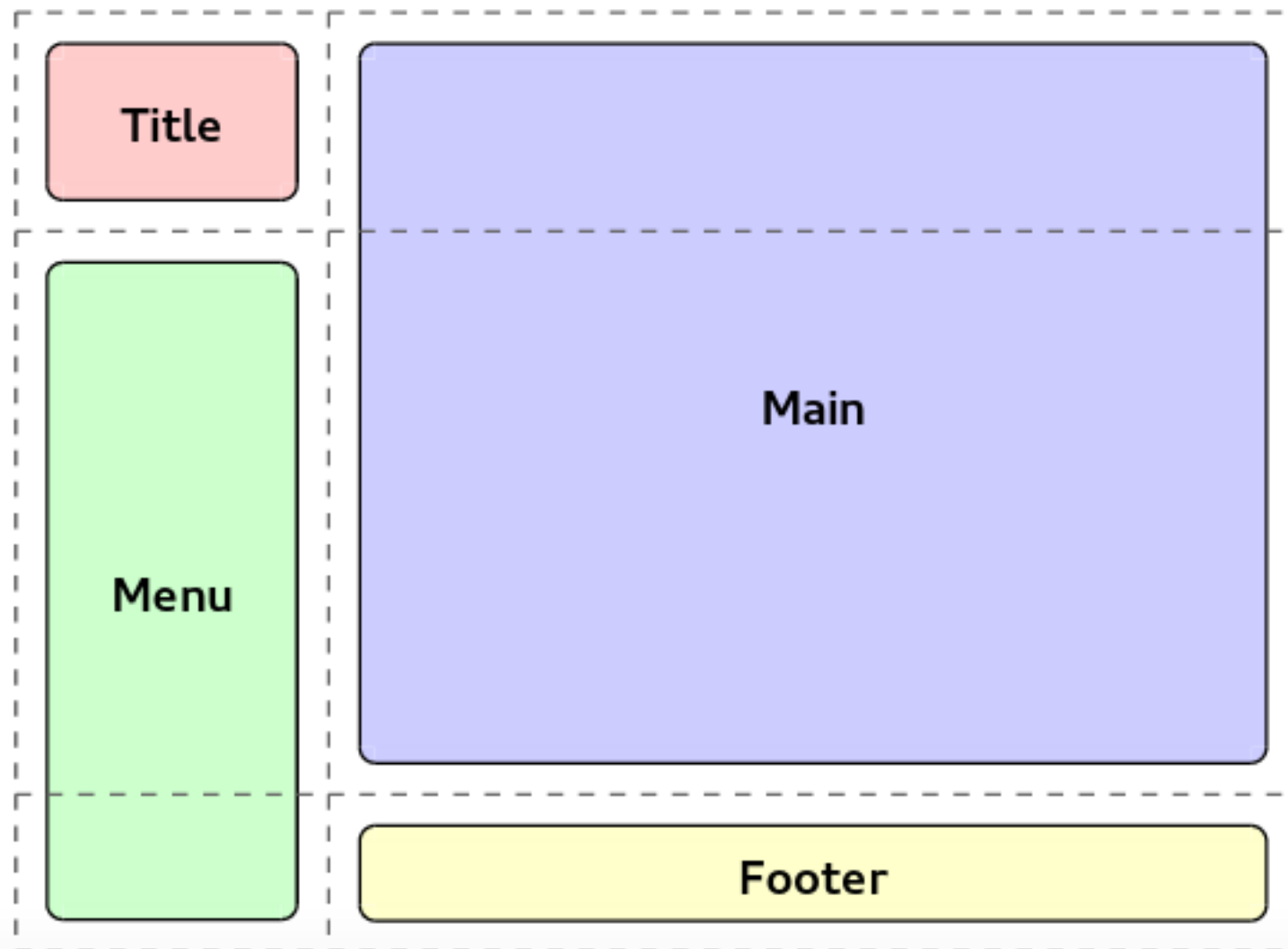


- Mechanism to divide the available space in **rows and columns**



Introduction

CSS Grid Layout

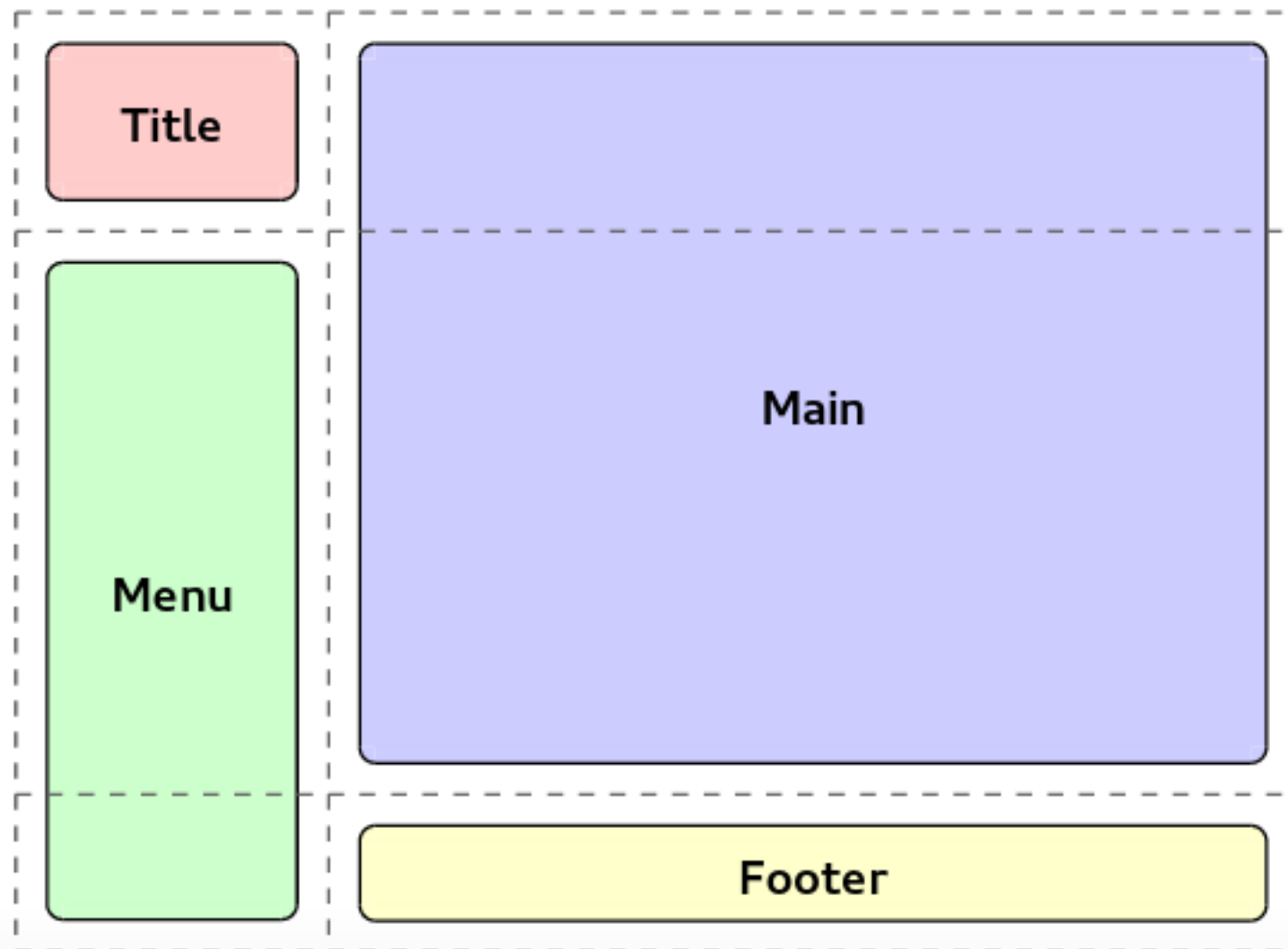


- Mechanism to divide the available space in **rows and columns**
- Predictable sizing behaviors



Introduction

CSS Grid Layout

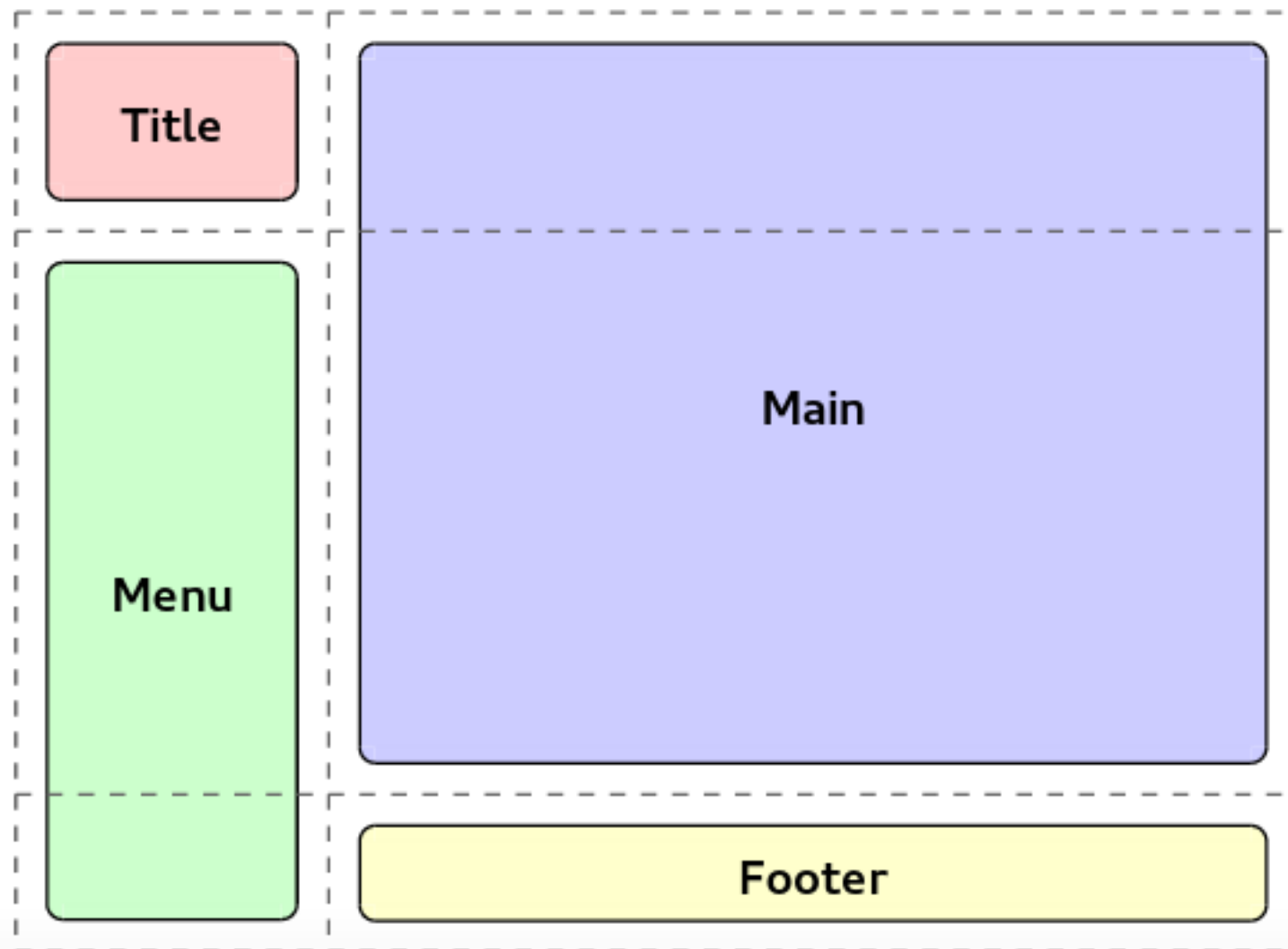


- Mechanism to divide the available space in **rows and columns**
- **Predictable sizing behaviors**
- Defines a set of **grid areas** where to precisely place elements of a web page



Introduction

CSS Grid Layout

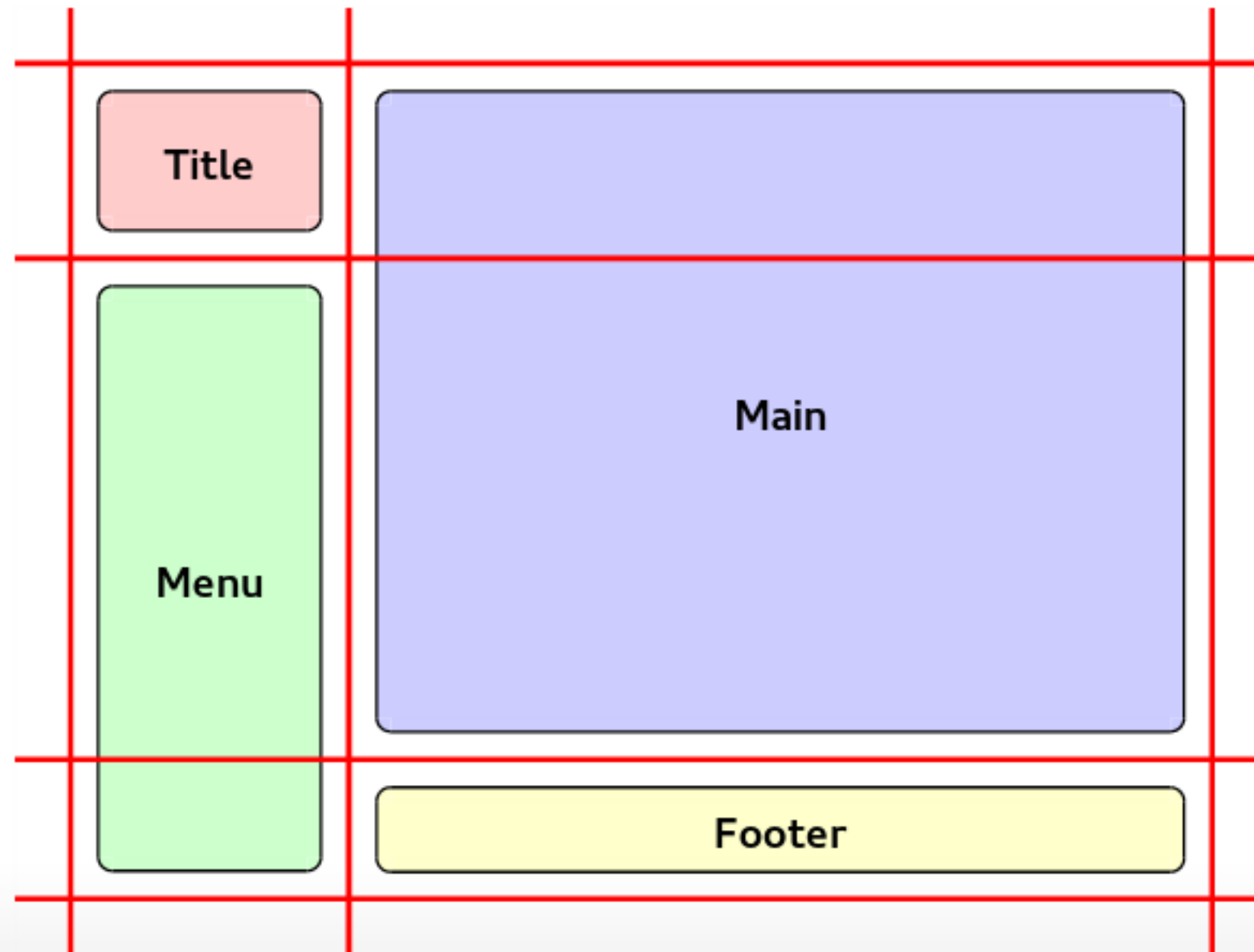


- Mechanism to divide the available space in **rows and columns**
- **Predictable** sizing behaviors
- Defines a set of **grid areas** where to precisely place elements of a web page
- Able to intelligently **reflow elements** optimizing locations and sizes



Concepts I

Grid Lines



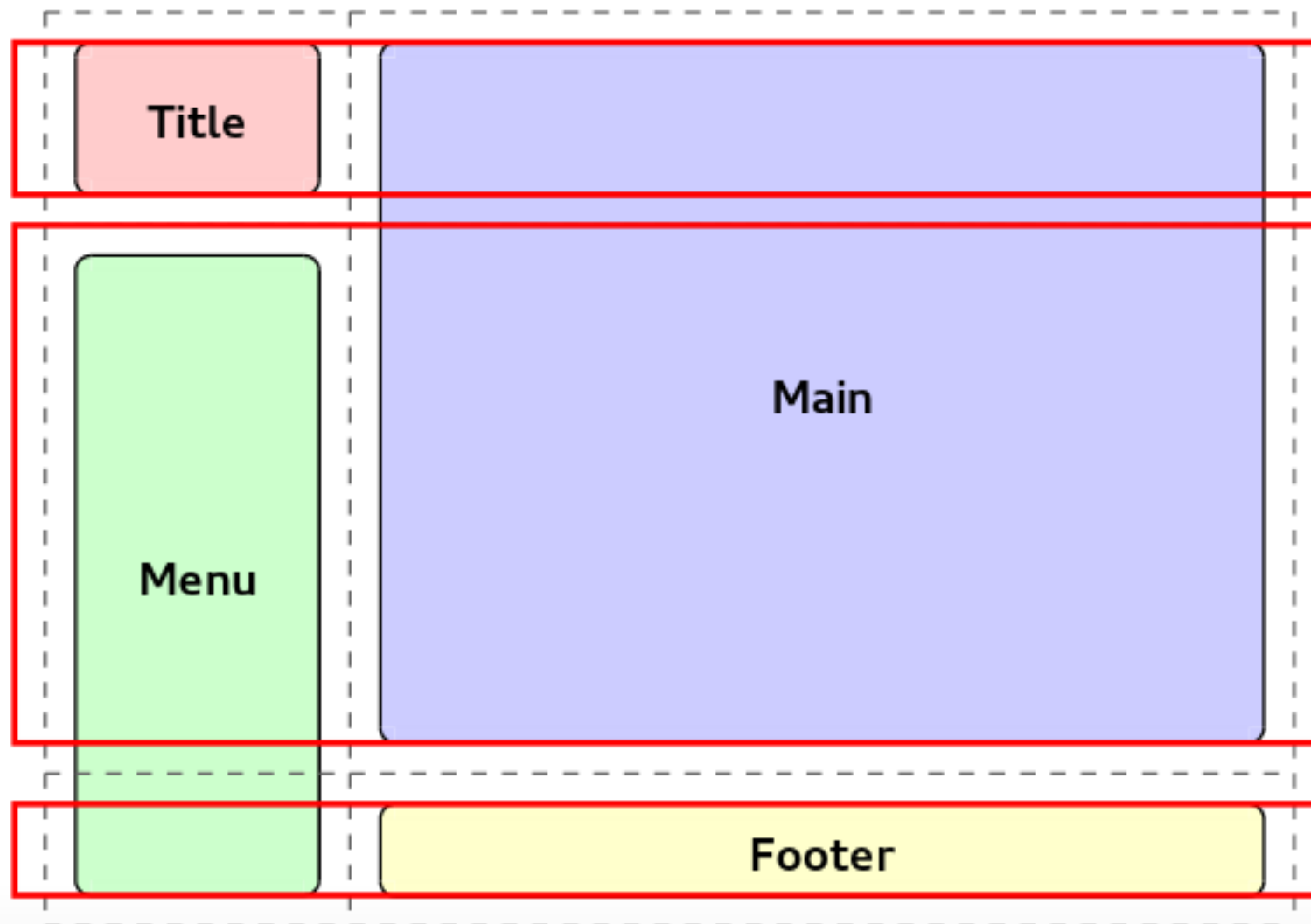
Grid lines are the horizontal and vertical dividing lines of the grid



Concepts II

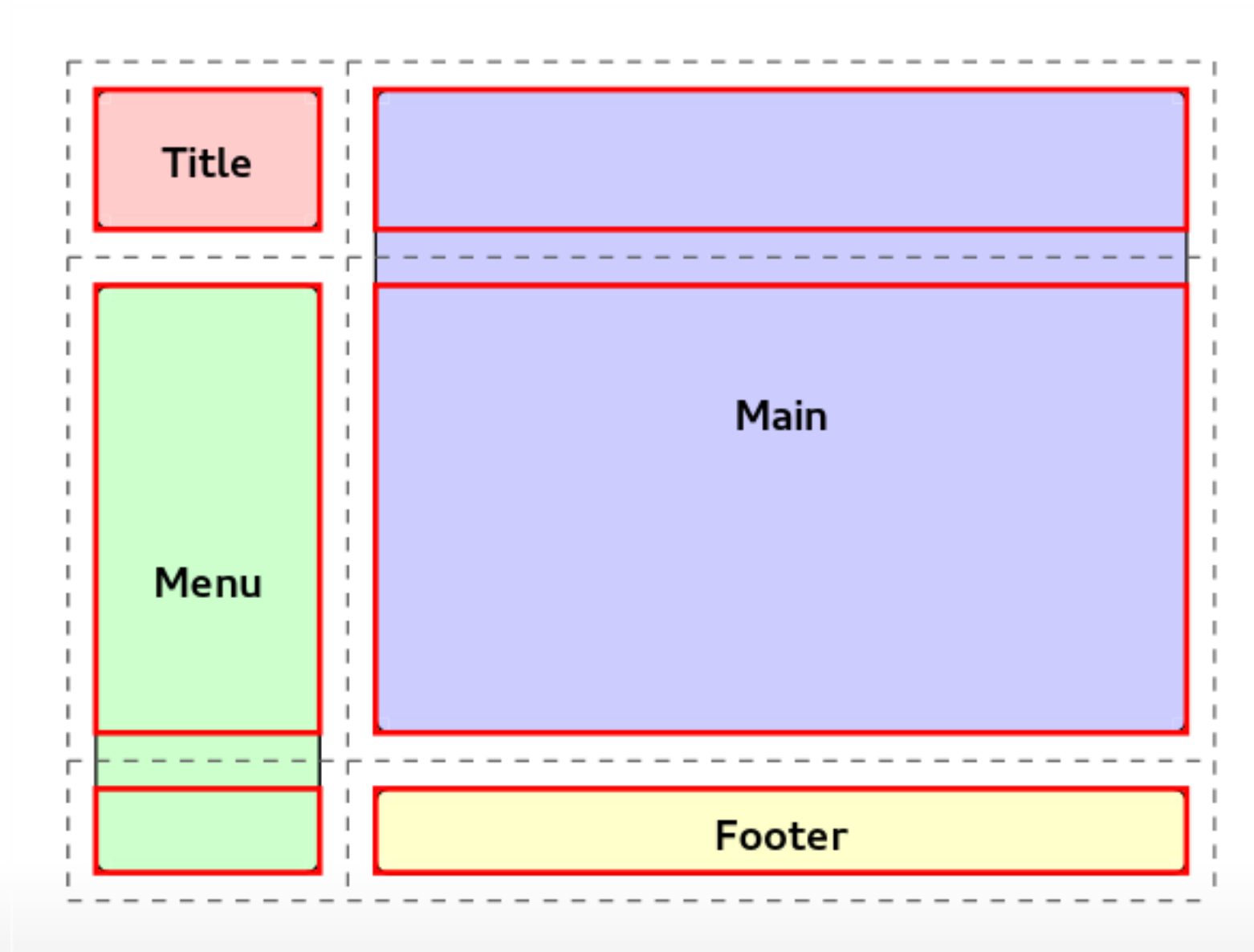
Grid Tracks

Grid track is a generic term for a grid column or grid row



Concepts III

Grid Cells

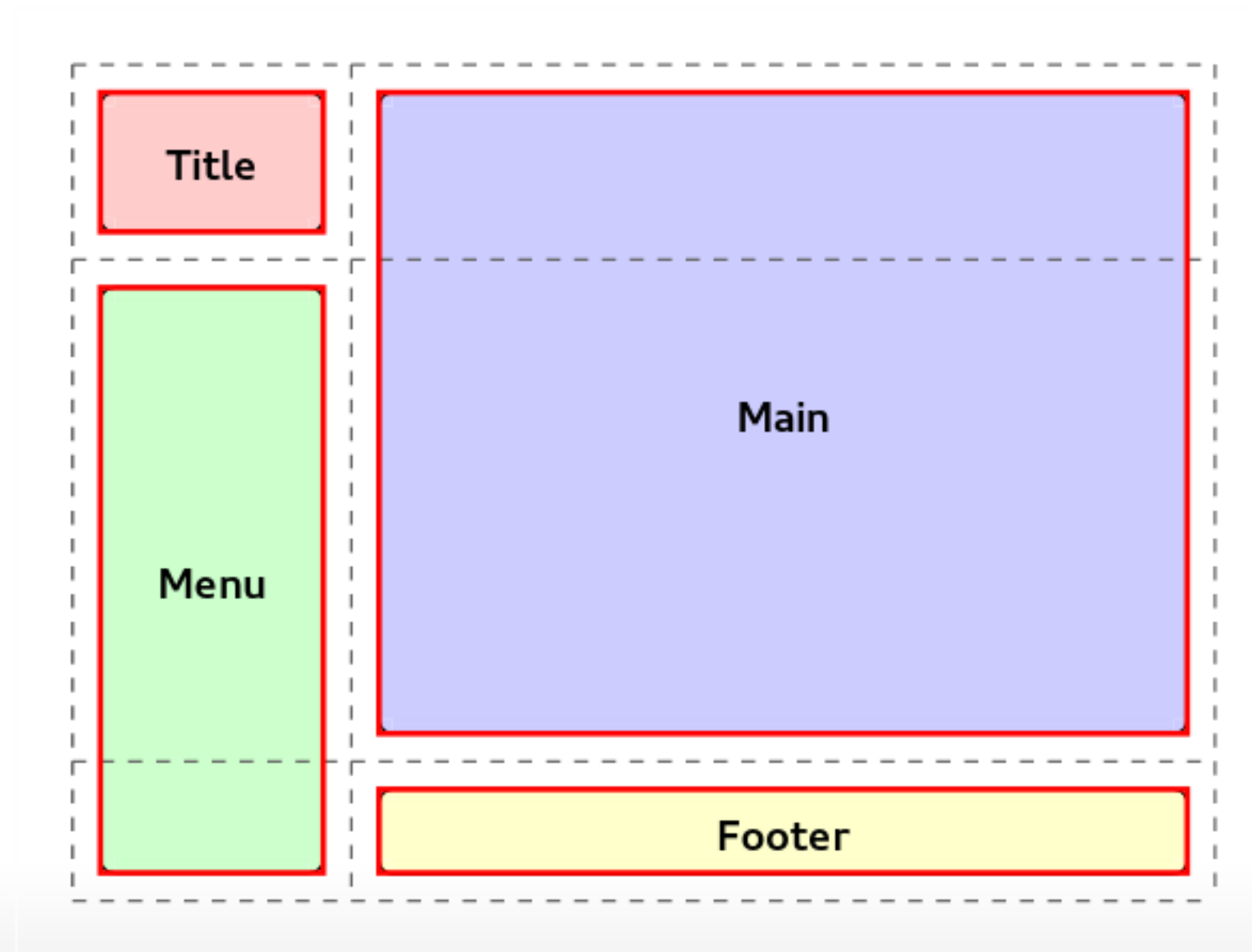


Grid cell is the space between two adjacent row and two adjacent column grid lines



Concepts (and IV)

Grid Areas



Grid area is the logical space used to lay out one or more grid items. It is bound by four grid lines, one on each side of the grid area



How can I use it?

CSS Grid Layout Syntax

<https://flic.kr/p/cYic5>



Syntax

Example: Plain HTML

ALICE'S ADVENTURES IN WONDERLAND Chapters

- Down the Rabbit-Hole
- The Pool of Tears
- A Caucus-Race and a Long Tale

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversations?'

So she was considering in her own mind (as well as she could, for the hot day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.

"Begin at the beginning and go on till you come to the end; then stop."

Lewis Carroll
From [Project Gutenberg](#)



Syntax

Defining the grid

```
/* Definition */
.grid {
  display: grid;
  grid-template-columns: 225px auto minmax(min-content, 9em);
  grid-template-rows: auto auto auto;
}
```

CSS

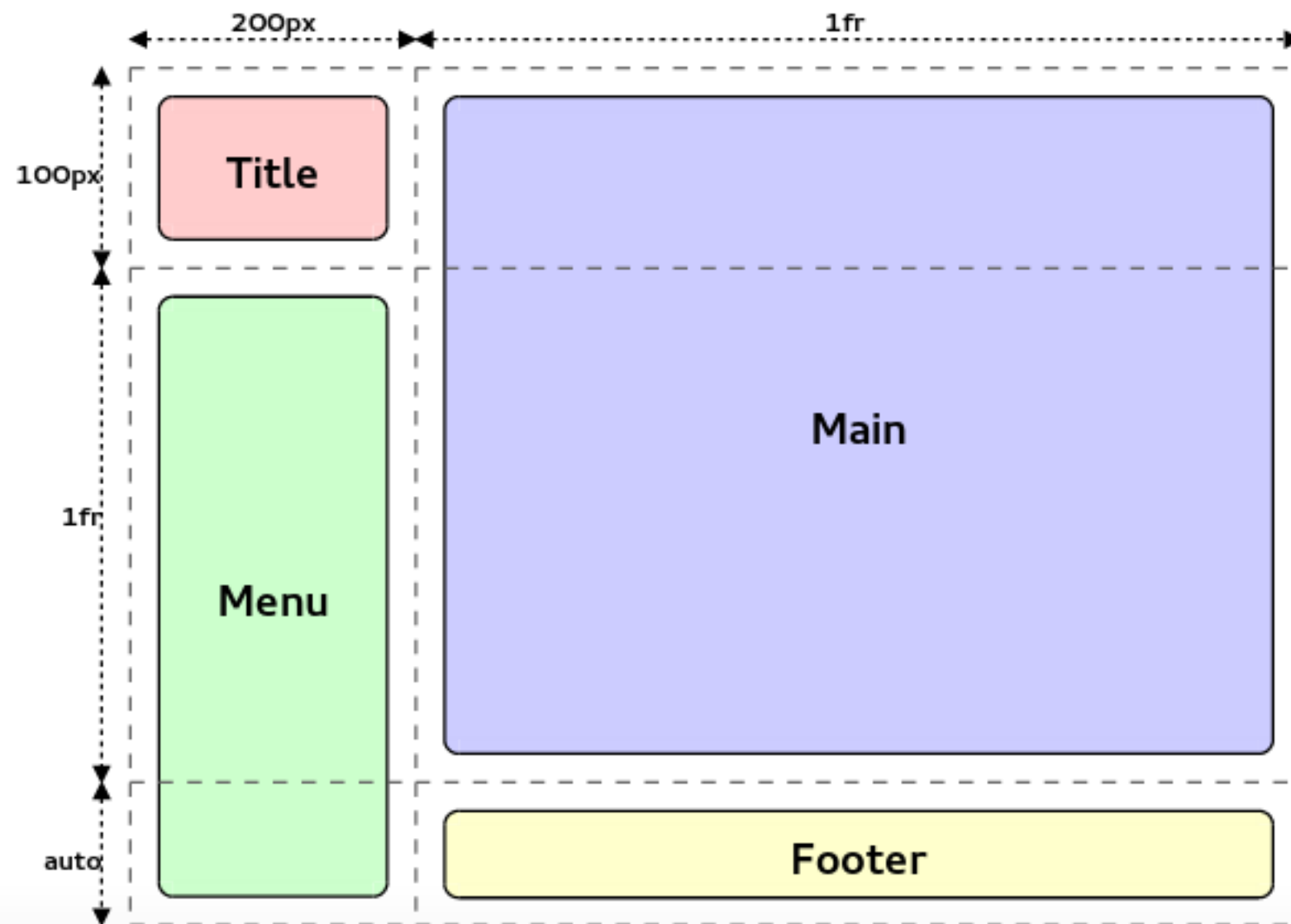
```
<div class="grid">
  <div>ALICE'S ADVENTURES IN WONDERLAND</div>
  <div>Chapters
    <ul>
      <li>Down the Rabbit-Hole</li>
      <li>The Pool of Tears</li>
    </ul>
  </div>
  <p>Begin at the beginning and</p>
  <div>Lewis Carroll</div>
</div>
```

HTML



Syntax

Track Breadths



- length: 10px
- percentage: 25%
- free space: 0.5fr
- max-content
- min-content
- minmax(min, max)
- auto

Everything is really a `minmax(min, max)`



Syntax

Example: Defining the grid

ALICE'S
ADVENTURES IN
WONDERLAND

Chapters

- Down the Rabbit-Hole
- The Pool of Tears
- A Caucus-Race and a Long Tale
- The Rabbit Sends in a Little Bill
- Advice from a Caterpillar
- Pig and Pepper
- A Mad Tea-Party
- The Queen's Croquet-Ground
- The Mock Turtle's Story
- The Lobster Quadrille
- Who Stole the Tarts?
- Alice's Evidence

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book'

*"Begin at the beginning
and go on till you come
to the end; then stop."*

Lewis Carroll

From [Project Gutenberg](#)



Oops... We need to position the items!



Positioning the items

Using longhands

CSS

```
.title {  
  grid-row-start: 1;  
  grid-row-end: 2;  
  grid-column-start: 2;  
  grid-column-end: 3;  
}  
.nav {  
  grid-row-start: 2;  
  grid-row-end: 3;  
  grid-column-start: 1;  
  grid-column-end: 2;  
}
```

HTML

```
<div class="grid">  
  <div class="title">ALICE'S ADVENTURES IN WONDERLAND</div>  
  <div class="nav">Chapters</div>  
  ...  
</div>
```



Positioning the items

Using shorthands

Using `grid-row` and `grid-column`

```
.title {  
  grid-row: 1 / 2;  
  grid-column: 2 / 3;  
}  
.nav {  
  grid-row: 2 / 3;  
  grid-column: 1 / 2;  
}
```

CSS

Or using `grid-area`

```
.title { grid-area: 1 / 2 / 2 / 3; }  
.nav { grid-area: 2 / 1 / 3 / 2; }
```

CSS



Positioning the items

Chapters

- Down the Rabbit-Hole
- The Pool of Tears
- A Caucus-Race and a Long Tale
- The Rabbit Sends in a Little Bill
- Advice from a Caterpillar
- Pig and Pepper
- A Mad Tea-Party
- The Queen's Croquet-Ground
- The Mock Turtle's Story
- The Lobster Quadrille
- Who Stole the Tarts?
- Alice's Evidence

ALICE'S ADVENTURES IN WONDERLAND

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversations?'

So she was considering in her own mind (as well as she could, for the hot day made her feel very sleepy and stupid), whether the pleasure of making a daisy-chain would be worth the trouble of getting up and picking the daisies, when suddenly a White Rabbit with pink eyes ran close by her.

From [Project Gutenberg](#)

"Begin at the beginning and go on till you come to the end; then stop."

Lewis Carroll



Can we make it easier?

Named Grid Lines and Named Areas



<https://flickr.com/photos/fo...>

Positioning the items

Named Grid Lines

```
.grid {  
  display: grid;  
  grid-template-columns: (start first) 200px (middle) 200px (end last);  
}  
.one { grid-column: start / middle; }  
.two { grid-column: first / last; }
```

CSS

```
<div class="grid">  
  <div class="one"></div>  
  <div class="two"></div>  
</div>
```

HTML

Example grid



Positioning items

Grid Areas

```
.grid {  
  display: grid;  
  grid-template-areas: " top top ."  
                      " bar . ."  
                      " bar . bottom";  
}  
.box-one { grid-area: top; }  
.box-two { grid-area: bar; }  
.box-three { grid-area: bottom; }
```

CSS

Example grid



Positioning Items

Implicit Named Areas and Implicit Named Lines

CSS

```
.grid {  
  display: grid;  
  grid-template-areas: "top top ."  
                      "bottom . .";  
  grid-template-rows: (line-start) 50px 100px (line-end);  
  grid-template-columns: 50px 100px (line-start) 150px (line-end);  
}  
.red-box { grid-area: bottom; } /* explicit area name */  
.green-box { grid-area: line; } /* implicit area name */  
.blue-box { grid-column: bottom-start / top-end; grid-row: 1; } /* implicit grid lines */
```

Example grid



Responsive Design

Using Media Queries

CSS

```
@media (orientation: portrait) {  
  .grid {  
    grid-template-areas: "title"  
                        "links"  
                        "content"  
                        "footer";  
  }  
}  
  
@media (orientation: landscape) {  
  .grid {  
    grid-template-areas: " . title . "  
                        "links content content"  
                        "links content content"  
                        " . footer footer ";  
  }  
}
```

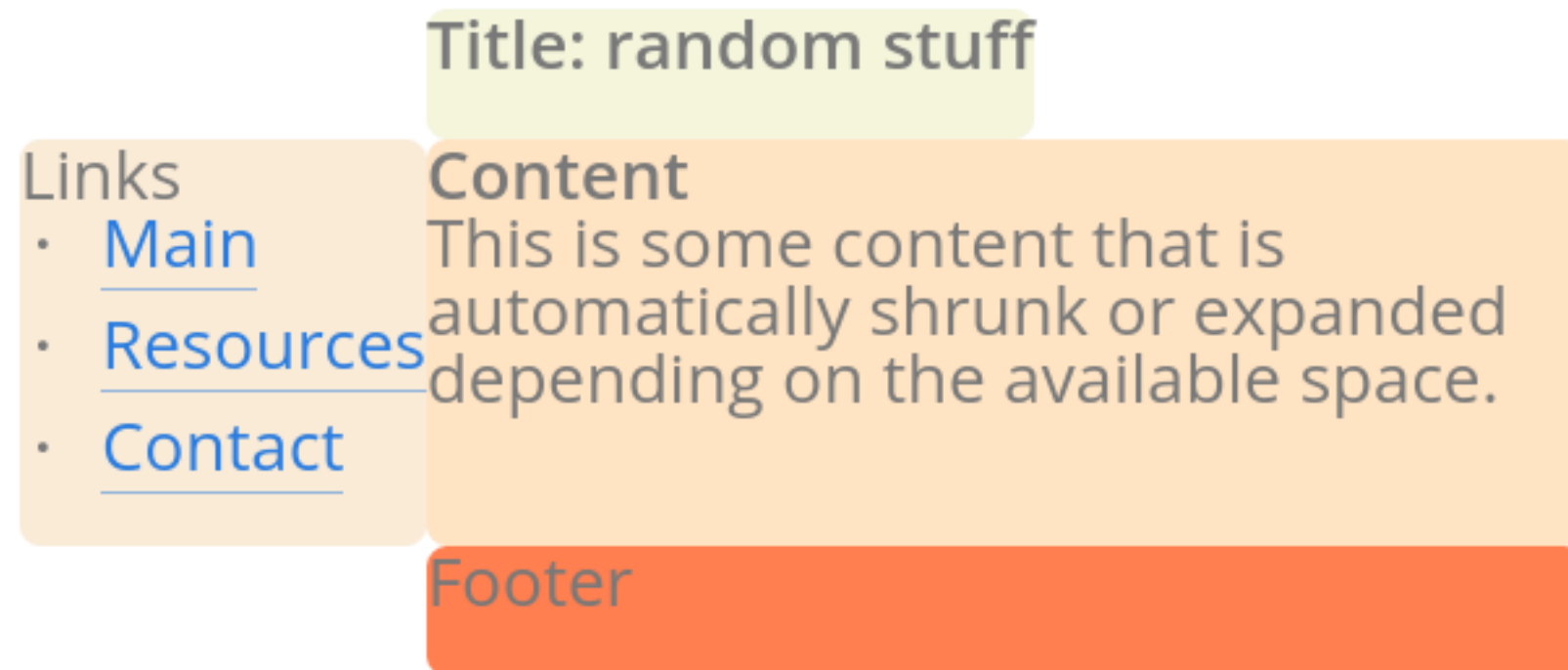
HTML

```
<div class="grid">  
  <div class="title"></div>  
  ...  
</div>
```



Responsive Design

Using Media Queries



Recent Developments

Alignment, Justification and Automatic Placement

<https://flic.kr/p/oNo72Z>

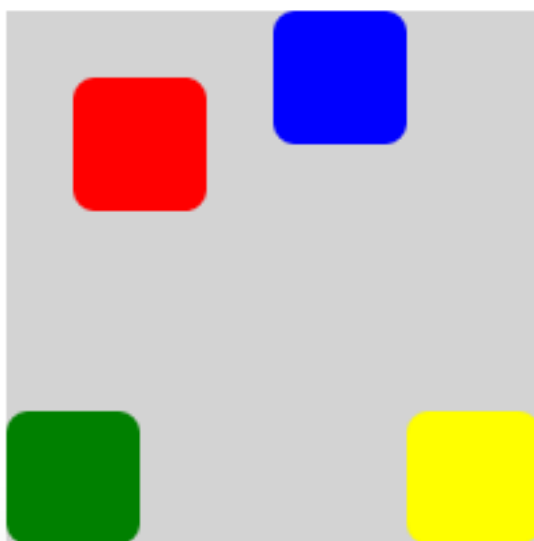


Alignment & Justification

CSS

```
.grid {  
  display: grid;  
  grid-template-areas: "one two"  
                      "three four";  
}  
.one { grid-area: one;   align-self: center; justify-self: center; }  
.two { grid-area: two;   align-self: start;                        }  
.three { grid-area: three; align-self: end;                        }  
.four { grid-area: four;  align-self: end;   justify-self: end; }
```

Example grid



Automatic Placement

Setting up a typical form

```
form {  
  display: grid;  
  grid-auto-flow: row;  
}  
label      { grid-column: 1; }  
input, textarea { grid-column: 2; }
```

CSS

```
<form>  
  <label>Name</label>  
  <input></input>  
  <label>Surname</label>  
  <input></input>  
  <label>Address</label>  
  <textarea></textarea>  
  <label>Mobile number</label>  
  <input type="tel"></input>  
  <label>Email</label>  
  <input type="email"></input>  
</form>
```

HTML



Automatic Placement

Form created using a Grid and Automatic Placement

Name	Jane
Surname	Doe
Address	Yosemite Valley
Mobile number	+19 982 834 948
Email	jane.doe@gmail.com



State of the Grid

Recent Changes, Current Status and Roadmap



<https://filc.kr/p/o3YBwv>

State of the Grid

Changes since BlinkOn2

- Moved grid position resolution to its own class
- Moved painting code to its own class
- Support for spanning items in grid auto placement algorithm
- Support for multiple modes in grid auto placement algorithm
- Improved margin, padding and border support
- Support for implicit grid lines and areas
- Support for `justify-items`, `justify-self`, `align-items` and `align-self`
- Several corner case bugs detected by the fuzzinator



State of the Grid

Current status

Specs feature	Ready?
Grid containers (<code>display: grid/inline-grid</code>)	Yes
Reordered grid items (<code>order</code> property)	Yes
Explicit grid properties (<code>grid-template-rows</code> and <code>grid-template-columns</code>)	Yes
Named grid lines	Yes
<code>repeat()</code> syntax and flexible length (<code>fr</code>) support	Yes
Subgrids	No
Grid areas support (<code>grid-template-areas</code>)	Yes



State of the Grid

Current status (II)

Specs feature	Ready?
Implicit grid areas support	Yes
Auto-generated rows&columns (<code>grid-auto-rows</code> and <code>grid-auto-columns</code>)	Yes
Auto-placement (<code>grid-auto-flow</code>)	Yes
Other shorthands (<code>grid</code> , <code>grid-template</code>)	Yes
Common patterns in grid placement (named areas, numeric indexes, named lines...)	Yes
Grid line based placement	Yes
Placement shorthands (<code>grid-column</code> , <code>grid-row</code> and <code>grid-area</code>)	Yes



State of the Grid

Current status (and III)

Specs feature	Ready?
Absolutely positioned grid items	Partially
Grid item placement algorithm	Yes
Column/Row axis alignment (<code>justify-self</code> , <code>justify-items</code> , <code>align-self</code> and <code>align-items</code>)	Yes
Grid alignment (<code>justify-content</code> and <code>align-content</code>)	Partially
Z-axis ordering	Yes
Track sizing algorithm	Yes (to be rewritten)
Fragmentation	No (to be specified)



State of the Grid

Roadmap

- Subgrids (likely for Level 2)
- Fragmentation (still under discussion inside CSSWG)
- Performance optimizations (we need actual use cases)
- Support for different writing modes
- More tests. Ongoing effort to create a W3C test suite
- Selection
- Integrate Grid in DevTools (a.k.a. WebInspector support)
- Rewrite the Track Sizing Algorithm



Hacking the Grid

RenderGrid and friends

<https://500px.com/photo/73135107/500px-by-sascha-kohlmann>

Hacking the Grid

Source Code Files

- Most of the code is in `core/rendering/RenderGrid.cpp`
 - Track Sizing Algorithm
 - Auto Placement Algorithm
- Parser and style resolver code
 - `css/CSSComputedStyleDeclaration.cpp`
 - `css/CSSGridLineNamesValue.cpp`
 - `css/CSSGridTemplateAreasValue.cpp`
 - `css/parser/CSSPropertyParser.cpp`
 - `css/resolver/StyleBuilderConverter.cpp`
 - `css/resolver/StyleBuilderCustom.cpp`



Hacking the Grid

Source Code Files (and II)

- Rendering code
 - `rendering/RenderGrid.cpp`
 - `rendering/style/GridCoordinate.h`
 - `rendering/style/GridLength.h`
 - `rendering/style/GridPosition.h`
 - `rendering/style/GridResolvedPosition.h`
 - `rendering/style/GridTrackSize.h`
- Painting code
 - `paint/GridPainter.cpp`



Acknowledgements



igalia

Bloomberg

- **Bloomberg** is sponsoring our work in CSS Grid Layout



Thank You!

igalia.com/browsers

[www people.igalia.com/svillar/](https://www.people.igalia.com/svillar/)





igalia