CSS - 1   Cheatsheet

Selectors → Element type

```
p {
    color : red;
}
```

Class Selector

```
.heading {
    background-color : yellow;
}
```

ID Selector

```
#heading {
    background-color : red;
}
```

Descendant Selector

```
div p {
    font-size : 24px;
}
```
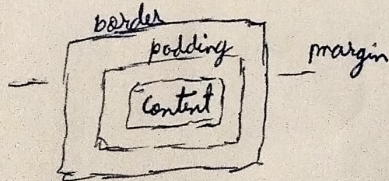
Child Selector

```
ul > li {
    list-style-type : square;
}
```

Attribute Selector

```
input [type = "text"] {
    border : 1px solid red;
}
```

Box Model →


```
.box {
    padding : 10px;
    margin : 20px;
    border : 1px solid red;
}
```

```css
    width: 200px;
    height: 100px;
}
```

Positioning →
```css
.head1 {
    position: relative;
    top: 10px;
    left: 5px;
}

.head1 {
    position: absolute;
    top: 0;
    right: 0;
}

.head1 {
    position: fixed;
    bottom: 0;
    left: 0;
}
```

Display →
```css
.div1 {
    display: block;
}

.div1 {
    display: inline;
}

.div1 {
    display: flex;
    justify-content: center;
    align-items: center;
}
```

Grid → To create complex grid-based layouts in CSS.

```css
.grid-container {
    display: grid;
}
```

```css
.grid-container {
    display: grid;
    grid-template-rows: 100px 200px;
    grid-template-columns: 1fr 2fr;
}
```

```css
.grid-container {
    display: grid;
    grid-gap: 10px;
}
```

```css
.grid-container {
    display: grid;
    grid-template-areas: "header header"
                         "body   body"
                         "footer footer"
}
```

```css
.grid-container {
    display: grid;
    grid-auto-flow: column;
}
```

Animation ->
```
@keyframes fade-in{
        0./.{
                opacity : 0;
        }
    100/. {
                opacity : 1;
        }
}

• element {
        animation: fade-in 1s ease-in 0.5s forwards ;
}
```

Properties -> animation-name, ~~dur~~ animation-duration, animation-timing-function
    (ease, linear, ease-in-out), animation-delay, animation-iteration-count,
    animation-direction (normal, reverse, alternate), animation-fill-mode
    (forwards, backwards, both, none), animation-play-state (running,
    paused).