Individual Characters

.	Any Character
\d	Digit (0-9)
\D	Not Digit
\w	Word Character
\W	Not Word Character
\s	White Space
\S	Not White Space

Anchors

\b	Word Boundaries
\B	Not Word Boundary
^	Beginning of String
$	End of String

Sets

[]	Matches Characters Inside of Brackets
[^]	Matches Characters Not in Brackets
|	Either or
()	Group

Quantifiers

*	0 or More
+	1 or More
?	0 or 1
{5}	Exactly That Many
{5,6}	Range of That Many {min, max}

Must . . . Escape

. [{ ( ) \ ^ $ | ? * + /

Phone Numbers

- \d\d\d.\d\d\d.\d\d\d\
- \d\d\d[-.]\d\d\d[-.]\d\d\d\d
- [89]00[.-]\d\d\d[-.]\d\d\d\d
- \d{3}[.-]\d{3}[.-]\d{4}

Hey Mister/Miss

- Mr\.?\s[A-Z]\w*
- M(rs|r|s)\.?\s[A-Z]\w*

Back References

- https://www.stonecoldmagic.com
- http://magicreviewed.com
- https://slcc.edu
- https://www.utah.gov
- https?://(www\.)?(\w+)(\.\w+)
- $1, $2, $3 = (www\.), (\w+), (\.\w+) respectively
- $2$3 = xyz.abc

PHP Regex Notes

- $x = array("123-456-7890", "123.456.7890", "123*456*7890", "123_456_7890");
- Regex = \d{3}[-.*_]\d{3}[-.*_]\d{4}
- $pattern = "/\d{3}[-.*_]\d{3}[-.*_]\d{4}/";
  - Note the addition of the forward slashes
- preg_match($pattern, $x[1]); returns 1 or 0;
- preg_replace($pattern, "chicken", $x[1]); returns "chicken"