- `lcfirst(string*)`: Lower Case First Letter.
- `ucfirst(string*)`: Upper Case First Letter.
- `strtoupper(string*)`: String To Upper Case.
- `strtolower(string*)`: String To Lower Case.
- `ucwords(string*, delimter)`: Upper Case First Letters Of String Words.
- `str_repeat(string*, counts*)`: Repeat String **n** Times.

- `implode(sep, arr*)` or `join()` : Join Array Elements In a String.
- `explode(sep*, str*, limit)` or `join()` : Join String Elements In an Array.
- `str_shuffle(str*)` : Shuffle Elements Of The Given String.
- `strrev(str*)` : Reverse String Characters.
- `trim(str*, char_list[...])` : Remove Leading And Ending Whitespaces.
- `ltrim(str*, char_list[...])` : Remove Leading Whitespaces.
- `rtrim(str*, char_list[...])` : Remove Ending Whitespaces.

- `chunk_split(str*, len, end_char)` : Split String To Chunks Of `len` Length And Add `end_char` To The End Of Every Chunk.
- `strlen(str*)` : Returns The Length Of The Given String.
- `str_split(str*, len)` : Split String Elements Into an Array.
- `strip_tags(str*, ...tags)` : Remove Any Existing HTML Tags Inside The String, To Omit A Specific Tag You Can Pass It To The Function (eg:`'<p>'`).
- `nl2br(str*, ClosingTag:bool)`: Convert Newlines Characters To `<br/>` or `<br>` In Case Of `ClosingTag=false`.

- `strpos(str*, value*, start)` : Returns The Index Of First Encouter Value.
- `strrpos(str*, value*, start)` : Returns The Index Of First Encouter Value From The Right.
- `stripos(str*, value*, start)` : Returns The Index Of First Encouter Value With Case Insensitive.
- `stropos(str*, value*, start)` : Returns The Index Of First Encouter Value From The Right With Case Insensitive.
- `substr_count(str*, value*, start, stop_after)` : Returns The Number Of Occurences Of The Value In The String.
  - If `stop_after=len` Is Specified Then The Search Will End In The Index `start + len`.

- `parse_str(str*, arr*)` : Parse a String URL ***Get*** Link That Contains

Data To Array With Keys And Values Containing The Data As In The Example:

- **str**: name=othmane&email=ot@gmail.com
- **arr**: [name => othmane, email => ot@gmail.com]

- `qoutemeta(str*)`: Escape Critical Characters That Can Warm Database Due To Some Security Issues.
  - `normal` => "Text [] () * + ."
  - `with method` => "Text \[\] \(\) * \+ \."
- `str_pad(str*, len*, char=" ", flag↓)`: Add Padding Chars To The String.
  - `STR_PAD_BOTH`: Centering Padding.
  - `STR_PAD_LEFT`: Add Padding To The Left.
  - `STR_PAD_RIGHT`: Add Padding To The Right.
- `strtr(str*, from*, to*)`: Change The `from` Value In String To `to` Value.
- `strtr(str*, arr*)`: Same As The Previous Method, But We Pass An Array Containing The Keys: `from` and Values: `to`.
  - `arr` : ["@" => "O", "#" => "M"]