OCaml

# DATA TYPES

|  |  |
| --- | --- |
| TYPE | EXAMPLE VALUES |
| int | 0,1,2,4, -1, -2 |
| bool | true, false |
| float | decimals |
| string | “Hello World” |
| char | “A”,”Z” |

# Operators

|  |  |
| --- | --- |
| Operator | EXAMPLE VALUES |
| - | Take an int (or a floart) and return its negation |
| not | Take a boll and return its negation |

# Binary Operators

|  |  |
| --- | --- |
| Operator | EXAMPLE VALUES |
| +,-,\*,/ | Take 2 ints and return their sum and so on |
| +.,-.,\*.,/. | Take 2 float and return their sum and so on |
| mod | Take 2 ints and return their integer remainder |
| >,>=,<,<= | Take 2 ints (or float) and return their comparison |
| = | Take 2 values and return equal or not |
| ^ | Take 2 string and return their concatenation into a new string |

# Declarations

* d ::= let id = e ;;
* let pi = 3.141592
* let rec – recursion
* let i = 10 ;;
  + let i = 10 in i-i ;;
  + - : int =0
* i = 3 ;;
  + bool = false
* i\*i ;;
  + int= 100
* let s = “string test” ;;
  + s ;;
    - string = “string test”
* let c = ‘g’
* let d = 22 and h = 3.4 and g =’h’ and s = “string test” ;;