```
Last login: Fri Jan 19 15:20:51 on ttys008
carbon:$ utop
```

```
        Welcome to utop version 2.0.2 (using OCaml version 4.06.0)!
```

```
Type #utop_help for help about using utop.

─( 15:44:15 )─< command 0 >──────────────────────────────────{ counter: 0 }─
utop # 1 + 2 ;;
- : int = 3
─( 15:44:15 )─< command 1 >──────────────────────────────────{ counter: 0 }─
utop # 1 + 2 +
3 + 4 ;;
- : int = 10
─( 15:44:36 )─< command 2 >──────────────────────────────────{ counter: 0 }─
utop # 1 * 3 ;;
- : int = 3
─( 15:45:05 )─< command 3 >──────────────────────────────────{ counter: 0 }─
utop # 2 * 3 ;;
- : int = 6
─( 15:45:23 )─< command 4 >──────────────────────────────────{ counter: 0 }─
utop # 5 < 3 ;;
- : bool = false
─( 15:45:26 )─< command 5 >──────────────────────────────────{ counter: 0 }─
utop # "Hello" ;;
- : string = "Hello"
─( 15:45:36 )─< command 6 >──────────────────────────────────{ counter: 0 }─
utop # "hello " ^ " class ! " ;;
- : string = "hello  class ! "
─( 15:45:47 )─< command 7 >──────────────────────────────────{ counter: 0 }─
utop # 'c' ;;
- : char = 'c'
─( 15:45:58 )─< command 8 >──────────────────────────────────{ counter: 0 }─
utop # 1.0 + 3.14 ;;
Error: This expression has type float but an expression was expected of type
         int
─( 15:46:10 )─< command 9 >──────────────────────────────────{ counter: 0 }─
utop # 1.0 +. 3.14 ;;
- : float = 4.14000000000000057
─( 15:48:11 )─< command 10 >─────────────────────────────────{ counter: 0 }─
utop # 1.0 / 3.0 ;;
Error: This expression has type float but an expression was expected of type
         int
─( 15:48:18 )─< command 11 >─────────────────────────────────{ counter: 0 }─
utop # 1.0 /. 3.0 ;;
- : float = 0.333333333333333315
─( 15:49:16 )─< command 12 >─────────────────────────────────{ counter: 0 }─
utop # 3.14 ;;
- : float = 3.14
```

```
─( 15:49:19 )─< command 13 >──────────────────────────────────{ counter: 0 }─
utop # 1.0;;
- : float = 1.
─( 15:49:52 )─< command 14 >──────────────────────────────────{ counter: 0 }─
utop # op + ;;
Error: Syntax error
─( 15:49:54 )─< command 15 >──────────────────────────────────{ counter: 0 }─
utop # op (+) ;;
Error: Unbound value op
─( 15:50:24 )─< command 16 >──────────────────────────────────{ counter: 0 }─
utop # (+);;
- : int -> int -> int = <fun>
─( 15:50:29 )─< command 17 >──────────────────────────────────{ counter: 0 }─
utop # float_of_int ;;
- : int -> float = <fun>
─( 15:50:32 )─< command 18 >──────────────────────────────────{ counter: 0 }─
utop # 3.14 + 5 ;;
Error: This expression has type float but an expression was expected of type
         int
─( 15:51:53 )─< command 19 >──────────────────────────────────{ counter: 0 }─
utop # 4 / 0 ;;
Exception: Division_by_zero.
─( 15:53:06 )─< command 20 >──────────────────────────────────{ counter: 0 }─
utop # Char.uppercase_ascii ;;
- : char -> char = <fun>
─( 15:53:29 )─< command 21 >──────────────────────────────────{ counter: 0 }─
utop # Char.uppercase ;;
Characters 0-14:
Warning 3: deprecated: Char.uppercase
Use Char.uppercase_ascii instead.
Characters 0-14:
Warning 3: deprecated: Char.uppercase
Use Char.uppercase_ascii instead.
- : char -> char = <fun>
─( 15:55:35 )─< command 22 >──────────────────────────────────{ counter: 0 }─
utop # let x = 4 in x + x ;;
- : int = 8
─( 15:55:40 )─< command 23 >──────────────────────────────────{ counter: 0 }─
utop # let x = 7 ;;
val x : int = 7
─( 15:57:00 )─< command 24 >──────────────────────────────────{ counter: 0 }─
utop # x + 7 ;;
- : int = 14
─( 15:57:20 )─< command 25 >──────────────────────────────────{ counter: 0 }─
utop # let x = 4 ;;
val x : int = 4
─( 15:57:30 )─< command 26 >──────────────────────────────────{ counter: 0 }─
utop # x + 5 ;;
- : int = 9
─( 15:58:04 )─< command 27 >──────────────────────────────────{ counter: 0 }─
```

```
utop # let x = "hello" ;;
val x : string = "hello"
─( 15:58:11 )─< command 28 >──────────────────────────────────{ counter: 0 }─
utop # let x = 5 in let y = 6 in x + y ;;
- : int = 11
─( 15:58:55 )─< command 29 >──────────────────────────────────{ counter: 0 }─
utop # let inc = fun x -> x + 1 ;;
val inc : int -> int = <fun>
─( 15:59:30 )─< command 30 >──────────────────────────────────{ counter: 0 }─
utop # inc 5 ;;
- : int = 6
─( 16:00:47 )─< command 31 >──────────────────────────────────{ counter: 0 }─
utop # inc (inc 6) ;;
- : int = 8
─( 16:01:27 )─< command 32 >──────────────────────────────────{ counter: 0 }─
utop # let circle_area = fun r -> 3.1415 *. r *. r ;;
val circle_area : float -> float = <fun>
─( 16:01:33 )─< command 33 >──────────────────────────────────{ counter: 0 }─
utop # circle_area 4.5 ;;
- : float = 63.6153750000000073
─( 16:06:26 )─< command 34 >──────────────────────────────────{ counter: 0 }─
utop # circle_area 4 ;;
Error: This expression has type int but an expression was expected of type
         float
─( 16:06:32 )─< command 35 >──────────────────────────────────{ counter: 0 }─
utop # let useless = fun r -> r ;;
val useless : 'a -> 'a = <fun>
─( 16:07:06 )─< command 36 >──────────────────────────────────{ counter: 0 }─
utop # let inc x = x +1 ;;
val inc : int -> int = <fun>
─( 16:08:59 )─< command 37 >──────────────────────────────────{ counter: 0 }─
utop # let add x y = x + y ;;
val add : int -> int -> int = <fun>
─( 16:09:58 )─< command 38 >──────────────────────────────────{ counter: 0 }─
utop # add 4 ;;
- : int -> int = <fun>
─( 16:10:26 )─< command 39 >──────────────────────────────────{ counter: 0 }─
utop # let inc = add 1 ;;
val inc : int -> int = <fun>
─( 16:14:36 )─< command 40 >──────────────────────────────────{ counter: 0 }─
utop # inc 5 ;;
- : int = 6
─( 16:14:45 )─< command 41 >──────────────────────────────────{ counter: 0 }─
utop # (add 3) 4 ;;
- : int = 7
─( 16:14:52 )─< command 42 >──────────────────────────────────{ counter: 0 }─
utop # add (3 4) ;;
Error: This expression has type int
       This is not a function; it cannot be applied.
─( 16:17:38 )─< command 43 >──────────────────────────────────{ counter: 0 }─
```

```
utop # add 3 ;;
- : int -> int = <fun>
─( 16:17:45 )─< command 44 >──────────────────────────────────────{ counter: 0 }─
utop # let foo = add 5 ;;
val foo : int -> int = <fun>
─( 16:17:51 )─< command 45 >──────────────────────────────────────{ counter: 0 }─
utop # foo 5 ;;
- : int = 10
─( 16:18:00 )─< command 46 >──────────────────────────────────────{ counter: 0 }─
utop # foo 56 ;;
- : int = 61
─( 16:18:02 )─< command 47 >──────────────────────────────────────{ counter: 0 }─
utop # (add 1) 3 ;;
- : int = 4
─( 16:18:05 )─< command 48 >──────────────────────────────────────{ counter: 0 }─
utop # List.map inc [1;2;3] ;;
- : int list = [2; 3; 4]
─( 16:20:07 )─< command 49 >──────────────────────────────────────{ counter: 0 }─
utop # List.map (add 1) [1;2;3] ;;
- : int list = [2; 3; 4]
─( 16:20:52 )─< command 50 >──────────────────────────────────────{ counter: 0 }─
utop # List.map (add 7) [1;2;3] ;;
- : int list = [8; 9; 10]
─( 16:21:04 )─< command 51 >──────────────────────────────────────{ counter: 0 }─
utop # List.map (fun x -> x + 3 * 5) [1;2;3] ;;
- : int list = [16; 17; 18]
─( 16:21:31 )─< command 52 >──────────────────────────────────────{ counter: 0 }─
utop # inc ;;
- : int -> int = <fun>
─( 16:22:03 )─< command 53 >──────────────────────────────────────{ counter: 0 }─
utop # float_of_int ;;
- : int -> float = <fun>
─( 16:43:21 )─< command 54 >──────────────────────────────────────{ counter: 0 }─
utop # float_of_int 4 ;;
- : float = 4.
─( 16:43:38 )─< command 55 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power n x = if n = 0 then x else x*. power (n-1) x ;;
val power : int -> float -> float = <fun>
─( 16:43:44 )─< command 56 >──────────────────────────────────────{ counter: 0 }─
utop # power 3 3.0 ;;
- : float = 81.
─( 16:45:36 )─< command 57 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power n x = if n = 0 then x else power (n-1) (x*.x) ;;
val power : int -> float -> float = <fun>
─( 16:45:41 )─< command 58 >──────────────────────────────────────{ counter: 0 }─
utop # power 3 3.0 ;;
- : float = 6561.
─( 16:46:07 )─< command 59 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power n x = if n = 0 then 1.0 else x*. power (n-1) x ;;
val power : int -> float -> float = <fun>
```

```
─( 16:46:08 )─< command 60 >──────────────────────────────────{ counter: 0 }─
utop # power 3 3.0 ;;
- : float = 27.
─( 16:46:21 )─< command 61 >──────────────────────────────────{ counter: 0 }─
utop # let cube = power 3 ;;
val cube : float -> float = <fun>
─( 16:46:28 )─< command 62 >──────────────────────────────────{ counter: 0 }─
utop # cube 4.5 ;;
- : float = 91.125
─( 16:46:41 )─< command 63 >──────────────────────────────────{ counter: 0 }─
utop # let add3 x y z =
     let add2 a b = a + b
     in add2 (add2 x y) z) ;;
Error: Syntax error
─( 16:46:45 )─< command 64 >──────────────────────────────────{ counter: 0 }─
utop # let add3 x y z =
     let add2 a b = a + b
     in add2 (add2 x y) z ;;
val add3 : int -> int -> int -> int = <fun>
─( 16:52:59 )─< command 65 >──────────────────────────────────{ counter: 0 }─
utop # let add3 =
  let add2 a b = a + b
  in  fun x ->
         fun y ->
            fun z -> add2 (add2 x y) z ;;
val add3 : int -> int -> int -> int = <fun>
─( 16:53:09 )─< command 66 >──────────────────────────────────{ counter: 0 }─
utop # let a = 4 in a + 6 ;;
- : int = 10
─( 16:54:04 )─< command 67 >──────────────────────────────────{ counter: 0 }─
utop # add2 ;;
Error: Unbound value add2
Hint: Did you mean add or add3?
─( 16:55:42 )─< command 68 >──────────────────────────────────{ counter: 0 }─
utop # add3 1 2 3 ;;
- : int = 6
─( 16:56:45 )─< command 69 >──────────────────────────────────{ counter: 0 }─
utop # ((add3 1) 2) 3 ;;
- : int = 6
─( 16:57:10 )─< command 70 >──────────────────────────────────{ counter: 0 }─
utop # add3 ;;
- : int -> int -> int -> int = <fun>
─( 16:57:45 )─< command 71 >──────────────────────────────────{ counter: 0 }─
utop # let add3 : int -> int -> int -> int =
  let add2 a b = a + b
  in  fun x ->
         fun y ->
            fun z -> add2 (add2 x y) z ;;
val add3 : int -> int -> int -> int = <fun>
─( 16:58:10 )─< command 72 >──────────────────────────────────{ counter: 0 }─
```

```
utop # let add3 : int -> (int -> (int -> int)) =
  let add2 a b = a + b
  in  fun x ->
         fun y ->
            fun z -> add2 (add2 x y) z ;;
val add3 : int -> int -> int -> int = <fun>
─( 16:58:48 )─< command 73 >──────────────────────────────────{ counter: 0 }─
utop # ((add3 1) 2) 3 ;;
- : int = 6
─( 16:59:16 )─< command 74 >──────────────────────────────────{ counter: 0 }─
utop # ((add3 1) 2) ;;
- : int -> int = <fun>
─( 16:59:21 )─< command 75 >──────────────────────────────────{ counter: 0 }─
utop # let foo = ((add3 1) 2) ;;
val foo : int -> int = <fun>
─( 17:07:34 )─< command 76 >──────────────────────────────────{ counter: 0 }─
utop # foo 3 ;;
- : int = 6
─( 17:07:43 )─< command 77 >──────────────────────────────────{ counter: 0 }─
utop #
```

Arg	Array	ArrayLabels	Assert_failure	Bigarray	Buffer	Bytes	BytesLabels	Callb