

Last login: Mon Apr 2 15:29:11 on ttys004

carbon:\$ cd Search

carbon:\$ utop

me to utop version 2.0.2 (using OCaml version 4.

Type #utop_help for help about using utop.

```
-( 15:39:50 )-< command 0 >-----{ counter: 0 }-
utop # let inc = fun x -> x + 1 ;;
val inc : int -> int = <fun>
-( 15:40:01 )-< command 1 >-----{ counter: 0 }-
utop # let inc x = x + 1 ;;
val inc : int -> int = <fun>
-( 15:40:01 )-< command 2 >-----{ counter: 0 }-
utop # print_endline "Hello" ; 1+2 ;;
Hello
- : int = 3
-( 16:05:50 )-< command 3 >-----{ counter: 0 }-
utop # 1+2 ; 3+4 ;;
Characters 0-3:
Warning 10: this expression should have type unit.
Characters 0-3:
Warning 10: this expression should have type unit.
- : int = 7
-( 16:06:36 )-< command 4 >-----{ counter: 0 }-
utop # print_endline ;;
- : string -> unit = <fun>
-( 16:06:36 )-< command 5 >-----{ counter: 0 }-
utop # #use "search_exceptions.ml";;
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
exception FoundSubSet of int list
val subsetsum_exn_on_found : int list -> int list option =
  <fun>
exception KeepLooking
-( 16:07:03 )-< command 6 >-----{ counter: 0 }-
utop # subsetsum_exn_on_found s ;;
- : int list option = Some [-6; 5; 1]
-( 16:08:53 )-< command 7 >-----{ counter: 0 }-
```

```

utop # #use "search_exceptions.ml";;
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
exception FoundSubSet of int list
val subsetsum_exn_on_found : int list -> int list option =
  <fun>
exception KeepLooking
val subsetsum_exn_not_found : int list -> int list option =
  <fun>
-( 16:09:02 )-< command 8 >-----{ counter: 0 }-
utop # subsetsum_exn_not_found s ;;
- : int list option = Some [-6; 5; 1]
-( 16:15:37 )-< command 9 >-----{ counter: 0 }-
utop # #use "search_exceptions.ml";;
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
exception FoundSubSet of int list
val subsetsum_exn_on_found : int list -> int list option =
  <fun>
exception KeepLooking
val subsetsum_exn_not_found : int list -> int list option =
  <fun>
-( 16:15:44 )-< command 10 >-----{ counter: 0 }-
utop # subsetsum_exn_not_found s ;;
- : int list option = Some [1; 5; -6]
-( 16:15:58 )-< command 11 >-----{ counter: 0 }-
utop # s;;
- : int list = [1; 3; -2; 5; -6]
-( 16:15:58 )-< command 12 >-----{ counter: 0 }-
utop #

```

Arg	Array	ArrayLabels	Assert_failure	Bigarray	Buffer	Bytes	Byt
-----	-------	-------------	----------------	----------	--------	-------	-----