

Last login: Mon Apr 2 13:19:49 on ttys006

carbon:\$ cd Search/

carbon:\$ utop

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

Type #utop_help for help about using utop.

```
-( 13:46:39 )-< command 0 >-----{ counter: 0 }-
utop # print_endline "Hello" ;;
Hello
- : unit = ()
-( 13:46:39 )-< command 1 >-----{ counter: 0 }-
utop # print_endline "Hello" ; 1 + 2 ;;
Hello
- : int = 3
-( 13:46:48 )-< command 2 >-----{ 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
-( 13:47:00 )-< command 3 >-----{ counter: 0 }-
utop # 1 + 2 ; print_endline "Hello" ;;
Characters 0-5:
Warning 10: this expression should have type unit.
Characters 0-5:
Warning 10: this expression should have type unit.
Hello
- : unit = ()
-( 13:47:15 )-< command 4 >-----{ 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
File "search_exceptions.ml", line 55, characters 9-26:
Error: This expression has type unit but an expression was expected of type
      int list option
-( 13:48:28 )-< 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
-( 13:50:03 )-< command 6 >-----{ counter: 0 }-
utop # subsetsum_exn_on_found s ;;
```

```

- : int list option = Some [-6; 5; 1]
-( 13:51:49 )-< 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
-( 13:51:56 )-< command 8 >-----{ counter: 0 }-
utop # subsetsum_exn_on_found s ;;
- : int list option = Some [1; 5; -6]
-( 13:52:14 )-< command 9 >-----{ counter: 0 }-
utop # s ;;
- : int list = [1; 3; -2; 5; -6]
-( 13:52:15 )-< command 10 >-----{ 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>
-( 13:52:19 )-< command 11 >-----{ counter: 0 }-
utop # subsetsum_exn_not_found s ;;
- : int list option = Some [1; 5; -6]
-( 13:59:58 )-< command 12 >-----{ counter: 0 }-
utop #

```

Arg	Array	ArrayLabels	Assert_failure	Bigarray	Buffer	Bytes	BytesLabels	Callbac
-----	-------	-------------	----------------	----------	--------	-------	-------------	---------