Introduction to OCAML

```
# 1 + 2;;
```

```
# let a = 3 + 4;;
```

```
# a * a;;
```

```
# let dbl = fun x -> x * 2;;
```

```
# dbl 10;;
```

```
# dbl (dbl 10);;
```

```
# let triple = fun x => x * 3;;
```

```
# (fun x -> x + 1) 10;;
```

```
# let inc x = x + 1;;
```

```
# inc 10;;
```

```
# let app5 f = f 5;;


# app5 inc;;


# app5 dbl;;


# let b = a * 2;;


# let adda x = x + a;;


# adda 10;;


# let a = 42;;


# b;;


# adda 10;;


# let f n = if n > 10 then 2 * n else n * n;;


# f 20;;


# f 5;;
```

```
# let rec fact n =
    if n = 0 then
      1
    else
      n * (fact (n-1));;




# fact 5;;




# fact 12;;




# fact 13;;




# 1=1;;




# 2=3;;




# "foobar";;




# String.length "foobar";;




# String.get "foobar" 5;;




# (2 * 3, 4 < 5, "foo" ^ "bar", String.get "baz" 2);;




# let step (a,b) = (a + b, a*b);;
```

```
# step (1,2);;


# step (step (1,2));;


# let rec stepuntil ((a,b),limit) =
    if a >= limit then
      (a,b)
    else
      stepuntil(step(a,b),limit);;




# stepuntil ((1,2), 100);;



# "baz" ^ "quux" ^ (string_of_int 17);;



# let print_pair (a,b) =
    print_string ("(" ^ (string_of_int a) ^ ","
                  ^ (string_of_int b) ^ ")\n");;



# let rec stepuntil ((a,b),limit) =
   if a >= limit then
     (a,b)
   else
    (print_pair (a,b);
     stepuntil(step(a,b),limit));;



# stepuntil ((1,2),100);;
```

```
# [1+2;3*4;6-5];;
```

```
# [1=1;2>3];;
```

```
# [1+2;3=4];;
```

```
# List.length [1+2;3*4;6-5];;
```

```
# let ns = [1+2;3*4;6-5];;
```

```
# let rec sum ns =
    match ns with
      [] -> 0
    | (n::ns') -> n + sum(ns');;
```

```
# sum ns;;
```

```
# sum [1;4;8;2;3;1;5;10];;
```