



- ▶ Introduction and overview
- ▼ **Basic types, definitions and functions**

#### Table of Contents


##### Basic Data Types

Week 1 Échéance le déc 12, 2016 at 23:30 UTC 


##### More Data Types

Week 1 Échéance le déc 12, 2016 at 23:30 UTC 


##### Expressions

Week 1 Échéance le déc 12, 2016 at 23:30 UTC 


##### Definitions

Week 1 Échéance le déc 12, 2016 at 23:30 UTC 



Week 1 Échéance le déc 12, 2016 at 23:30 UTC 

##### Recursion

Week 1 Échéance le déc 12, 2016 at 23:30 UTC 

- ▶ Basic data structures
- ▶ More advanced data structures
- ▶ Higher order functions
- ▶ Exceptions, input/output and imperative constructs
- ▶ Modules and data abstraction

## STRING IDENTIFIERS (2/2 points)

Suppose that a variable `word` exists and is a string.

Define a variable `sentence` that uses 5 string concatenations to create a string containing 9 times `word`, separated by commas ( `' , '` ).

This time, experiment with defining local `let ... in` s to store the partial results.

## YOUR OCAML ENVIRONMENT

```
1 let word1 = word;;
2 let sentence =
3   let word = word ^ "," in
4   let word = word ^ word in
5   let word = word ^ word in
6   let word = word ^ word in
7   let word = word ^ word1 in word;;
8
```

Evaluate >

Switch >>

Typecheck

Reset Templ

Rechercher un cours



Full-screen |

Check & Sa

Exercise complete (click for details)

2 pts

This time, word is "bark" .

Found sentence with compatible type.

Correct value "bark,bark,bark,bark,bark,bark,bark,bark,bark"

1 pt

Testing how many times you concatenated.

Correct value 5

1 pt

[A propos](#)

[Aide](#)

[Contact](#)

[Conditions générales d'utilisation](#)

[Charte utilisateurs](#)

[Politique de confidentialité](#)

[Mentions légales](#)



Rechercher un cours

