

# WEB 2

L2INFO - PART 4

# PHP IS AN OBJECT LANGUAGE

**CLASSES ARE TEMPLATES**  
**OBJECTS ARE INSTANCES**

**A CLASS IS STRUCTURE AND BEHAVIOUR**

# OBJECT TYPES

- Class
- Interface
- Trait
- Abstract

# CLASS

```
class Person {  
    private $name;  
  
    public function setName($name) {  
        $this->name = $name;  
    }  
  
    public function displayName() {  
        echo $this->name;  
    }  
}  
  
$michiel = new Person();  
$michiel->setName('michiel');  
$michiel->displayName();
```

# METHODS

- `__construct()`
- `__destruct()`
- `__toString()`

# VISIBILITY

- abstract
- static
- final



# INHERITANCE

EXTENDS & IMPLEMENTS

# PHP CONFIGURATION

- Uses configuration files
- Can be overridden at runtime with specific functions
- Or using `ini_set`

# PHP AS APACHE MODULE

```
<?php phpinfo(); ?>
```

# PHP ON COMMAND LINE

```
$ php -i
```

# PHP CLI

Offers a lot of possibilities. Can be used as to execute files. Can launch a webserver. Can check files syntax

# OTHER

- Error and exception handling

# PHP EXECUTION

3 levels

- ERROR
- WARNING
- NOTICE

# ERROR CONFIGURATION

- `error_reporting`
- `display_errors`



# ERROR LEVELS IN CONFIGURATION

- E\_ALL
- E\_ERROR
- E\_WARNING
- E\_NOTICE
- E\_NONE

# EXCEPTIONS

```
throw new \Exception('Something went wrong !');
```

# EXCEPTIONS

```
try {  
    callFunctionWithException();  
} catch (\Exception $e) {  
    // Do whatever you want  
} catch (\ProductNotFoundException $a) {  
    // Do something else  
}
```

# CREATE NEW EXCEPTIONS

```
class ProductNotFoundException extends \Exception {  
  
}
```

# EXERCICES

For each of the following exercises, you must use object oriented design. One file per exercise.

# EXERCICE 1

Create an array with at least 5 people. Each person must have a first name, a last name, a gender and a birthdate.

Display a table with the list of people. For each line, display the full name, the age and gender. Display the number of days before their birthday.

- Use object design for a Person
- Use the php object DateTime

# EXERCICE 2

## TRAVELLING SALESMAN PROBLEM

Find the shortest and the longest way to go through all points.

	A	B	C	D	E
A	-	7	12	3	9
B	9	-	12	7	5
C	8	16	-	11	15
D	3	5	6	-	9
E	4	11	3	5	-

# EXERCICE 3

## SIMULATING A SHOPPING PAGE

Create a single page displaying a list of products. Display an "Add to basket" button for each product. Display each product of basket, the quantity ordered and the price for each product. Add a clear basket button.