```python
from dataclasses import dataclass

@dataclass
class Course:
    title: str
    date: str
    teacher: str

    def print(self):
        print(f"Le {self.date}, c'est cours {self.title} avec {self.teacher}.")

icc_cours8 = Course(title="Modélisation et classe",
                    date="10.11.2023",
                    teacher="Patrick Wang")
icc_cours8.print()
```

Information, Calcul et Communication

Partie Programmation

Cours 8 : Modélisation et classes

10.11.2023

Patrick Wang

Au sommaire cette semaine

1. Bilan post-midterm
2. Introduction du projet
3. Les «dataclasses»

Au sommaire cette semaine

1. Bilan post-midterm
2. Introduction du projet
3. Les «dataclasses»

1. Bilan post-midterm

Suite du semestre

- Les concepts :
  - Variables
  - Structures de contrôle (`if`, `for`, `while`)
  - Les fonctions
  - Les structures de données (`List`, `Tuple`, `Set`, `Dict`)
- Leurs utilisations :
  - Créer des branchements conditionnels
  - Répéter des instructions
  - Parcourir (indice ou élément ?) des chaînes de caractères, listes, ensembles, dictionnaires
  - Décomposer un problème en sous-problèmes plus facile à résoudre

1. Bilan post-midterm

Suite du semestre

- Aujourd'hui : voir une nouvelle façon de modéliser un problème
- Deux semaines consacrées au projet
- Puis, quelques notions plus ou moins avancées

Au sommaire cette semaine

1. Bilan post-midterm
2. Introduction du projet
3. Les «dataclasses»

2. Introduction du projet

Seam carving pour redimensionner une image

Selon une idée et projet original de Jamila Sam et Barbara Jobstmann

- Image rectangulaire 391px × 265px
- Comment la redimensionner (par exemple en image carrée) sans la déformer ni la rogner sur les bords ?

2. Introduction du projet

Selon une idée et projet original de Jamila Sam et Barbara Jobstmann

Seam carving pour redimensionner une image

Redimensionnée en 265px × 265px

2. Introduction du projet

Selon une idée et projet original de Jamila Sam et Barbara Jobstmann

Seam carving pour redimensionner une image

Rognée en 265px × 265px

2. Introduction du projet

Selon une idée et projet original de Jamila Sam et Barbara Jobstmann

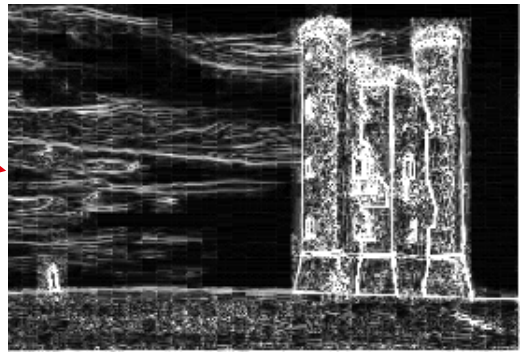
Seam carving pour redimensionner une image

Redimensionnée avec seam carving

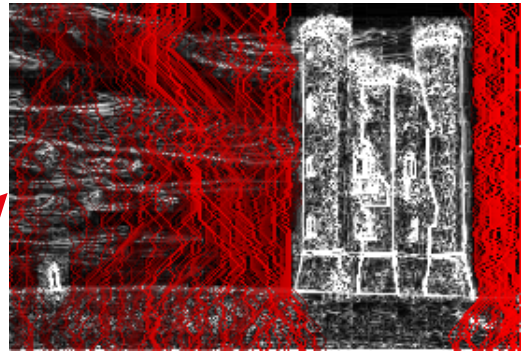
2. Introduction du projet

Principe du *seam carving*

Pour chaque pixel, calcul de l'information portée par rapport à ses voisins



Recherche de «lignes verticales » de basse information



Suppressions successives de ces lignes dans l'image originale

Au sommaire cette semaine

1. Bilan post-midterm
2. Introduction du projet
3. Les «dataclasses»

3. Les «dataclasses»

Motivation

- Avoir des variables, c'est parfois pas satisfaisant pour modéliser ce que l'on souhaite
- Ça serait intéressant de pouvoir définir ses propres «types» de variables
- Exemple avec le problème des 100 prisonniers :
  - «Ça serait bien d'avoir un type de variable `Prisoner`»
  - «Ça serait bien d'avoir un type de variable `Box`»
  - «Ça serait bien d'avoir un type de variable `Simulation`»

3. Les «dataclasses»

C’est quoi une «classe» ?

- Une classe est une (nouvelle) structure de données qui permet de modéliser des choses complexes en lui définissant plusieurs attributs (ou champs)
- Un objet est une instance d’une classe, autrement dit : un objet est une variable dont le type est une classe

3. Les «dataclasses»

Comment définir une classe et instancier un objet ?

```python
from dataclasses import dataclass   # <- Importation du module

# On ajoute ici un "décorateur", qui indique que ce qui suit est une dataclass
@dataclass
class Teacher:
    # Les trois lignes suivantes définissent des attributs
    first_name: str
    last_name: str
    office: str

# On instancie un objet grâce à une fonction particulière : "constructeur"
pw = Teacher(first_name="Patrick", last_name="Wang", office="1234")
```

3. Les «dataclasses»

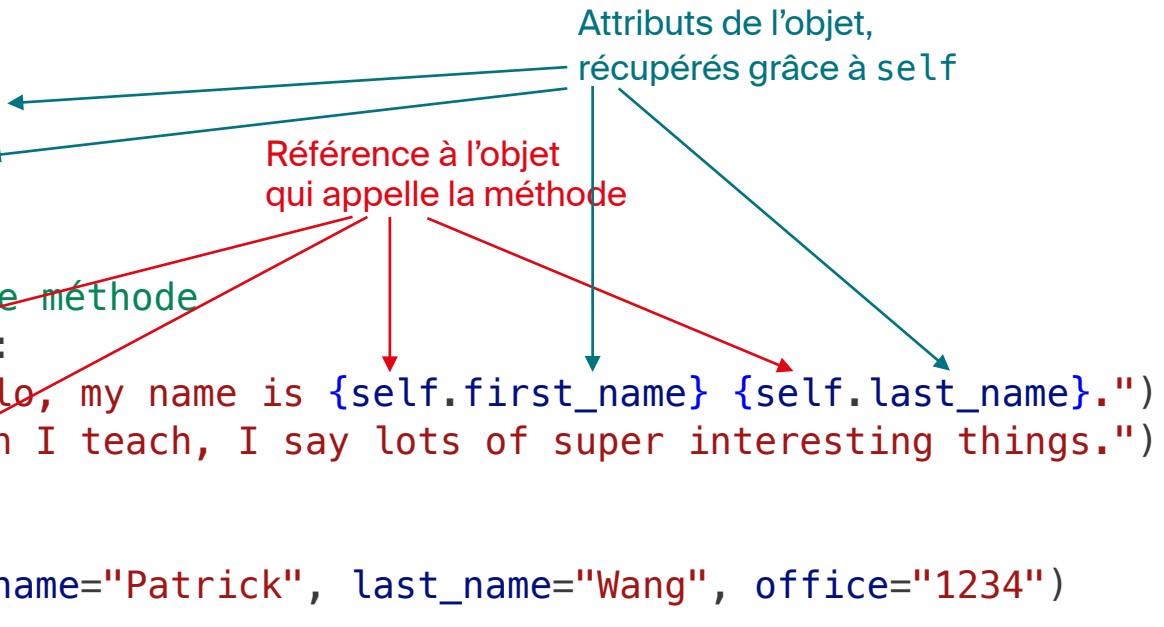
Définir des fonctions sur les classes : les méthodes

```python
from dataclasses import dataclass

@dataclass
class Teacher:
    first_name: str
    last_name: str
    office: str

    # On définit une méthode
    def teach(self):
        print(f"Hello, my name is {self.first_name} {self.last_name}.")
        print(f"When I teach, I say lots of super interesting things.")


pw = Teacher(first_name="Patrick", last_name="Wang", office="1234")
pw.teach()
```



3. Les «dataclasses»

Nouvelle modélisation du problème des 100 prisonniers

```python
@dataclass
class Box:
    number: int
    key_inside: int
```

```python
@dataclass
class Prisoner:
    number: int
    opened_boxes = List[Box]
    found_key: bool = False

    def look_for_key(self, boxes: List[Box]):
        ...

    def open_box(self, boxes: List[Box], box_number: int) -> int:
        ...
```

```python
@dataclass
class Simulation:
    prisoners: List[Prisoner]
    boxes: List[Box]

    def run(self):
        ...

    def is_successful(self) -> bool:
        ...
```

Bilan

- Découverte du concept de classe
- Une classe peut avoir des attributs et des méthodes
- Mise en œuvre des classes pour proposer une nouvelle modélisation du problème des 100 prisonniers
  - Ne pas hésiter à comparer les deux codes pour voir la différence