Classes

Friday, January 8, 2021 8:39 PM

A class is a template for creating objects with similar data
Capitalize the first letter in all the words in your class

```
class Bravais_Lattice:
```

Object stored in a class is called an "instance"

```
Cu = Bravais_Lattice()
```

Information about the instance is written as

```
Cu.lattice_parameter = 5
```

The information following the class is called a "field"
fields should not be capitalized and separated with snakecase

So why use a class instead of the variable types we've learned about so far?
Classes have methods, initialization, and help text!

We start with the init method (for initialization also called constructor)
We can add functions

```
class Bravais_Lattice:
    def __init__(self, crystal_structure, centering)
        self.crystal_strucure = structure
        self.centering = centering
        if structure == 'cubic':
            self.alpha=90
            self.beta=90
```

```
                self.gamma=90

Cu = Bravais_Lattice('cubic','face-centered')
```

Add help text with special string called a doc string
''' or """ to start and end

Consider how classes can work with other functions

```
def atoms_per_cell(self)
''' return atoms per cell'''
if self.crystal_structure == 'cubic' and self.centering == 'face-centered'
     atoms_per_cell = 4
     return atoms_per_cell


Cu = Bravais_Lattice('cubic','face-centered')
print(Bravais_Lattice.atoms_per_cell
```