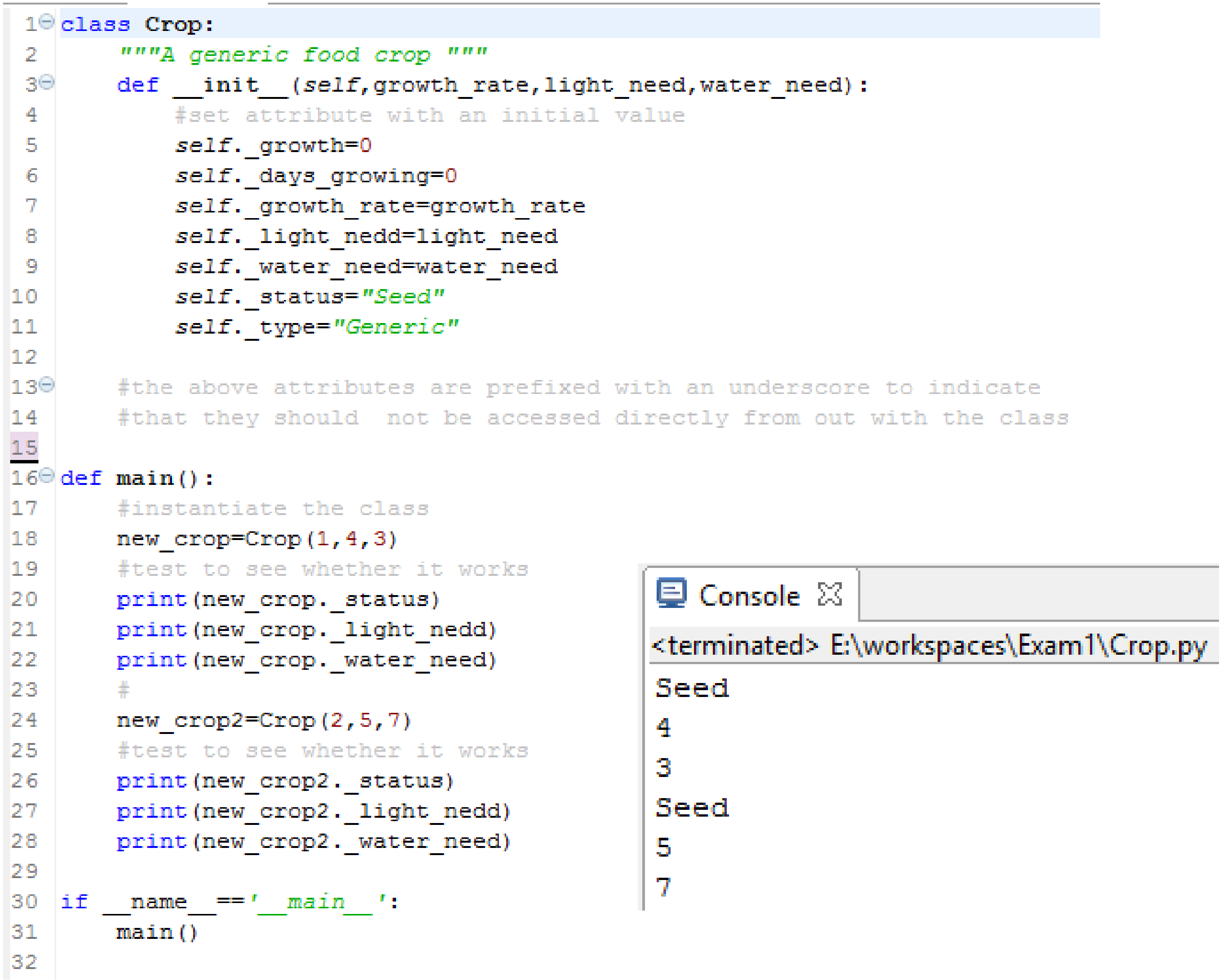
PYTHON

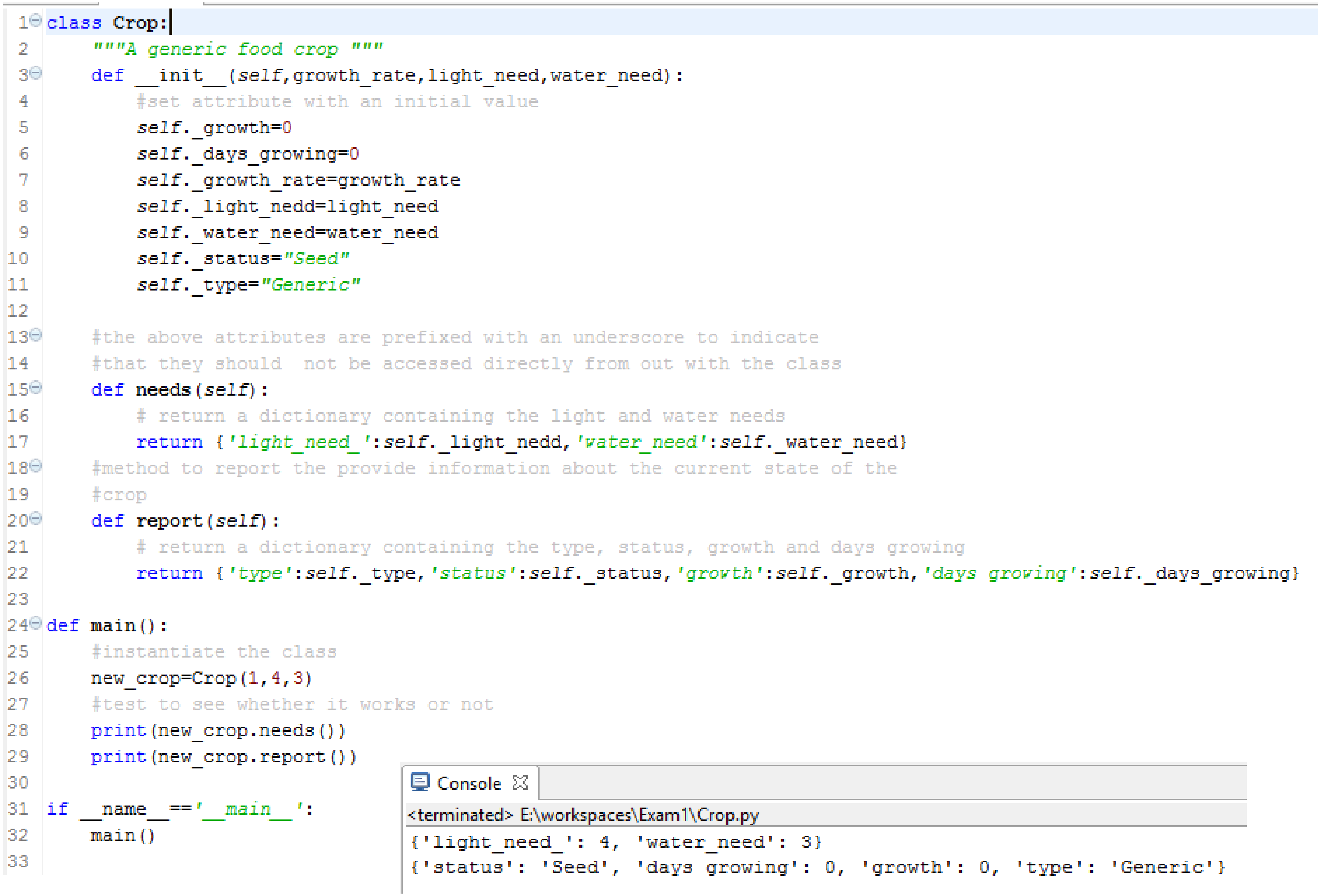
1. **CLASS**



```python
class Crop:
    """A generic food crop """
    def __init__(self,growth_rate,light_need,water_need):
        #set attribute with an initial value
        self._growth=0
        self._days_growing=0
        self._growth_rate=growth_rate
        self._light_nedd=light_need
        self._water_need=water_need
        self._status="Seed"
        self._type="Generic"

    #the above attributes are prefixed with an underscore to indicate
    #that they should  not be accessed directly from out with the class

def main():
    #instantiate the class
    new_crop=Crop(1,4,3)
    #test to see whether it works
    print(new_crop._status)
    print(new_crop._light_nedd)
    print(new_crop._water_need)
    #
    new_crop2=Crop(2,5,7)
    #test to see whether it works
    print(new_crop2._status)
    print(new_crop2._light_nedd)
    print(new_crop2._water_need)

if __name__=='__main__':
    main()

```

```
<terminated> E:\workspaces\Exam1\Crop.py
Seed
4
3
Seed
5
7
```

2. METHOD



```
class Crop:
    """A generic food crop """
    def __init__(self,growth_rate,light_need,water_need):
        #set attribute with an initial value
        self._growth=0
        self._days_growing=0
        self._growth_rate=growth_rate
        self._light_nedd=light_need
        self._water_need=water_need
        self._status="Seed"
        self._type="Generic"

    #the above attributes are prefixed with an underscore to indicate
    #that they should  not be accessed directly from out with the class
    def needs(self):
        # return a dictionary containing the light and water needs
        return {'light_need_':self._light_nedd,'water_need':self._water_need}
    #method to report the provide information about the current state of the
    #crop
    def report(self):
        # return a dictionary containing the type, status, growth and days growing
        return {'type':self._type,'status':self._status,'growth':self._growth,'days growing':self._days_growing}

def main():
    #instantiate the class
    new_crop=Crop(1,4,3)
    #test to see whether it works or not
    print(new_crop.needs())
    print(new_crop.report())

if __name__=='__main__':
    main()
```

```
<terminated> E:\workspaces\Exam1\Crop.py
{'light_need_': 4, 'water_need': 3}
{'status': 'Seed', 'days growing': 0, 'growth': 0, 'type': 'Generic'}
```