Solutions

Each answer shows the steps, then the **result**.

2.1

- the class is the design (the car model): what every car of that model has and can do
- an object is one real car built from it, with its own state, such as its own colour and mileage

2.2

- attributes: `colour`, `brightness`
- methods: e.g. `switch_on`, `switch_off`, `dim`

2.3

```
def __init__(self, colour):
    self.colour = colour
    self.brightness = 0
```

- `__init__` with `self` and a `colour` parameter
- both attributes set through `self`

2.4

- `a.clicks` **0**: reset after one click
- `b.clicks` **2**
- `Counter.total_made` **2**: two objects were made

3.1 B

- objects are made from a class

3.2 B

- `p` and `q` refer to the same object

3.3 C

- one value shared by the class, not one per student

3.4

(a)

- `name`, `height`, `watered`

(b)

- `fern = Plant("fern", 12)`

(c)

- `fern` watered, then grows 3: 15
- the second `fern.grow(4)` does nothing: `fern` **15**
- `cactus` was never watered: **5**

(d)

- `grow` changes the height only when `watered` is `True`; `cactus` was never watered
- the first `grow` sets `watered` back to `False`, so the second call does nothing

(e)

```
def needs_water(self):
    return not self.watered
```

- a method with `self`
- returns the opposite of `watered`