1.

```
class Vehicle:
    pass
```

1.

```
class Vehicle:
    def __init__(self, max_speed, colour):
        self.max_speed = max_speed
        self.colour = colour

vehicle = Vehicle(150, 'red')
print(vehicle.max_speed, vehicle.colour)
```

1.

```
class Vehicle:
    def __init__(self, max_speed, colour):
        self.max_speed = max_speed
        self.colour = colour

    def update_max_speed(self, value):
        self.max_speed += value

    def update_colour(self, new_colour):
        self.colour = new_colour

vehicle = Vehicle(150, 'red')
print(vehicle.max_speed, vehicle.colour)

vehicle.update_max_speed(50)
vehicle.update_colour('blue')
print(vehicle.max_speed, vehicle.colour)
```

1.

~~~
```py
class Bus(Vehicle):
    pass

bus = Bus()
print(bus.max_speed, bus.colour)
```
~~~

1.

~~~
```py
bus = Bus()
print(type(bus))

# <class '__main__.Bus'>
```
~~~

1.

~~~
```py
bus = Bus()
print(isinstance(bus, Vehicle))

# True
```
~~~

1.

~~~
```py
class Bus(Vehicle):
    def __init__(self, max_speed, colour, seating_capacity):
        super().__init__(max_speed, colour)
        self.seating_capacity = seating_capacity

    def calculate_ticket_price(self):
        initial_price = self.seating_capacity * 0.05
        return initial_price + (10 / 100)

bus = Bus(40)
print(bus.calculate_ticket_price())
```
~~~

1.

~~~
```py
class Bus(Vehicle):
    def __init__(self, max_speed, colour, seating_capacity):
        super().__init__(max_speed, colour)
        self.seating_capacity = seating_capacity

    def calculate_ticket_price(self):
        initial_price = self.seating_capacity * 0.05
        return initial_price * 1.1
~~~

~~~
    def __repr__(self):
        return repr(f'Max speed: {self.max_speed}, Colour: {self.colour}, Seating capacity: {self.seating_capacity}')

bus = Bus(120, 'white', 40)
print(repr(bus))

# 'Max speed: 120, Colour: white, Seating capacity: 40'
```
~~~