```
from abc import ABC,abstractmethod
class vehicle(ABC):
    @abstractmethod
    def start(self):
        pass
    @abstractmethod
    def stop(self):
        pass
class car(vehicle):
    def start(self):
        print("Ignition start")
    def stop(self):
        print("Ignition Stop")
class bike(vehicle):
    def start(self):
        print("Kick Start")
    def stop(self):
        print("Switch engine off")
class scooty(vehicle):
    def start(self):
        print("Self start")
    def stop(self):
        print("Switch off engine")
c1=car()
b1=bike()
s1=scooty()
c1.start()
c1.stop()
b1.start()
b1.stop()
s1.start()
s1.stop()
```

Output:

```
Ignition start
Ignition Stop
Kick Start
Switch engine off
Self start
Switch off engine
```

2)now lets try creating object of class vehicle we should’nt get output but an error saying we cant create  objects of abstract class

```
v1=vehicle()
v1.start()
v1.stop()
```

Output:

```
TypeError                              Traceback (most recent call last)
Cell In[3], line 1
----> 1 v1=vehicle()
      2 v1.start()
      3 v1.stop()

TypeError: Can't instantiate abstract class vehicle with abstract methods start, stop
```

```
In [ ]:
```

3)lets try intitializing while removing unique start function in car class
We will get error since class car will inherit abstract method of vehicle,since car is a child of vehicle,.

```
from abc import ABC,abstractmethod
class vehicle(ABC):
    @abstractmethod
    def start(self):
        pass
    @abstractmethod
    def stop(self):
        pass
class car(vehicle):

    def stop(self):
        print("Ignition Stop")
class bike(vehicle):
    def start(self):
        print("Kick Start")
    def stop(self):
        print("Switch engine off")
class scooty(vehicle):
    def start(self):
        print("Self start")
    def stop(self):
        print("Switch off engine")
c1=car()
b1=bike()
s1=scooty()
```

```
c1.start()
c1.stop()
b1.start()
b1.stop()
s1.start()
s1.stop()
```

Output:

```
TypeError                                 Traceback (most recent call last)
Cell In[7], line 23
     21     def stop(self):
     22         print("Switch off engine")
---> 23 c1=car()
     24 b1=bike()
     25 s1=scooty()

TypeError: Can't instantiate abstract class car with abstract method start
```