ABSTRACTION

Vehicle

no-of-tyres; X
start(); X

---

class Vehicle
{
        int no-of-tyres;
abstract void start();
}

Car

no-of-tyres = 4
start()
  2 Sop("starts with key");
  3

Scooter

no-of-tyres = 2
start()
  2
  3 Sop("starts with kick");

1. A method without body (no implementation) is known as abstract method.

Vehicle

no-of-tyres; X
start(); X

---

abstract class Vehicle
{
        int no-of-tyres;
①   abstract void start();
}

Car

no-of-tyres = 4
start()
  2 Sop("starts with key");
  3

Scooter

no-of-tyres = 2
start()
  2
  3 sop("starts with kick");

2. A method must always be declared in an abstract class, or we can say that if a class has an abstract method, it should be declared abstract as well.

www.smartprogramming.in

Vehicle | Car | Scooter

no-of-tyres; X
start(); X

abstract class Vehicle
{
  int no-of-tyres;
  abstract void start();
}

no-of-tyres = 4
start()
{
  SOP("starts with key");
}

no-of-tyres = 2
start()
{
  SOP("starts with kick");
}

* class Car extends Vehicle
{
  void start()
  {
    SOP("starts with key");
  }
}

3. If a regular class extends an abstract class, then the class must have to implement all the abstract methods of abstract parent class or it has to be declared abstract as well.

11:58 / 22:25

---

www.smartprogramming.in

Vehicle | Car | Scooter

no-of-tyres; X
start();

abstract class Vehicle
{
  int no-of-tyres;
  abstract void start();
}

no-of-tyres = 4
start()
{
  SOP("starts with key");
}

no-of-tyres = 2
start()
{
  SOP("starts with kick");
}

* class Car extends Vehicle
{
  void start()
  {
    SOP("starts with key");
  }
}

* class Scooter extends Vehicle
{
  void start()
  {
    SOP("starts with kick");
  }
}

4. Abstract methods in an abstract class are meant to be overridden in derived concrete classes otherwise compile-time error will be thrown.

Vehicle
└ no_of_tyres; X
  start(); X

abstract class Vehicle → object creat X
{
  ✓ int no_of_tyres;
  ① abstract void start();
}

Car
no_of_tyres = 4
└ start()
  2 sop("starts with key");
  3

* class Car extends Vehicle
{
  void start()
  {
    sop("starts with key");
  }
}

Scooter
no_of_tyres = 2
└ start()
  sop("starts with kick");
  3

* class Scooter extends Vehicle
{
  void start()
  {
    sop("starts with kick");
  }
}

5. Abstract classes cannot be instantiated, means we can't create an object of Abstract class.

iF We able to create an object , then we are also able to call the method , but the method doesn't have body then what's the use of that.

But we can make reference of that.