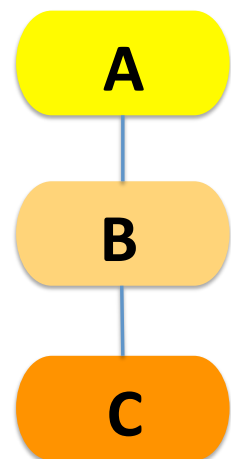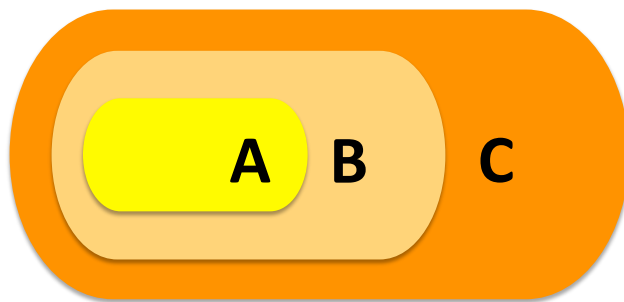# super

```java
public class Vehicle{
  private String color;
  public Vehicle(String c) {color=c;}
  public void moveForward(){…}
  public void moveBackward(){…}
}
public class Car extends Vehicle{
  private int noPass;
  public Car(int n, String c){…}
  public void enter(int n){…}
  public void exit(int n){…}
}
```
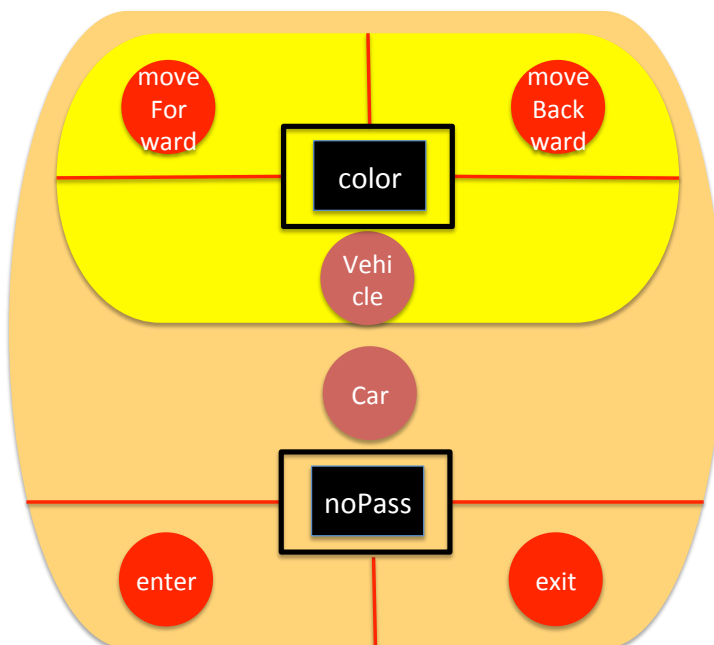
Source: Pixabay

Source: Pixabay

```java
public class Vehicle{
  private String color;
  public Vehicle(String c) {color=c;}
  public void moveForward(){…}
  public void moveBackward(){…}
}
public class Car extends Vehicle{
  private int noPass;
  public Car(int n, String c)
    {color=c; noPass=n;}
  public void enter(int n){…}
  public void exit(int n){…}
}
```


Source: Pixabay

```java
public class Vehicle{
  private String color;
  public Vehicle(String c) {color=c;}
  public void moveForward(){…}
  public void moveBackward(){…}
}
public class Car extends Vehicle{
  private int noPass;
  public Car(int n, String c)
    {super(c); noPass=n;}
  public void enter(int n){…}
  public void exit(int n){…}
}
```

Source: Pixabay

```java
public class Vehicle{
  private String color;
  public Vehicle(String c) {color=c;}
  public void moveForward(){…}
  public void moveBackward(){…}
}
public class Car extends Vehicle{
  private int noPass;
  public Car(int n, String c)
   {          noPass=n; super(c);}
  public void enter(int n){…}
  public void exit(int n){…}
}
```

Source: Pixabay

```java
public class A {
  public A() {System.out.print("A, ");}
}

public class B extends A {
  public B() {super(); System.out.print("B, ");}
}

public class C extends B {
  public C() {super(); System.out.print("C.");}
}

        C c = new C();
```

A, B, C.

A
B
C

Summary

- 1st thing to do: call superclass constructor
- Cannot be called later