创建于：2020/03/23

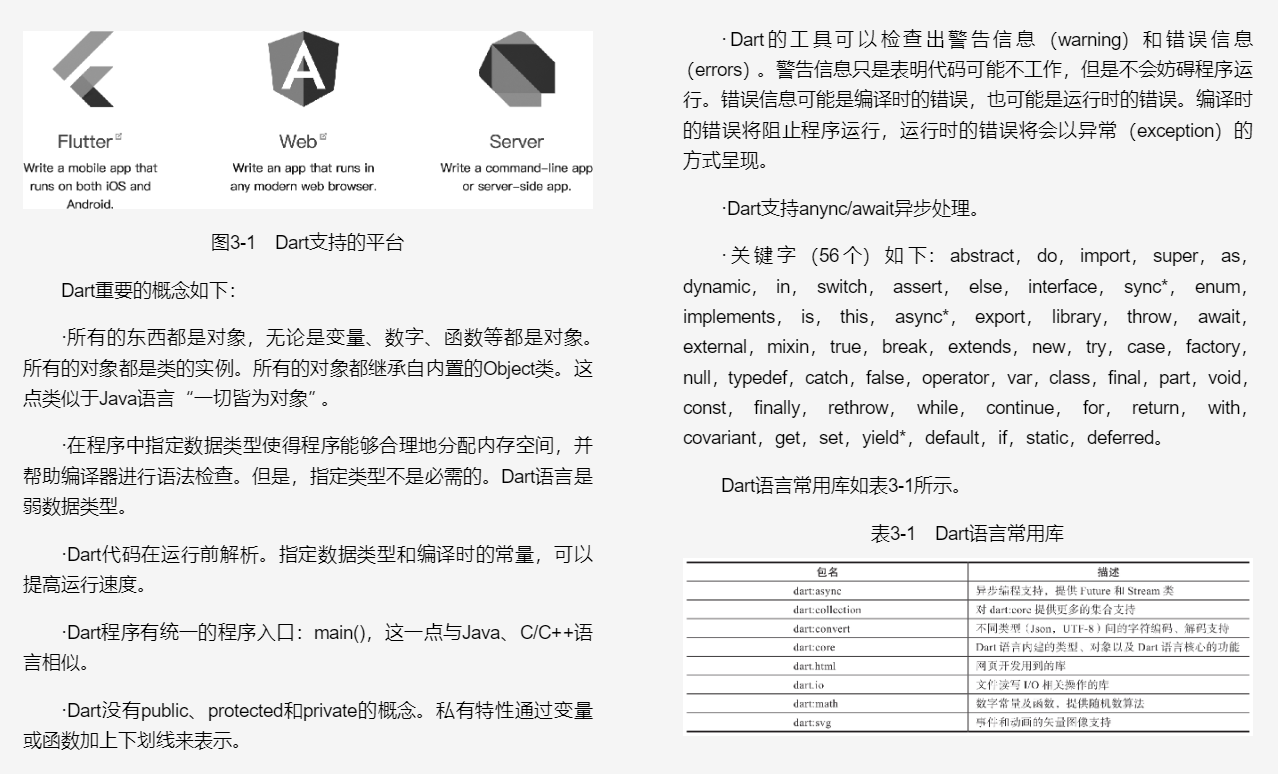
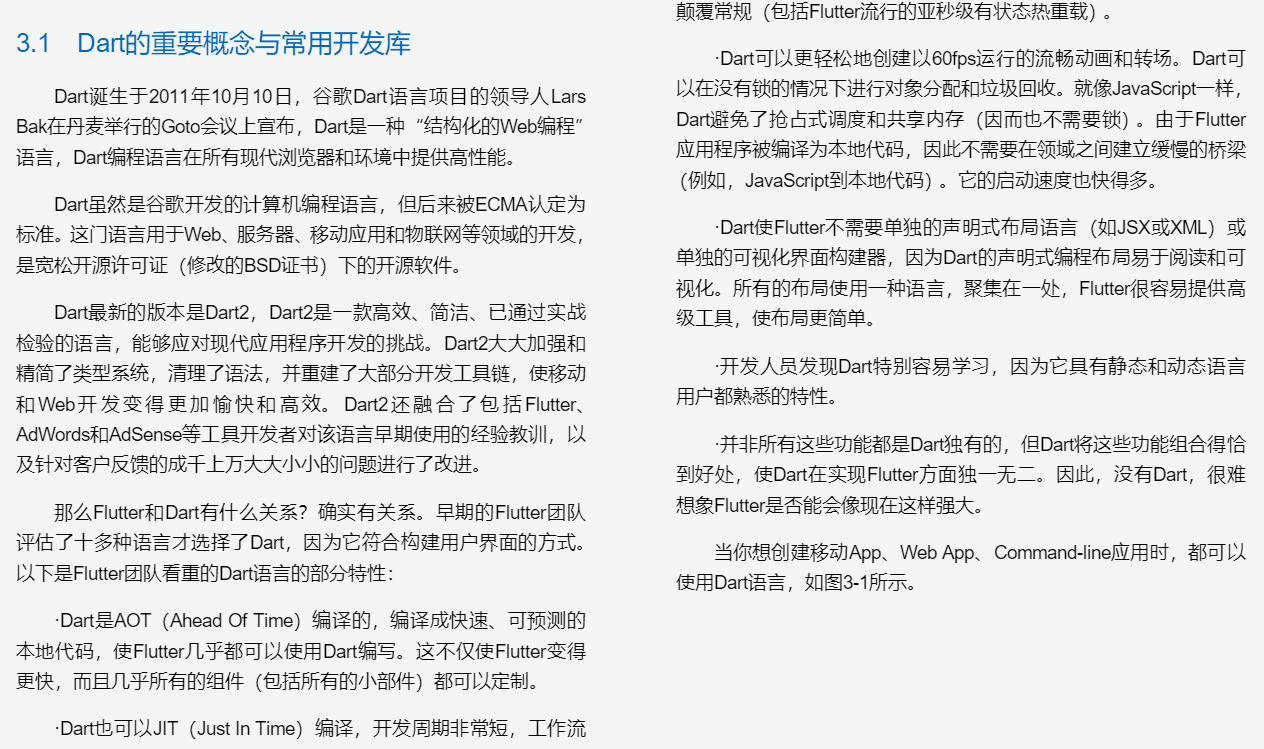
学习文档记录，较啰嗦

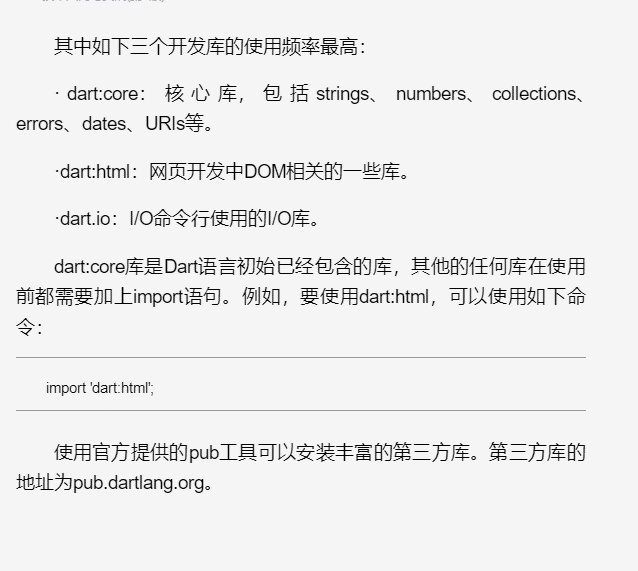
# 参考资料

1.Dart语法：<https://www.jianshu.com/p/9e5f4c81cc7d>

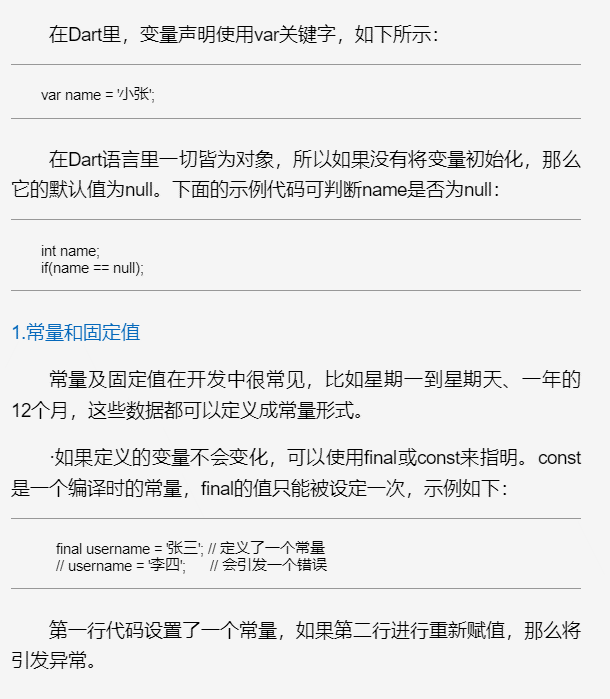
# dart概述

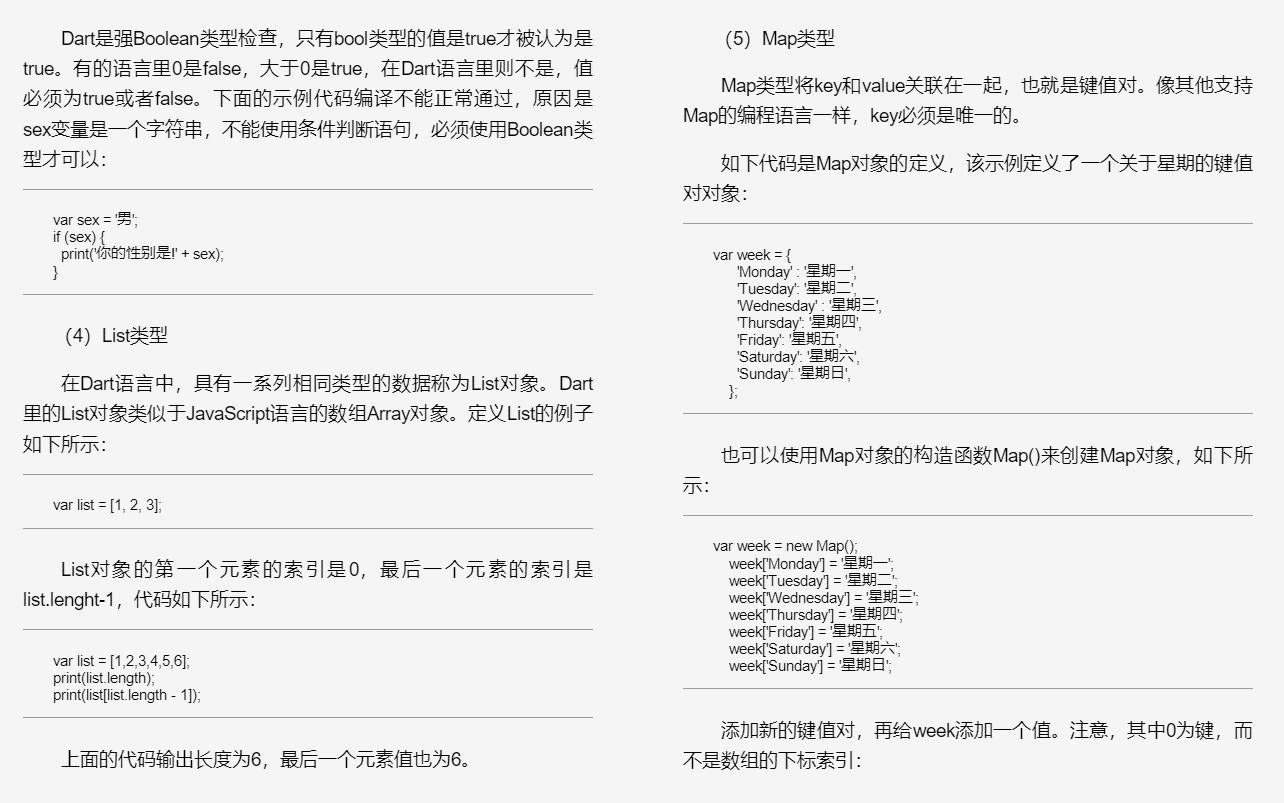
dart是一个面向对象的语言。



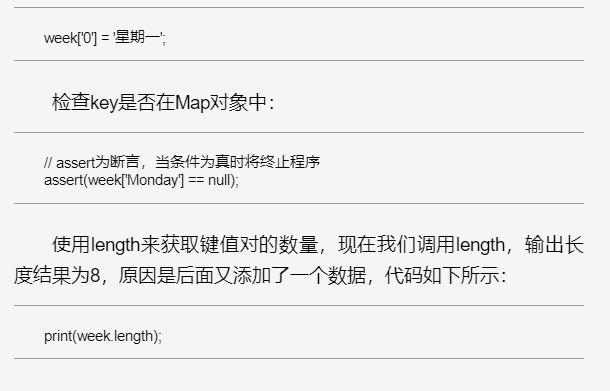


# 变量与基本数据类型









# 函数

函数也是对象，属于Function对象。

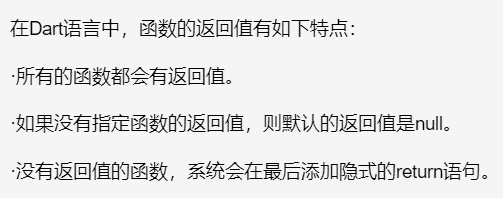
main()是dart程序的入口函数。

可选参数：{}或[]都可以。{}可选的命名参数，[]可选的位置参数。

void printInfo(String name, int age, {String form = '火星'})

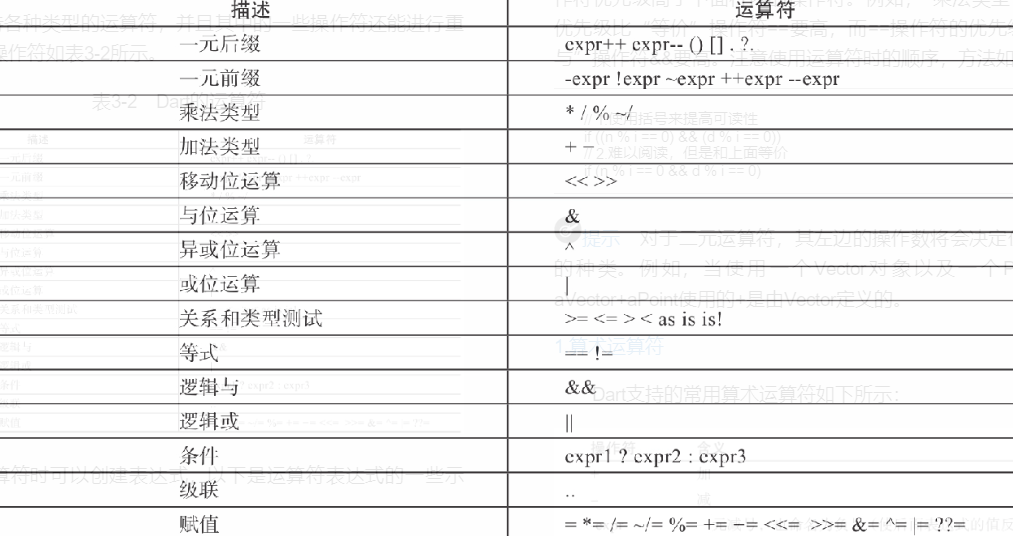
void printInfo(String name, int age, [String form = '火星'])

函数返回值：

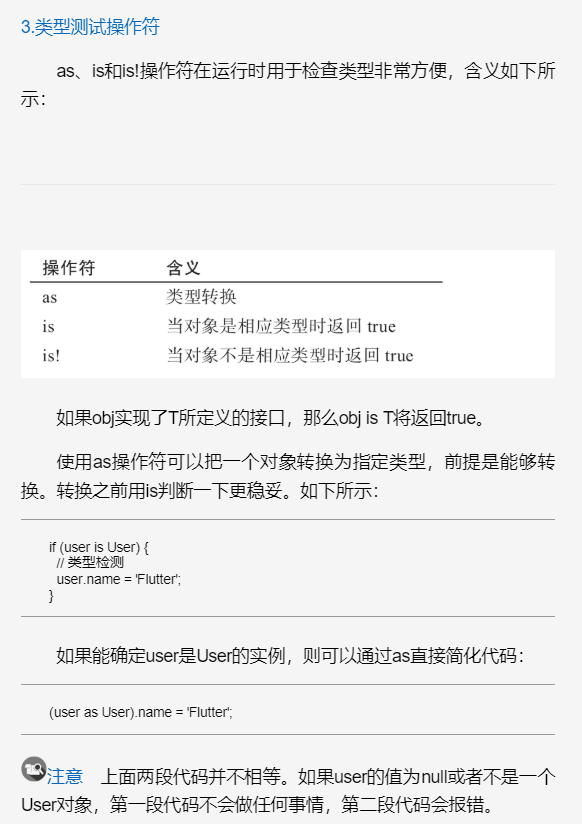


# 运算符

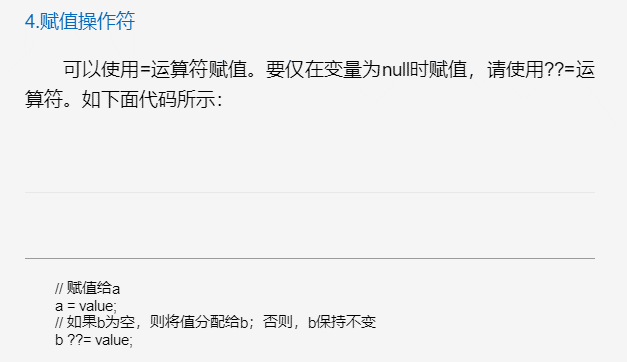
所有的运算符：



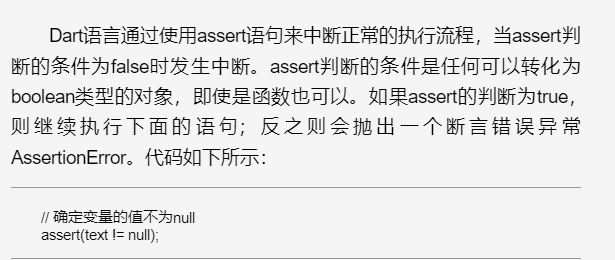
类型测试操作符：



赋值运算符：=,??=



# 断言



# 六、面向对象

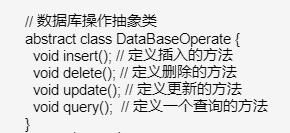
Dart作为高级语言，支持面向对象的很多特性，并支持基于mixin的继承方式。基于mixin的继承方式是指一个类可以继承自多个父类，相当于其他语言里的多继承。

类中所有的变量都会隐式的定义setter方法，针对非的变量会额外添加getter方法。

普通类：

class Test {  
 int age;  
 String name;  
  
 //普通构造函数  
 Test(this.age, this.name);  
  
 //命名构造函数  
 Test.age(this.age, {this.name});  
}

抽象类：只定义好方法，不实现具体方法，具体实现由调用者完成。使用abstract关键字。



使用?.来确认前操作数不为空, 常用来替代. , 避免左边操作数为null引发异常。





单继承：extends

class TestA extends TestB {}

多继承：with

class TestA extends TestB with TestC {}

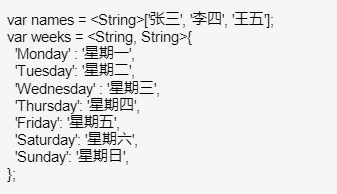
# 七、枚举类

枚举类型是一种特殊的类，通常用来表示相同类型的一组常量。

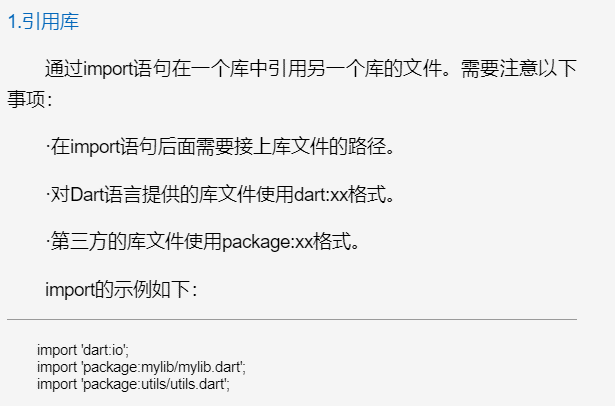
enum Type { student, teacher }  
print(Type.values);//枚举类中所有的值

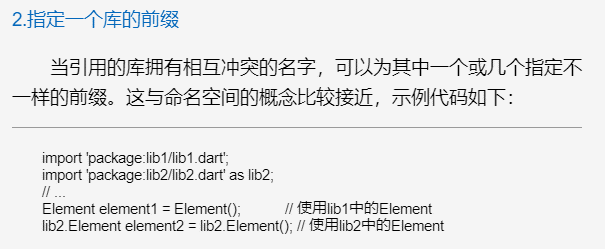
# 八、泛型

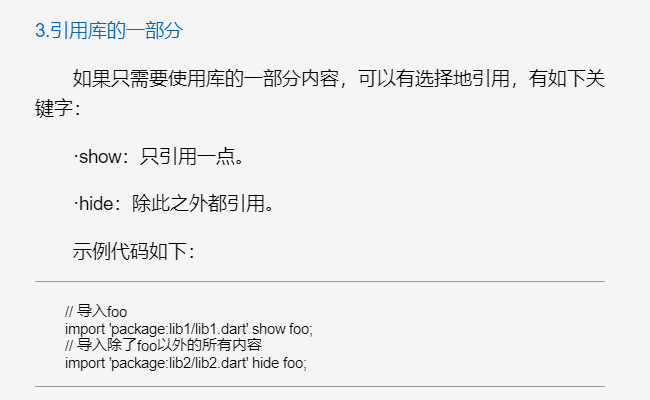
泛型通常是为了类型安全而设计，适当的指定泛型类型会生成更好的代码，可以减少代码重复。使用<T>的方式实现泛型。



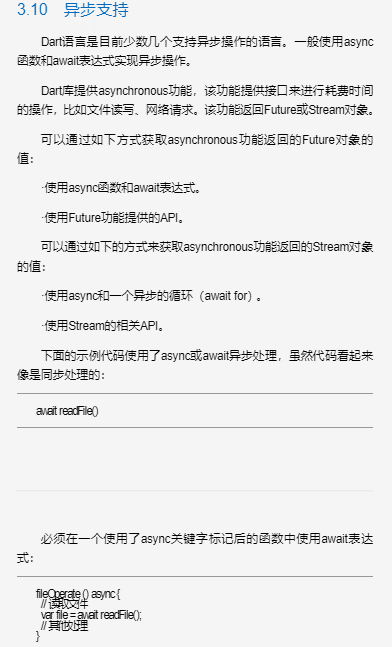
# 九、库的使用







# 十、异步



# 十一、元数据

