Data Types

1. Basic data types
   1. String “”, ‘’
   2. bool
   3. num
      1. int
      2. double

* 모두 객체. 우클릭 go to type definition으로 class 확인가능

1. Lists / 대괄호 사용
   1. 사용법
      1. var numbers = [1, 2, 3, 4];
      2. List<int> numbers = [1, 2, 3, 4];
   2. collection if, collection for 사용가능
      1. collection if, : 존재할 수도 아닐 수도 있는 요소 추가 가능

ex> [1, 2, 3, 4, if(true) 5]

* 1. 마지막 요소 뒤에 , 붙여주면 세로로 정렬 가능

1. String Interpolation
   1. text에 변수를 추가하는 방법

ex> 작따, 큰따 상관 없음

var name = ‘경희’;

var age = 20;

var greeting = ‘내 이름은 $name 입니다. 나이는 ${age+1}입니다.’; => 모두 text로 만들어준당

* 주의점, 안에 I’m 같이 따옴포 쓸 땐 앞에 escape 기호 써주기

1. Collection For
   1. 사용법

var oldFriends = [‘nico’, ‘lynn’,];

var newFriends = [‘lewis’, ‘ralph’, ‘darren’,

for(var friend in oldFriends) “★ $friend”, ];

1. Maps : JS,TS의 object와 같은 것 / 중괄호 사용
   1. key : value로 이루어짐
      1. key와 value는 적는 것에 따라 컴파일러가 type를 정해줌.

var students = {key : value};

* + 1. key와 value type 지정 가능

Map<int, bool> student ={ key : value };

type은 List일 수도 있다.(복잡한 것들…)

반대로 List의 타입이 Map일 수도

그러나 이런 경우엔 class를 더 많이 쓴다. (추후 다룸)

1. Sets : 모든 아이템이 unique / 중괄호 사용 <> List
   1. 중괄호/ 대괄호 사용으로 Set / List 구분 가능
      1. var numbers = { } / var numbers = [ ];
   2. 혹은 type 지정 가능
      1. Set<int> numbers = {}

Functions

void(return type) sayHello(String name){

print(“Hello $name nice to meet you“)

}

String sayHello(String name){

return “Hello $name nice to meet you“ ;

}

* 바로 return만 하는 함수의 경우 화살표 함수 가능
* fat arrow syntaz

String sayHello(String name) => “Hello $name nice to meet you“ ;

num plus(num a, num b) => a+b;

Named Parameters

String sayHello(String name, int age, String country){

return “Helloi $name, you are $age, and you are from $country”

}

sayHello(‘nico’, 19, ‘cuba’);

파라미터 순서 기억하기 어려움

순서 관계 없이 쓰는 법

파라미터 중괄호로 쌈.

String sayHello({String name, int age, String country}){

return “Helloi $name, you are $age, and you are from $country”

}

sayHello(name : ‘nico’, age : 19, country : ‘cuba’);

>파라미터가 null이라면?