

Data Types

- Dart는 완전히 객체 지향 언어이다.
- 모든 것은 객체로 이루어져 있음
 - 심지어 함수까지

```
void main() {  
  String name = "nico";  
  bool alive = true;  
  int age = 12;  
  double money = 69.99;  
  
  num x = 12;  
  x = 1.1  
  // num은 int 일 수도 double 일 수도 있다.  
  // num class는 int와 double의 부모 class  
}
```

List

```
void main() {  
  var numbers = [1,2,3,4];  
}
```

- dart에서 list가 멋진 점은
 - collection if와 collection for을 지원한다는 것

collection if

```
void main() {  
  var giveMeFive = true;  
  var numbers = [  
    1,  
    2,  
    3,  
    4,  
    if (giveMeFive) 5  
  ];  
}
```

```
// 위 코드와 같은 의미
void main() {
    var giveMeFive = true;
    var numbers = [
        1,
        2,
        3,
        4,
    ];
    if (giveMeFive) {
        numbers.add(5);
    }
}
```

String interpolation

- 변수 사이에 변수를 넣는 것
- \$와 변수이름을 적어주면 됨
- 값을 계산하려면 {}를 통해 입력

```
void main() {
    var name = 'nico';
    var greeting = 'Hello everyone, my name is $name, nice to meet you!'
}
```

```
void main() {
    var name = 'nico';
    var age = 10;
    var greeting = "Hello everyone, my name is $name, I'm ${age + 2}!"
}
```

collection for

- collection if 처럼 list안에서 조건문을 수행
- 아래 코드는 string interpolation도 활용했다

```
void main() {
    var oldFriends = ['nico', 'lynn'];
    var newFriends = [
        'lewis',
        'ralph',
        'darren',
        for (var friend in oldFriends) "★ $friend",
    ];
}
```

```
print(newFriends);
}

//[lewis, ralph, darren, ★ nico, ★ lynn]
```

Maps

- 파이썬의 dictionary와 같은 것

```
void main() {
    var player = {
        'name' : 'nico',
        'xp' : 19.99,
        'superpower' : false,
    };
}
```

- 위의 방식으로 player를 선언하면
- 컴파일러가 <String , object> 이렇게 type을 인식함

```
void main() {
    Map<int, bool> player = {
        1: true,
        2: false,
        3: true,
    };
}

void main() {
    Map<List<int>, bool> player = {
        [1, 2, 3, 5]: true,
    };
}
// 위와 같은 복잡한 방식으로도 명시 가능
```

- 위의 방식으로 선언하면
- key와 value의 타입이 명시한 타입이 아니라면 오류가 난다.

```
void main() {
    List<Map<String, Object>> players = [
        {'name': 'jason', 'age': 12},
        {'name': 'jason', 'age': 12},
    ];
}
// 리스트 안에 map을 넣은 형태
```

Set

- set 안의 요소들은 유니크함
- 파이썬의 tuple과 같다

```
void main() {  
    var numbers = {1, 2, 3, 4}; // = Set<int> numbers = {1, 2, 3, 4};  
    numbers.add(1);  
    numbers.add(1);  
    numbers.add(1);  
    numbers.add(1);  
    print(numbers);  
}  
  
// {1, 2, 3, 4}
```