Dynamic: 여러가지 타입을 가질 수 있는 변수에 쓰는 키워드

ex) var name; : 변수를 아무것도 지정해주지 않고 선언했을 때

이 때 변수가 dynamic이 되는데

name = ‘hello’;

name = 12;

name = true;

위 같이 자신이 원하는 값을 넣을 수 있다.

🡪 변수가 어떤 타입인지 알기 어려울 때 사용한다.

🡪 dynamic name; 처럼 변수 var이라고 적지 않고 dynamic이라고 명시해줘도 된다.

🡪 조건문으로 데이터타입을 지정할 수도 있다.

ex) if (name is string) {

name.

}

여기서, name.이 string이라는 데이터가 비어 있는지 또는 길이, hashcode, compare, contains 등 모든 걸 알 수 있다.

🡪 if문 밖으로 나오면 가시 타입을 확인해줘야 한다 – (Q. 명시적 자료형 변환과 같은 효과인가?)

null safrty : 개발자가 null 값을 참조할 수 없도록 하는 것 🡪 널 값을 참조하면 런타임 에러가 뜨기 때문(사용자 기기에서 발생할 에러, 컴파일러가 못 잡는 에러를 방지) (NoSuchMethodError)

그렇다고 언어에서 null을 삭제할 수 없다. Null은 부재를 의미하기 때문(채워지지 않은 값)

🡪 어떤 변수가 널이 될 수 있는지 표시해줘야 한다. Dart 변수는 기본적으로 nullable이 아님

데이터 타입일 수도 널 일 수도 있다는 것을 변수에 ?를 붙여 변수 혹은 데이터가 널이 될 수 도 있음을 명시해주는 것이다.

🡪 그 변수를 사용하기 전에 확인해줘야 한다.

Void main() {

String? nico = ‘nico’;

nico = null;

if (nico != null) {

nico.isNotEmpty;

}

}

위와 같이 확인할 수 있지만 if를 지우고 단축문법으로 nico?. isNotEmpty;를 사용할 수 있다.

그렇게 값이 존재하는지 확인하고 이후에 연산을 진행하면 된다.

Nico가 널이 아닐 때만 isNotEmpty 속성을 줄 것임.