

02

Mixin

Dart

멤버를 공유하는 믹스인

믹스인 알아보기

- 믹스인은 mixin이라는 예약어로 선언

```
• 믹스인 선언

mixin MyMixin {
  int data1 = 10;
  void myFun1() {
    print('MyMixin... myFun()...');
  }
}
```

- 믹스인에는 변수와 함수를 선언할 수 있지만 클래스가 아니므로 생성자는 선언할 수 없습니다.

```
• 믹스인에 생성자 선언 오류

mixin MyMixin {
  int data1 = 10;
  static int data2 = 20;

  MyMixin() {} // 오류

  void myFun1() {
    print('MyMixin... myFun()...');
  }
  static void myFun2() {}
}
```

멤버를 공유하는 믹스인

믹스인 알아보기

- 믹스인은 생성자를 가질 수 없으므로 객체를 생성할 수도 없습니다.

• 믹스인 객체 생성 오류

```
mixin MyMixin {  
  int data1 = 10;  
  void myFun1() {  
    print('MyMixin... myFun()...');  
  }  
}  
  
main() {  
  var obj = MyMixin() ; // 오류  
}
```

멤버를 공유하는 믹스인

믹스인 사용하기

- 믹스인에 선언한 멤버를 다른 클래스에서 사용할 수 있습니다. 이때 with 예약어를 사용합니다.

```
• 믹스인 멤버 이용

mixin MyMixin {
  int mixinData = 10;
  void mixInFun() {
    print('MyMixin... mixInFun()...');
  }
}

class MyClass with MyMixin {
  void sayHello() {
    print('data : $mixinData');
    mixInFun();
  }
}
```

멤버를 공유하는 믹스인

믹스인 사용하기

• 믹스인을 다중 상속처럼 활용한 예

```
mixin MyMixin {  
  int mixinData = 10;  
  void mixInFun() {  
    print('MyMixin... mixInFun()...');  
  }  
}  
  
class MySuper {  
  int superData = 20;  
  void superFun() {  
    print('MySuper... superFun()');  
  }  
}  
  
class MyClass extends MySuper with MyMixin {  
  void sayHello() {  
    print('class data : $superData, mixin data : $mixinData');  
    mixInFun();  
    superFun();  
  }  
}
```

```
main() {  
  var obj = MyClass();  
  obj.sayHello();  
}
```

▶ 실행 결과

```
class data : 20, mixin data : 10  
MyMixin... mixInFun()...  
MySuper... superFun()
```

멤버를 공유하는 믹스인

믹스인 사용하기

- with 예약어로 믹스인을 이용하는 클래스의 객체는 믹스인 타입으로 사용할 수 있습니다.

• 믹스인 타입 객체 선언

```
class MyClass with MyMixin {}

main() {
  var obj = MyClass();

  if (obj is MyMixin) {
    print('obj is MyMixin');
  } else {
    print('obj is not MyMixin');
  }

  MyMixin obj2 = MyClass();
}
```

▶ 실행 결과

obj is MyMixin

멤버를 공유하는 믹스인

믹스인 사용 제약

- 특정 타입의 클래스에서만 사용하도록 제한
- 스인 선언부에 on 예약어로 해당 타입을 지정

• 믹스인 타입 제한

```
mixin MyMixin on MySuper {  
  }  
  
class MySuper {  
  }  
  
class MyClass extends MySuper with MyMixin { // 성공  
  }  
  
class MySomeClass with MyMixin { // 오류  
  }
```

멤버를 공유하는 믹스인

with로 클래스 이용하기

- 클래스도 with로 이용

```
• with로 클래스 이용하기

class SomeClass {
  int someData = 10;
}

class MyClass with SomeClass {
  void sayHello() {
    print('someData : $someData');
  }
}
```


멤버를 공유하는 믹스인

with로 클래스 이용하기

- with 예약어로 사용할 클래스에는 생성자를 선언할 수 없습니다.

• 생성자가 있는 클래스를 with로 사용하면 오류

```
class User {  
  User() {}  
}  
  
class MyClass with User { // 오류  
}
```



감사합니다

단단히 마음먹고 떠난 사람은
산꼭대기에 도착할 수 있다.
산은 올라가는 사람에게만 정복된다.



윌리엄 셰익스피어
William Shakespeare