

⇒ Implement 2 more API's from fakestore.

⇒ ResponseEntity

↳ HTTP Status Code
↳ Headers
↳ Response Body

Generics in JAVA.

JAVAS.

Generic was introduced in Java 5

In generics, Whenever you define a datatype, int, bool, string, you must define this placeholder

List < Integer >

List < String >

List < Animal >

List < Student >

⋮

Class Pair < T, U > {

Placeholder

This String is replaced by T

T first;

U second;

T getfirst() {
return this.first;

}

U getsecond() {
return this.second;

}

setfirst (T first) {
this.first = first;

}

}

If you place class Pair as <T, U> and use that class somewhere else with define datatype, so this String will be replaced by T everywhere you want to use String.

Pair < String, Integer > p = _____;



before JAVAS.

Object

List < Object > list = _____

list.add(student)

list.add(car)

No compile time

safety

with objects no compile safety

Before JAVAS

Object-

After JAVAS.

List < Animal >

List < Dog >

} Compile
Time
Safety-

List < Student > list = _____ ;

list.add(car)

⇒ Backward Compatible.

⇒ If a new feature is being launched, the earlier should keep working.

⇒ At run time, every data type will be Object only.

⇒ Type Erasure.

Once the application has been compiled then all the datatype will be object at runtime. The scope of generics datatype remains only till compile time.

This property is called type erasure since the datatype will be erased at runtime

