



TOUR OF SCALA

TUPLES

In Scala, a tuple is a value that contains a fixed number of elements, each with its own type. Tuples are immutable.

Tuples are especially handy for returning multiple values from a method.

A tuple with two elements can be created as follows:

```
val ingredient = ("Sugar" , 25)
```

This creates a tuple containing a `String` element and an `Int` element.

The inferred type of `ingredient` is `(String, Int)` , which is shorthand for `Tuple2[String, Int]` .

To represent tuples, Scala uses a series of classes: `Tuple2` , `Tuple3` , etc., through `Tuple22` . Each class has as many type parameters as it has elements.

Accessing the elements

One way of accessing tuple elements is by position. The individual elements are named `_1` , `_2` , and so forth.

```
println(ingredient._1) // Sugar
println(ingredient._2) // 25
```

Pattern matching on tuples

A tuple can also be taken apart using pattern matching:

```
val (name, quantity) = ingredient
println(name) // Sugar
println(quantity) // 25
```

Here `name` ’s inferred type is `String` and `quantity` ’s inferred type is `Int` .

Here is another example of pattern-matching a tuple:

```
val planets =
  List(("Mercury", 57.9), ("Venus", 108.2), ("Earth", 149.6),
        ("Mars", 227.9), ("Jupiter", 778.3))
planets.foreach{
  case ("Earth", distance) =>
    println(s"Our planet is $distance million kilometers from the sun")
  case _ =>
}
```

Or, in `for` comprehension:

```
val numPairs = List((2, 5), (3, -7), (20, 56))
for ((a, b) <- numPairs) {
  println(a * b)
}
```

Tuples and case classes

Users may sometimes find it hard to choose between tuples and case classes. Case classes have named elements. The names can improve the readability of some kinds of code. In the planet example above, we might define `case class Planet(name: String, distance: Double)` rather than using tuples.

More resources

- Learn more about tuples in the [Scala Book](#)

[← previous](#)

[next →](#)

Contributors to this page:

 ckipp01

 SethTisue

 mlachkar

 ashawley

 hatemogi

 NthPortal

 pkua

 fineconstant

 mghildiy

DOCUMENTATION

- Getting Started
- API
- Overviews/Guides
- Language Specification

DOWNLOAD

- Current Version
- All versions

COMMUNITY

- Community
- Mailing Lists
- Chat Rooms & More
- Libraries and Tools
- The Scala Center

CONTRIBUTE

- How to help
- Report an Issue

SCALA

- Blog
- Code of Conduct
- License

SOCIAL

- GitHub
- Twitter

