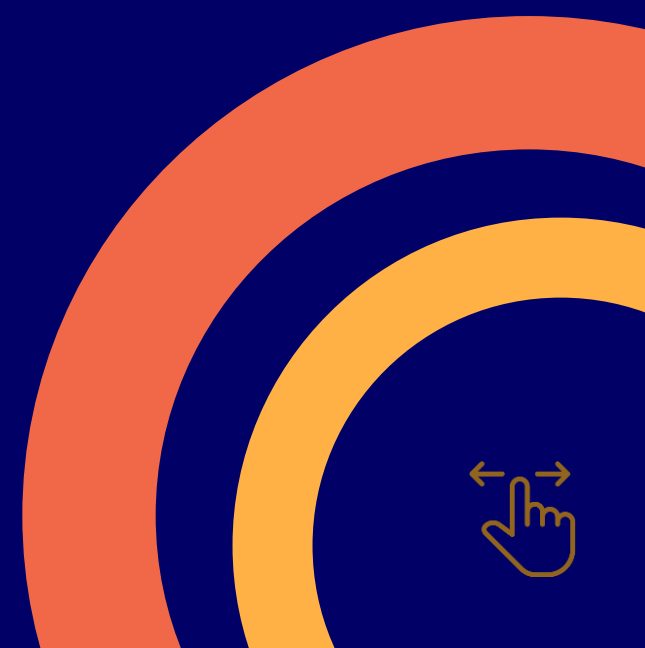




Extension Methods

در زبان Dart



وقتی امکان تغییر کد یک لایبرری رو نداریم اما نیاز داریم عملکرد جدیدی بهش اضافه کنیم، از Extension method استفاده می‌کنیم.



string_api.dart

```
extension NumberParsing on String {  
  int parseInt() {  
    return int.parse(this);  
  }  
  // ...  
}
```



main.dart

```
import 'string_api.dart';  
// ...  
print('42'.parseInt()); // Use an extension method.
```



نمی‌تونیم Extension method رو روی متغیر نوع dynamic فراخوانی کنیم، چون Extension method تنها روی مقادیری قابل فراخوانیه که نوع‌شون در زمان کامپایل قابل استنتاج و مشخص باشه. برای مثال کد زیر استثنای زمان اجرا داره:

```
main.dart

dynamic d = '2';
print(d.parseInt()); // Runtime exception: NoSuchMethodError
```

این محدودیت، یک ویژگی مهم Extension method رو توجیه می‌کنه: Extension method ها به صورت ایستا resolve میشن و برای همین فراخوانی‌شون مثل فراخوانی یک تابع ایستا، سریع هست.





main.dart

```
void main() {  
  print(3.addTwo);  
  print('Dash'.firstChar);  
  print(true.asInteger);  
}  
  
extension UsableElsewhere on String {  
  String get firstChar => substring(0, 1);  
}  
  
extension _UsableOnlyInThisFile on int {  
  int get addTwo => this + 2;  
}  
  
extension on bool {  
  int get asInteger => this == true ? 1 : 0;  
}
```

خارج از این فایل
قابل مشاهده است

فقط در این فایل
قابل مشاهده است

بدون نام (Unnamed)
و بنابراین فقط در همین
فایل قابل استفاده است



Operator Extension



main.dart

```
void main() {  
  final betty = Person('Betty Holberton');  
  final jean = Person('Jean Bartik');  
  
  var team = jean + betty;  
  var otherTeam = betty;  
  if (team > otherTeam) { ... }  
}  
  
extension GroupArithmetic on Group {  
  Group operator +(Person person) => Group({...people, person});  
  bool operator >(Group other) => people.length > other.people.length;  
}  
  
class Group {  
  final Set<Person> people;  
  const Group(this.people);  
}  
  
class Person {  
  final String name;  
  const Person(this.name);  
}
```



Generic Extension



Extension می‌تونه پارامتر generic داشته باشه:

```
generic_extension.dart

extension MyFancyList<T> on List<T> {
  int get doubleLength => length * 2;
  List<T> operator -() => reversed.toList();
  List<List<T>> split(int at) => [sublist(0, at), sublist(at)];
}
```

نوع T بر اساس نوع ایستای لیستی که Extension روی اون فراخوانی شده، محدود (bound) میشه.

