

Flat Map (flatMap, switchMap, concatMap, exhaustMap)

There are 4 “Flattening Strategies”

- **“Merge”** Strategy — не делает ничего особенного, просто подписывается на каждый из внутренних потоков
- **“Switch”** Strategy — отписывается от последнего, когда приходит новый
- **“Concat”** Strategy — подписывается на следующий поток, только когда полностью завершится предыдущий (значения других потоков кешируются и не теряются)
- **“Exhaust”** strategy — не подписывается ни на какой следующий поток, пока не закончится первый (значения следующих потоков не кешируются)

Subjects

- **Subject** — позволяет получать значения только после подписки
- **BehaviorSubject** — кеширует последнее значение и выдает его при подписке. Обязан быть проинициализирован стартовым значением. Часто его инициализируют значением null, что потом влечет дополнительные проверки в подписчиков типа `.pipe(filter(v=>v!=null),...`
- **ReplaySubject** — кеширует произвольное количество записей которые потом всплывают при подписке. Если проинициализировать значением буфера = 1, по сути будет вести себя также как и BehaviorSubject, но при этом не потребует проверок на null, так как не имеет инициализационного значения.
- **AsyncSubject** — ждет завершения и кидает только последнее значение

Join streams

Merge — когда очередность не важна.

```
//emit outputs from one observable
merge(
  first.pipe(mapTo('FIRST!')),
  second.pipe(mapTo('SECOND!')),
  third.pipe(mapTo('THIRD')),
  fourth.pipe(mapTo('FOURTH'))
);
//output: "FOURTH", "THIRD", "SECOND!", "FOURTH", "FIRST!", "THIRD", "FOURTH"
.subscribe(val => console.log(val));
```

Concat — когда важен порядок слушателей.

```
concat(
  of(1, 2, 3),
  // subscribed after first completes
  of(4, 5, 6),
  // subscribed after second completes
  of(7, 8, 9)
)
// log: 1, 2, 3, 4, 5, 6, 7, 8, 9
.subscribe(console.log);
```

Race — будет слушать только тот поток, с которого первым пришло сообщение, от остальных отпишется.

CombineLatest — кеширует последнее значение каждого из стримов, когда приходит очередное значение выдает пачку с последними значениями каждого из стримов.

Zip — выдает пачку, когда получит по одному новому значению из каждого из стримов.

ForkJoin — пачка из последних значений каждого из стримов.

WithLatestFrom — как и CombineLatest, кеширует значения всех потоков, но выдает пачку значений только когда происходит событие в "ведущем" потоке, на остальные потоки не реагирует.