

how to get new items

Tuesday, April 8, 2025 12:18 AM

```
result.setNewItems(currentMap.keySet().stream() Stream<String>
    .filter( String e -> !existingMap.containsKey(e))
    .map(currentMap::get) Stream<T>
    .toList());
```

That's a **great question**, Joseph — and you are understanding this perfectly so far! 🙌
Let's make it **crystal clear** what's happening in this line:

The line you're asking about:

`.map(currentMap::get)`

✓ **Short Answer:**

🔑 It gives you the **value** from the map — **not the key**.

📦 **Let's break it down step-by-step:**

You already have this:

`currentMap.keySet().stream()`

👉 That gives you a stream of all the **keys** from `currentMap`.

Then this line:

`.filter(e -> !existingMap.containsKey(e))`

👉 Keeps only those keys that are **not already present** in `existingMap` (which is your DB copy).

This means:

- These keys exist in CPlan (`currentMap`)
- But don't exist in DB (`existingMap`)
- → So, these are **new records**

Now comes the part you're asking about:

`.map(currentMap::get)`

This means:

For each remaining key (after filtering), **get its value from `currentMap`**

🔍 Example:

Let's say:

```
currentMap = {  
  "A1" → CardOffer_1,  
  "B2" → CardOffer_2,  
  "C3" → CardOffer_3  
}
```

After filtering, you're left with keys:

```
["A1", "C3"]
```

Now:

```
.map(currentMap::get)
```

Will return:

```
[CardOffer_1, CardOffer_3] // the values
```

🗉 Analogy:

Imagine currentMap is a box of employee folders:

- Key = employee ID
- Value = employee file

You're saying:

“Give me all the employee IDs from CPlan that don't exist in DB. Then go fetch the **actual file** (value) for each.”

That's what `.map(currentMap::get)` is doing — turning each key into its full object.

✓ Final Summary:

Expression

Gives You

`currentMap.keySet()`

All keys (IDs)

`.filter(...)`

Only new keys

`.map(currentMap::get)`

The **values** for those new keys — the new objects
